Enforcing Secure Access Control in Integrated Systems over the Internet

Jianming Yong [1]
Michael S Lane [1]
Mark Toleman [1]
Yun Yang [2]

[1] Department of Information Systems
Faculty of Business
University of Southern Queensland
e-mail: "yongj; lanem; toleman"@usq.edu.au

[2] CICEC - Centre for Internet Computing and E-Commerce
School of Information Technology
Swinburne University of Technology
e-mail: yyang@it.swin.edu.au

Abstract

This paper addresses some security issues raised by system integration over the Internet. System integration increases the complexity of security relationships exponentially, making it difficult to manage and enforce secure access control. By implementing a two-tier security architecture, all relevant systems security is enforced in a set of integrated systems.

Keywords

Security, Access control, System integration

INTRODUCTION

The impact of the Internet on the way organisations in the modern economies conduct business is undeniable. The networked knowledge-based economy has meant organisations have opened their information systems up to the outside world to facilitate the exchange of information and business transaction processing in real time. Systems integration over the Internet is becoming increasingly important for industry specific applications such as supply chain management, customer relationship management, procurement and the need for business partners to access and exchange information between ERP (Enterprise Resource Planning) systems. Therefore it is not surprising that system integration, including data integration, tool integration, and process integration, has become an important research topic. There has been significant effort from both industry and academia to facilitate systems integration over the Internet. From the industry perspective there are a number of standards organisations and vendor organisations which are working to provide solutions in system integration over the Internet as shown in Table 1.

Table 1 Systems integration - standards organisations and vendor organisations

Standards and vendors organisations	Role in systems integration
ebXML (http://www.ebxml.org),	XML-based infrastructure for all e-business
Biztalk (http://www.biztalk.org),	Enterprise application integration
cXML (http://www.cxml.org),	Data integration for e-commerce
SAP (http://www.sap.com),	Providing products of system integration
Peoplesoft (http://www.peoplesoft.com),	Providing products of system integration
OAG (www.openapplications.org),	eBusiness and Application Integration
RosettaNet (http://www.rosettanet.org)	Standards for integration in IT industry

From the academic perspective, there have been a number of research initiatives to address the challenges posed by system integration over the Internet as showed in Table 2.

Table 2 Systems integration - academic initiatives

Academic initiative	Contribution to systems integration
Garlic system (Haas et al.)	Query optimization for data integration
TSIMMIS (Garcia-Molina et al.)	Project of data integration
Information Manifold (Levy et al.)	Prototype of information system integration
Herms (Adali et al.)	Mediators for querying integrated systems
Razor (Friedamn and Weld).	Information gathering plan for an integrated system
Tulwila (Ives et al.)	Query execution for data integration

These research projects in combination with the technologies and standards initiatives are providing the technologies and industry specific standards that make systems integration over the Internet a reality.

However, little practical research has been conducted on security issues raised by systems integration over the Internet. In particular, one aspect of systems security, 'access control', which is a relatively simple function to administer for a single system, becomes incredibly complex to administer when beginning to integrate systems over the Internet. It is relatively easy to define secure access control for an individual computer or even a simple computer system. For example, a computer can easily be set up to control the access based on user's name and password. For a simple system, users can be divided into different groups and different groups will have different access privileges. However for a complex integrated system, especially one connected to the Internet, it is not easy to manage its secure access control across a range of potentially dispersed systems. At the same time, because integrated systems are being used to conduct significant business activities over the Internet, security has become a priority.

The security of an information system can be viewed from two aspects, internal security and external security. External security is concerned with preventing threats from outside. In most cases, because an integrated system is connected to the Internet, many users who do not belong to this integrated system still can reach certain part of this integrated system. In this paper we do not address security risks from outside threats and vulnerabilities because all Internet-connected computers or systems would have the same external threats and vulnerabilities. In this paper, we are concerned with the internal security of integrated systems in particular business context such as a supply chain etc. Internal security is concerned with implementing security policies so that internal users can "reasonably" use the integrated system. Within the integrated system, a good security management strategy will help to solve the security issues raised by access across a number of interconnected systems. This paper will focus on the proof of concept for a two tier security architecture that can manage the complexity of secure access control for multiple users of integrated systems over the Internet. It details the mathematical proof of concept which provides the foundations for a system administrator to enforce secure access control across a number of different systems which are integrated over the Internet for common business activities.

