SECURITY MODELS FOR WEB-BASED APPLICATIONS

Using traditional and emerging access control approaches to develop secure applications for the Web.

James B.D. Joshi, Walid G. Aref, Arif Ghafoor, and Eugene H. Spafford

THE RAPID PROLIFERATION of the Internet and the cost-effective growth of its key enabling technologies are revolutionizing information technology and creating unprecedented opportunities for developing large-scale distributed applications. At the same time, there is a growing concern over the security of Web-based applications, which are rapidly being deployed over the Internet [4]. For example, e-commerce—the leading Web-based application—is projected to have a market exceeding $1 trillion over the next several years. However, this application has already become a security nightmare for both customers and business enterprises as indicated by the recent episodes involving unauthorized access to credit card information. Other leading Web-based applications with considerable information security and privacy issues include telemedicine-based health-care services and online services or businesses involving both public and private sectors. Many of these applications are supported by workflow management systems (WFMSs) [1]. A large number of public and private enterprises are in the forefront of adopting Internet-based WFMSs and finding ways to improve their services and decision-making processes, hence we are faced with the daunting challenge of ensuring the security and privacy of information in such Web-based applications [4].

Typically, a Web-based application can be represented as a three-tier architecture, depicted in the figure, which includes a Web client, network servers, and a back-end information system supported by a suite of databases. For transaction-oriented applications, such as e-commerce, middleware is usually provided between the network servers and back-end systems to ensure proper interoperability. Considerable security challenges and vulnerabilities exist within each component of this architecture. Existing public-key infrastructures (PKIs) provide encryption mechanisms for ensuring information confidentiality, as well as digital signature techniques for authentication, data integrity and non-repudiation [11]. As no access authorization services are provided in this approach, it has a rather limited scope for Web-based applications.

The strong need for information security on the Internet is attributable to several factors, including the massive interconnection of heterogeneous and distributed systems, the availability of high volumes of sensitive information at the end systems maintained by corporations and government agencies, easy distribution of automated malicious software by malfeasors, the ease with which computer crimes can be committed anonymously from across geographic boundaries, and the lack of forensic evidence in computer crimes, which makes the detection and prosecution of criminals extremely difficult.

Two classes of services are crucial for a secure Internet infrastructure. These include access control services and communication security services. Access

control services protect Internet resources from unauthorized use, whereas communication security services ensure confidentiality and integrity of data transmitted over the network, in addition to nonrepudiation of services to the communicating entities. An important prerequisite for access control is user authentication, the process that establishes the identity of a user. In the context of the Internet, we assume authentication is handled by the communication security services.

Security in the Web Environment

End users are exposed to several security and privacy risks when using Web browsers, and browser vulnerabilities can result in compromising the security of a Web client [4]. Information about a user such as login name or machine name can be collected and used to profile the user, thus raising serious privacy concerns. Cookies, the data stored on the client's machine and exchanged between the Web client and the Web server to maintain connection information, can be used for the purpose of gathering such information. A source of vulnerability at the client site also comes from the use of executable content on the Web, such as Java applets, ActiveX controls, and the like. The current improvement in JDK1.2, which allows signed applets, requires the client to use a security policy for downloadable applets. Many sites also use push technology to deliver Web content to clients. This process can result in serious security breaches, as the content provider can exploit browser vulnerabilities by sending malicious executable code or by overwhelming the system by pushing a high volume of information.

Network servers are the places where most network services are located, such as the Web server, the mail server, and so forth. Firewall technology has become the most popular defense for these servers against the open untrusted Internet, as depicted in Figure 1. Though firewalls can prevent illegitimate traffic from traveling from the Internet to corporate networks, legitimate requests that pass through a firewall may be used for a data-driven attack on the networks or back-end systems [4, 5]. Configuration of firewalls and network servers is a formidable and error-prone task. This emphasizes the need to restrict or reduce complexity at the firewalls and networks and complement firewalls with robust host-based security.

In large corporate intranets, the insider attack is a growing security concern. A joint study on computer crimes conducted by the Computer Security Institute (CSI) and the FBI indicates that the most serious losses in enterprises occur through unauthorized access by insiders, and 71% of respondents had detected unauthorized access by insiders [6]. Therefore, there is a strong need for developing new access control models or extending the existing ones to neutralize security threats and address the diverse security requirements of Web-based applications.