The structure of this paper is as follows: Firstly, the theories and frameworks which underpin the various aspects of secure systems access control are described and discussed. Then, a mathematical proof of concept for implementing secure access control for integrated systems is provided. Next, a two tier security framework for integrated systems is outlined and discussed with particular emphasis on the importance of tier 1 in enforcing secure access control. Some conclusions are drawn in relation to the two tier architecture for secure access control in integrated systems, with particular emphasis on the proof of concept of the tier 1 – framework for managing secure access control for multiple users across multiple systems over the Internet. Finally, implications of this research for current and future work on the security of the integrated systems over the Internet are discussed.

ACCESS CONTROL IN SYSTEMS SECURITY

Traditionally Role-based Access Control (RBAC) is frequently used by many organisations to implement their security strategy. Bacon, Moody and Yao (2002) has described a model of RBAC by OASIS(Organisation for the Advancement of Structured Information Standards). OASIS published a role-based access control architecture for achieving secure interoperation of services in an open, distributed environment, The aim of OASIS is to allow autonomous management domains to specify their own access control policies and to interoperate subject to service level agreements. Services define roles and implement formally specified policy to control role activation and service use; users must present the required credentials, in an appropriate context, in order to activate a role or invoke a service. All privileges are derived from roles, which are activated for the duration of a session only. In addition, a role is deactivated immediately if any of the conditions of the membership rule associated with its activation become false. These conditions can test the context, thus ensuring active monitoring of security. To support the management of privileges, OASIS introduces appointment. Users in certain roles are authorized to issue other users with appointment certificates, which may be a prerequisite for activating one or more roles. The conditions for activating a role at a service may include appointment certificates as well as prerequisite roles and constrains on the context. An appointment certificate does not

therefore convey privileges directly but can be used as a credential for role activation. Role-based access control, in associating privileges with roles, provides a means of expressing access control that is scalable to large numbers of principals.

Other access control methods include Temporal Role-based access control (TRBAC)(Bertino, Bettini & Ferrari 2000), Team-based Access Control (TMAC) (Georgiadis et al. 2001), Generalized Role-based Access Control (GRBAC) (Covington, Moyer & Ahamad 2000). TRBAC introduces periodic activation and deactivation, and role triggers for expressing temporal dependencies. Periodic activation and deactivation support time-limited authorization. TMAC is directly associated with a team, which is a group of users in specific roles, collaboratively working on a common task, the privileges that a user has are determined by his/her current team. GRBAC extends traditional RBAC by introducing object roles and environment roles in addition to subject roles. An object role represents a facet of the requested object. These roles are activated automatically by the system. The access control only can satisfy the classification of users and services statically. When integration of heterogeneous systems is required, how can the access control policies be actively achieved? This issue is discussed in later sections.

NEW ACCESS CONTROL REQUIREMENTS FOR INTEGRATED SYSTEMS

Before integration of a set of potentially quite different systems in terms of users needs and security, the access control requirements for users that will need to access other individual systems within the set of integrated systems needs to be established. It is important to establish the desired level of access control for each system for individual users. Otherwise, in large set of integrated systems, the number of individual users could be exponentially quite large resulting in performance issues. An in-depth requirements analysis of user access needs in the integrated set of systems is required. This will allow the system administrator to determine the access rights for each individual in the integrated system. An integrated system will consist of a large number of individual systems, which actually implement their own access control policies separately. After integration, some systems might need to access other systems for cooperation, which requires a mechanism to look after all individual systems as a whole to ensure the system security. Some research has addressed aspects of these requirements. Riet & Janssen (1998) identified database security from database systems to ERP systems,. Olivier (1998) addressed application-level security for workflow systems, and Soshi & Maekawa (1997) dealt with security architecture for open distributed systems. They all discussed security issues from certain aspects of system integration, but they did not deal with the overall requirements from the perspective of system integration. In particular, none addressed access control for an integrated system. The following sections address how that mechanism can be achieved and provide a proof of concept.

SECURITY ISSUES BEFORE SYSTEM INTEGRATION

Each system has its security policies. These security policies are centred around authentication and access control.

Authentication

Authentication is a process which determines whether a user's login information is valid, in other words, whether a user has a legal account in the system. Thus, the authentication policy will identify whether the user is the correct person to access the system. For example, most Unix systems base authentication on usernames and passwords. The system keeps a password data file, /etc/passwd. When a user wants to login this system, the system will prompt for the username and password. On login, the system will verify the inputs with /etc/passwd file which is an encrypted file. If successful, the user is a legal user and can use this system otherwise, the user would be denied access to the system. Thus the authentication helps to decide the set of legal users for the system. Assume the legal user set as $U(u_1, u_2, u_3, \ldots, u_n)$. This means there are n legal users in the system. Each user has a unique user name. Each user name will be associated with a password. If a user is allowed to access a system, what resources will be available to this user? In another words, what access privileges will the user have? This issue will be described in a latter section.

Access control

After a system authenticates a user and accepts access, the system must typically enforce certain limitations, restrictions or privileges associated with access to system resources. The example of Unix access control (Viega and Voas) is used to demonstrate how the authentication and access control can be implemented in an integrated set of systems over the Internet.

A formal model of system security based on the authentication and access control is developed. A system starts from a root user. The root user is configured at the birth of the system. From then, the root creates users, groups, etc:

$root \rightarrow U(u_1, u_2, u_3, \ldots, u_t)$ --- root creates users in the system
$root \rightarrow G(g_1, g_2, g_3, \ldots, g_m)$ --- root creates groups in the system
$g1 \leftrightarrow P1(p_{11}, p_{12}, \ldots, p_{1x})$ --- different groups will be assigned different privileges.
$g2 \leftrightarrow P2(p_{21}, p_{22}, \ldots, p_{2y})$ --- each privilege will be associated with the right to access/execute certain resources.

$gm \leftrightarrow Pm(p_{m1}, p_{m2}, \ldots, p_{mx})$
if $\partial u \in g_i$ then $u \leftrightarrow Pi(p_{i1}, p_{i2}, \ldots, p_{iy})$
if $u \in g_i \,\&\, g_j$ then $u \leftrightarrow Pi \bigcup Pj \quad i \neq j \quad 1 <= i,j <= m$
if $u \notin g_i, \quad 1 <= i <= m$ then u ↔ system default privileges for each user in the system

The default privileges are based on ownership, that is, if data/programs belong to a user, then this user can operate on his/her data/programs, and also the system supplies a minimum running environment for the user. Thus, if users have default privileges, they are generally less of a security threat to the system if their User IDs and passwords are stolen. However those group users with high privileges, especially root, for example, their User IDs and passwords are stolen, are a high security threat to the system. It is very important to know that different users in the system will present different security vulnerabilities to the system. According to these security concerns, different systems might implement different security policies. Thus after integration of these systems, security polices for the individual systems should not be reduced, and at the same time we expect that the whole integrated system could have an overall security mechanism to ensure that each system can interact with the other integrated systems securely. Before we work out an effective security solution for an integrated system, we need to know what the potential security threats are for this integrated system in order to implement its electronic business strategy in the current e-market environment. In the following sections, we will discuss the security threats from outside and inside respectively.

Threat from Inside

Threats from internal employees: Former employees who leave under bad circumstance are potentially problematic because of their knowledge of the system. They have the ability to collect a multitude of information about the company and achieve their own purposes. Current employees can also potentially wreak havoc on a company's computer system, especially by breaking into unauthorised data. Thus it is very important for the integrated system to implement effective security policies to prevent this happening.

Sniffers: Most computer networks, especially LAN, are configured as a multipoint so that the computers share a communication circuit. All connected computers can reach all data which is transmitted on the shared circuit. If software is used to intercept the sensitive information from the shared circuit, for example User IDs and password, etc, this behaviour is called sniffing. How security policies can be developed to prevent installation such software has become the concern of system integration over the Internet.