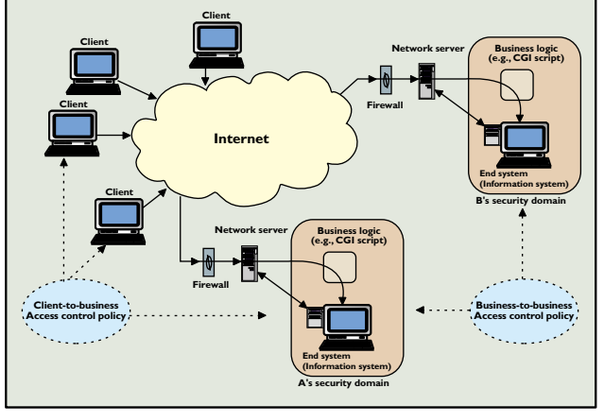
Justification for Access Control

Public-key infrastructures have been an important development for addressing the security concerns of

TERRY MIURA

Web applications. Users can be authenticated using PKI facilities, however, such facilities do not provide any mechanism for access control at the end systems. The fact that insider attacks constitute a considerable threat further accentuates the need for robust host-based security, whereby substantial authentication and access control services must be deployed at the host. The insider attack threat further demonstrates a strong need for efficient security management and administration functions in an enterprise. Host-based security can also help the network servers and firewalls for added intranet security. Security models that allow efficient security management and administration can also be extended for multidomain environments, where interactions among heterogeneous policy domains are intensive. Typical applications of multidomain environments include e-commerce, corporate databases, and digital government. Such applications need to interconnect and interoperate their business logic while protecting sensitive information.

Multilayered architecture for Web-based applications.



The Web primarily uses a hypertext approach for information dissemination. With the growth of e-commerce applications, the Web is rapidly being transformed into an activity- or transaction-intensive environment. Security models for hypertext-based systems are rare and still in their infancy stages. For the Web, access models and mechanisms should facilitate dynamic changes in the content and context of information, allow monitoring of the state of the system, and facilitate carrying out transactional activities. Existing access models lack these features.

Access Control Models

Information systems security refers to protection of information systems against unauthorized access to or modification of information, whether in storage, processing or transit, and against denial of service to authorized users, including measures necessary to detect, document, and counter such threats. The main goals of information security are confidentiality or secrecy, integrity, availability, accountability, and assurance. The goal of confidentiality is to ensure the information is not accessed by an unauthorized person. The goal of information integrity is to protect information from unauthorized modification. Information availability ensures the information is available when needed and is not made inaccessible by malicious data-denial activities. Information accountability ensures that every action of an entity can be uniquely traced back to the entity. Security assurance is the degree of confidence in the security of the system with respect to predefined security goals.

Several models have been proposed to address the access control requirements of distributed applications. Traditional access control models are broadly categorized as discretionary access control (DAC) and mandatory access control (MAC) models. New models such as role-based access control (RBAC) or task-based access control (TBAC) models have been proposed to address the security requirements of a wider range of applications. We briefly highlight the main differences among these models and provide an assessment of their suitability for supporting Web-based applications.

Discretionary Access Control (DAC) Model

In DAC models, all the subjects and objects in a system are enumerated and the access authorization rules for each subject and object in the system are specified. Subjects can be users, groups, or processes that act on behalf of other subjects. If a subject is the owner of an object, the subject is authorized to grant or revoke access rights on the object to other subjects at his discretion. DAC policies are flexible and the most widely used for Web-based applications. However, these policies do not provide high security assurance. For example, DAC allows copying of data from one object to another, which can result in allowing access to a copy of data to a user who does not have access to the original data. Such risks can propagate to the entire Web environment, causing serious violation of security goals.

Among the existing representations of DAC models, a noticeable one is the HRU (Harrison, Ruzzo and Ullman) access control matrix (ACM) model [5]. The matrix specifies access rights of subjects for accessing objects in the system. In conjunction with

ACM, the HRU model uses a set of commands to construct the overall authorization scheme. Safety in HRU is in general undecidable. The basic safety problem is to determine whether there exists a reachable state in which a particular subject possesses a particular privilege that it did not previously possess.