Viruses: Viruses can cause disasters for the system hardware and software. It is important to implement an effective security policy to prevent viruses from damaging system hardware and/or software. For example, each computer would have virus protection software installed to ensure that no virus can be brought into the system.

Threats from outside

Data theft: Someone outside is always interested in getting some sensitive data from the remote system for the personal intention. They try everything to steal these sensitive data. It can be prevented by implementing effective security policies.

Unauthorized access: Usually hackers try to gain access to system data files and resources from outside or from remote computers. Hacker attacks can be prevented by implementing effective security policies for the system.

Denial of Service Attacks: A denial of service attack is used by an outside individual to destroy, shutdown, or degrade a computer or network resource. The goal of such an attack is to flood the communication ports and memory buffers to prevent the receipt of legitimate messages and services of legitimate requests for connections. When a system is put into use for E-Commerce, its purpose is to provide its services to its legitimate customers or partners. But because of the denial of service attacks, the system's commercial function cannot be implemented. It is important to use an effective security policy, such as setting up good rules in firewall, to prevent this happening.

In order to effectively prevent all the threats whether from inside or from outside, when we implement systems integration, we have to develop an effective security polices for an integrated system. In the following sections, we address how to implement an effective security policy for an integrated system.

SECURITY MODELING AFTER SYSTEM INTEGRATION

Assume there are n systems, which will be integrated. We represent these systems as S1, S2, S3, …, Sn. S1has n_1 internal users, m_1 groups and p_1 privileges. S2 has n_2 internal users, m_2 groups and p_2 privileges. S3 has n_3 internal users, m_3 groups and p_3 privileges. Sn has n_n internal users, g_n groups and p_n privileges. Now for the integrated system, the number of internal users is a sum of all respective systems' internal user number: $n_1+n_2+n_3+\ldots+n_n$.

What privileges should the integrated system allow? It is obvious that all the privileges in the individual system should be included in the integrated system. Thus for the integrated system, privileges P will be defined as:

$P= \bigcup Pi \quad 1<=i<=n$ Pi represents the privileges of individual system i.

Next different groups for the integrated system are defined. Each group will be assigned relative privileges from P. These groups will operate all the individual systems in the same way. If a user belongs to one of these groups, then the user has all privileges, the group has been given, to operate on all available resources within the integrated system. For example, suppose there is a group which is responsible for web services. Now assume there are several users in this group. This group will have all authorities to operate on any web servers within an integrated system. Now any user from this group can operate on any web server within the integrated system. From this example, these groups exist beyond an individual system.

In other words, there is a need to know how integrated system security policies effectively cooperate with all previous individual security policies. The following security model for system integration is proposed: Two-tier security architecture for a large integrated system to enhance its overall security.

Tier 1 Security of the Integrated Systems

Tier 1 is concerned with the integrated system which has an overall security view for all integrated systems. The relationships of security elements in the integrated system are as follows.

Relationship 1 (R1), for users and systems, express this relationship as R1(U×S). U is the set of all users, U1,U2, …, Un, in the integrated system. S is the set of individual autonomous systems, S1, S2, S3, …, Sn. Ui is the set of users in autonomous Si, while 1<=i<=n.

Relationship 2 (R1), for groups and systems, express this relationship as R2(G×S). G is the set of all groups, G1, G2, G3, …, Gn, in the integration system. S is the set of individual autonomous systems, S1, S2, S3, …, Sn. Gi is the set of groups in system Si, while 1<=i<=n.

Relationship 3 (R3), for privileges and users, express this relationship as R3(P×U). P is the set of all privileges, P1, P2, P3, …, Pn, in the integrated system. U is the set of all the users in the integrated system. Pi is the set of the privileges in autonomous system Si, while 1<=i<=n.

Relationship 4 (R4), for privileges and groups, express this relationship as R4(P×G). P is the set of all privileges in the integrated system. G is the set of all the groups in the integration system.

Relationship 5 (R5), for groups and users, express this relationship as R5(G×U). G is the set of all groups in the integration system. U is the set of all the users in the integrated system.

Relationship 6 (R6), for privileges and systems, express this relationship as R6(P×S). P is the set of all privileges in the integrated system. S is the set of individual autonomous systems.

The relationships R1, R2, R3, R4, R5, and R6 and their contribution to the overall security concerns for the integrated system are not analysed in this paper and as such the mathematic proof of concepts are not included in this paper.

Tier 2 Security of Individual Autonomous System

This tier deals with security policies of individual autonomous systems. Thus different systems will have quite different security policies. These policies are based on various role-based access controls in the internal system. Most individual systems distinguish their legal users or groups by the User IDs/Group IDs as well as the passwords from unauthorised users/groups. In Tier 2, security policies only deal with users, groups and privileges, which are related to defined roles.

SECURITY FRAMEWORK FOR AN INTEGRATED SYSTEM

The security of an integrated system is divided into two tiers. Tier 2 handles individual autonomous systems. Tier 1 handles the whole integrated system. This relationship can be illustrated in Figure 1.

The security of an integrated system is divided into two tiers. Tier 2 handles individual autonomous systems. Tier 1 handles the whole integrated system. This relationship can be illustrated in Figure 1.

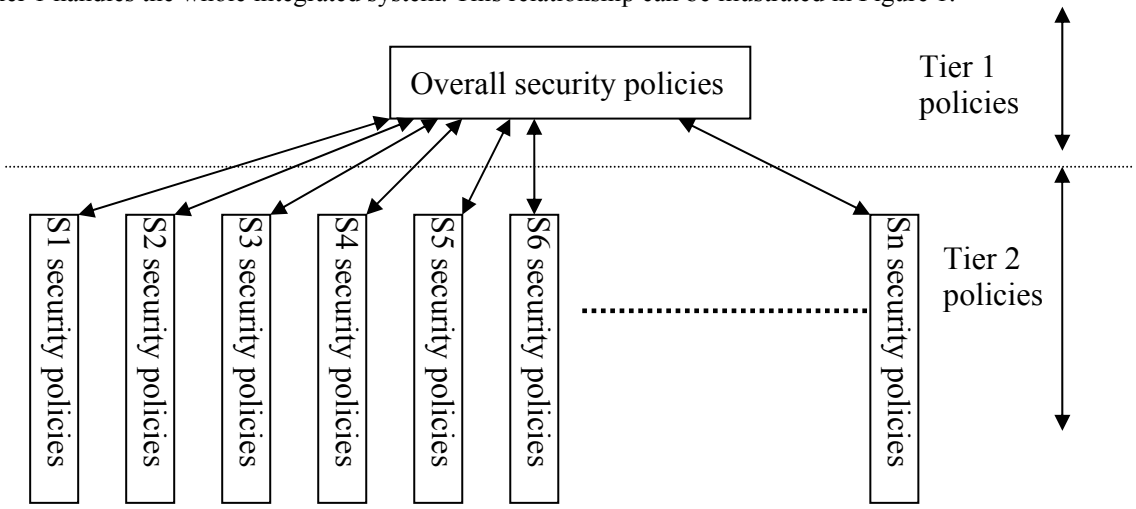

Figure 1 Logical security policies of an integrated system

From Figure 1, security policies can be implemented. First, if a user or a group wants to access any local system (S1, S2, S3, …, Sn), assume it is S1, and the user or group belongs to S1, then this user or group will only be checked by S1 security policies, which only relate to Tier 1's polices. This is exactly the same as prior to system integration. Now, suppose, a user or a group belongs to S1, and wants to access other systems (S2, S3, …, Sn), assume the user or group will access S3. Table 3 shows appropriate steps for this situation:

Table 3 Steps required to determine if an individual system user can access another system within an integrated set of systems using the Tier 1 security architecture

Steps	Actions
1	S3 finds that the user or the group is not one of its local users or groups.
2	S3 sends a request to the security server which is in charge of security policies of the integrated system. Based on the responses from the security server in Tier 1, S3 will know what privileges can be granted to that user or group.
3	S3 will allow the access to the user or group according to its privileges or reject the access to the user or group because the user or group cannot be authenticated by this higher hierarchical security server, which implements the security policies at Tier 1.