Several new models have recently been proposed for systems for which safety problems are decidable and tractable. Most of these models are based on the notion of security type, and include the Schematic Protection Model (SPM), the Typed Access Matrix (TAM) model, and the Dynamically Typed Access Control (DTAC) model [7]. Unlike SPM and TAM, which have subject types and object types, DTAC makes no distinction between subjects and objects. The DTAC model uses a dynamic typing mechanism that makes it suitable for a dynamic environment such as the Internet. In DTAC, a safety invariant is maintained by carrying out static analysis and dynamic checks on the security aspects of the system. This feature gives DTAC the power to model task-based security [7]. By grouping entities into types, this model can reduce the size of the configuration and can enhance the administrative functions. While these extensions are intended to broaden the scope of ACM-based models, they are still in the theoretical development stage, with little or no experimental results.

Mandatory Access Control (MAC) Model

In a MAC model, all subjects and objects are classified based on predefined sensitivity levels that are used in the access decision process. An important goal of a MAC model is to control information flow in order to ensure confidentiality and integrity of the information, which is not addressed by DAC models. For example, to ensure information confidentiality in defense applications, a MAC model can be implemented using a multilevel security mechanism that uses no read-up and no write-down rules, also known as Bell-LaPadula restrictions. These rules are designed to ensure that information does not flow from a higher sensitivity level to a lower sensitivity level. To achieve information integrity, the access rules are formulated as no-read-down and no-write-up [8]. The goal in this case is not to allow the flow of low integrity information to high integrity objects. The Chinese Wall policy, which addresses conflict of interest issues relevant to financial industries, can also be implemented using a MAC model [8]. For Web-based applications, multilevel classification of information may be an essential requirement that can be enforced by a service provider to distinguish among the users and the type of information being accessed.

Unlike DAC, MAC models provide more robust protection mechanisms for data, and deal with more specific security requirements, such as an information flow control policy [8]. However, enforcement of MAC policies is often a difficult task, and in particular for Web-based applications, they do not provide viable solutions because they lack adequate flexibility. Furthermore, organizational security needs are often a mixture of policies that may need to use both DAC and MAC models, which necessitates seeking solutions beyond those provided by DAC and MAC models only. Originally, these models were not intended for Web-based applications. In particular, their design philosophy was not intended to serve hypertext-based systems, which is common in a Web-based environment. The hypertext information model uses special objects such as links, frames or slots, document nodes, and so forth, all of which need to be protected [2]. Hypertext systems are characterized by three features, which include information about the connections among data items, their unique navigational aspects, and the absence of a schema. Although extensions enabling these models to address security concerns have been proposed in the literature, more challenging issues such as control of copy and dissemination of information, active object management, and support for multiple data types and complex interrelationships have yet to be explored in order to develop viable solutions for Web-based applications.

Role-based Access Control (RBAC) Model

Role-based access control (RBAC) models are receiving increased attention as a generalized approach to access control because they provide several well-recognized advantages [7]. As roles represent organizational responsibilities and functions, a role-based model directly supports arbitrary, organization-specific security policies. The RBAC models have been shown to be "policy-neutral" [7] in the sense that using role hierarchies and constraints, a wide range of security policies can be expressed, including traditional DAC and MAC, and user-specific ones. Security administration is also greatly simplified by the use of roles to organize access privileges. For example, if a user moves to a new function within the organization, the user can simply be assigned to the new role and removed from the old one, whereas in the absence of an RBAC model, the user's old privileges would have to be individually revoked, and new privileges would have to be granted. Special administrative roles can be designated to manage other roles. Such administrative

roles can be hierarchically organized to provide a well-organized security management structure, which is desirable in large Web-based enterprises where security management becomes a complex task. Several authorization-constraints may need to be enforced in an organization to protect information misuse and prevent fraudulent activities. A typical authorization constraint, which is relevant and well-known in the security area, is separation of duties (SOD). Reducing the risk of fraud by not allowing any individual to have sufficient authority within the system to single-handedly perpetrate fraud is the intent of SOD. Such constraints can be easily expressed using an RBAC model through SOD constraints on roles, user-role assignments and role-privilege assignments. Furthermore, using assigned roles, users can sign on with the least privilege set required for any access. In case of inadvertent errors, such least privilege assignment can ensure minimal damage.