Thus for a whole integrated system, the security policies are divided into Tiers 1 and 2. Sometimes Tier 2 can satisfy the security requirements alone. Sometimes Tier 2 needs the cooperation of Tier 1 to satisfy the security requirements to allow non-local users/groups to access a local system. Through this two tiers' security mechanism, a robust security system has been established for an integrated system so that all the individual autonomous systems cannot only keep their original security properties but also flexibly accept access from outside their integrated business partners.

CONCLUSIONS AND IMPLICATIONS

The security requirements of an integrated system have been analysed. Six relationships (R1, R2, R3, R4, R5, R6), which correlate all the security requirements of individual autonomous systems (S) in a set of integrated systems, users (U), groups (G), and privileges (P). Based on these six relationships, a two-tiered security architecture for an integrated system is proposed, such a large scale system integrated over the Internet could serve E-Commerce purpose. This two-tiered security architecture ensures that the overall integrated system has a very reliable security policy with each individual autonomous system keeping its original security properties to service to its local users/groups but also utilising Tier 1's functionality to service all other users within the integrated system. Through this combination of Tier1 and Tier 2, the whole integrated system's security can be effectively implemented and at the same time any individual system's security can also be reinforced.

REFERENCES

Adali, S., Candan, K., Papakonstantinou, Y. and Subrahmanian, V. (1996) Query caching and optimization in distributed mediator systems. In Proc. Of ACM SIGMOD Conf. on management of Data, Montreal, Canada, 25(2), 137-146.

Bacon, J., Moody, K. & Yao, W. (2002) A Model of OASIS Role-Based Access Control and its Support for Active Security, ACM Transactions on Information and System Security, vol. 5, no. 4, 492-540.

Bertino, E., Bonatti, P. A. & Ferrari, E. (2001) TRBAC: A temporal role-based access control model, ACM Transactions on Information and System Security (TISSEC), 4(3), 191-233.

Covington, M. J., Moyer, M. J. & Ahamad, M. (2000) Generalized role-based access control for future applications, the 23rd National Information Systems Security Conference, Baltimore, Maryland, USA.

Friedman M. and Weld. D. (1997) Efficient execution of information gathering plans. In Proc. Of the International Joint conference in artificial Intelligence, Nagoya, Japan, 123-132.

Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajaraman, A., Sagiv, Y., Ullman, J. and Widom, J. (1997) The TSIMMIS project: Integration of herterogerneous information sources. Journal of Intelligent Information systems, 8(2): 117-132.

Georgiadis, C., Mavridis, I., Pangalos, G. & Thomas, R. K. (2001) Flexible team-based access control using contexts, the 6th ACM sysposium on Access Control Models and Technologies, New York, May 3-4, 21-30.

Haas, L., Kossmann, D., Wimmers, E., and Yang, J. (1997) Optimizing queries across diverse data source. In Proc. Of the Int. Conf. on Very Large Data Base (VLDB), Athens, Greece, 276-285.

Ives, Z. G., Florescu, D., Friedman, M. A., Levy, A. Y., And Weld, D. S. (1999) An adaptive query Execution system for data integration, In Proc. Of ACM SIGMOD Int. Conf. on Management of Data (SIGMOD), 299-310. ACM Press.

Levy, A.Y., Rajaraman, A., and Ordille, J. J. (1997) Querying heterogeneous information source descriptions. In Proceedings of the 15th International Joint Conference in Artificial Intelligence. Nagoya, Japan, 231-241.

Olivier, M. S. (1998) Specifying Application-level Security in Workflow Systems, the 9th International Workshop on Database and Expert Systems Applications (DEXA'98), Vienna, Austria, 346-351, IEEE.

Riet, R. V. D, Janssen, W. (1998) Security moving from Database Systems to ERP Systems, The 9th International Workshop on Database and Expert Systems Applications (DEXA'98), Vienna, Austria, 273-280, IEEE.

Soshi, M., Maekawa, M. (1997) The Saga Security System: A Security Architecture for Open Distributed Systems, the 6th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS '97), Tunis, TUNISIA, 53-58, IEEE.

Viega, J., Voas, J. (2000) The Pros and Cons of Unix and Windows Security Polices. IEEE IT Pro Septmeber|October, 40-45.

COPYRIGHT