An important consideration in RBAC systems is the possible temporal constraints that may exist on roles, such as the time and duration of role activations, and timed-triggering of a role by an activation of another role [7]. Using an RBAC model is a highly desirable goal for addressing the key security requirements of Web-based applications in general, and WFMSs in particular. Roles can be assigned to workflow tasks so that a user with any of the roles related to a task may be authorized to execute it. However, the challenge is to develop a robust RBAC framework to handle the complex security needs of a WFMS, where temporal, nontemporal, and dependency constraints among roles and tasks exist.

A recent implementation of an RBAC system for the Web environment (RBAC/Web) has been reported in [3]. The implementation consists of a Web server to enforce RBAC policies and an administrative tool to allow security administration. The system places no requirements on the browser. When a user issues an access request, a role is assigned to the requester after establishing a session using the available authentication and confidentiality services. These services include the Secure Socket Layer (SSL), Secure HTTP (SHTTP), and an authentication mechanism that uses username/passwords. To ensure better administration, RBAC/Web can be integrated with an administrative model such as URA97 (User-Role Assignment '97), which uses administrative roles to manage other roles.

Several other RBAC implementations have been developed, including the hyperDrive System developed by the Internal Revenue Service, TrustedWeb, getAccess by enCommerce, and SESAME. TrustedWeb requires specific software in the client machine. The I-RBAC (RBAC for an intranet) model [9] uses software agents to distinguish between the local role hierarchies and the global role hierarchy of the entire intranet. The local network objects are known only to the local servers, whereas the global network objects are known throughout the intranet. Information about mapping between the global roles and local roles is kept in a database and is used when a global network object needs to access an object on another server. The disadvantage of I-RBAC is that maintaining consistent information about the roles becomes difficult as the number of roles increases.

A key feature of RBAC is its potential support for a multidomain environment, which makes it an attractive candidate for Web-based applications. Role-hierarchy mapping between two RBAC-based policy domains can be used to define a metapolicy for secure interoperation.

Access Control Models for Tasks and Workflows

The models discussed previously use the subject-object view toward security. These models have a limited scope and are not flexible enough to allow access policies based on the content of information or the nature of tasks/transactions in a WFMS. WFMSs have emerged as a key technology for enabling activity-intensive Web applications that require extensive automated transactional functions. Such applications typically constitute a complex mix of tasks and transactions that span departmental, organizational, geographical and cultural boundaries, further exacerbating the complexity of Web security. Although there exists a pressing need to develop access control models that can provide strong support for activity and task-intensive applications, no existing access control models have the capability to address the major security issues related to these applications.

Several authorization models related to WFMSs have been proposed. A viable approach to enforce arbitrary security requirements during the execution of workflow tasks is to assign roles to workflow tasks [1]. The workflow tasks of Web-based applications can be distributed over multiple heterogeneous security domains, and may have strict temporal and inter-task dependency constraints. In addition, roles assigned to tasks may have their own temporal and nontemporal constraints that may be static or dynamic in nature. Although the use of an RBAC framework for ensuring workflow security has been proposed in the literature, substantial extensions are needed to address security issues related to Web

applications and WFMSs.

To address the security issues related to task-oriented systems and to effectively serve the unique needs of such systems, researchers in [10] propose a family of task-based access control (TBAC) models that constitutes four models arranged in form of a hierarchy. The TBAC0 model represents the base model that provides the basic or the minimum facilities, such as tasks, authorization steps, and their dependencies. The TBAC1 model is an extension of TBAC0 that includes the composite authorizations of two or more authorization steps. The TBAC2 model is another extension of TBAC0 that allows both static and dynamic constraints. The TBAC3 model is a consolidated model that has features of both the TBAC1 and TBAC2 models.

Approaches and features compared.

Appraoch	Features from Web Perspective
DAC	• Ownership-based, flexible, most widely used, does not provide high degree of security, and hence low assurance • Typed versions such as SPM, TAM, and DTAC are expressive but have little or no experience base • DTAC can handle dynamic changes and task-based control (better than RBAC) • Most cannot be used where classification levels are needed • Typed versions have tried to include classification levels
MAC	• Administration-based • Information flow control rules • High level of security, and hence high assurance, but less flexible
RBAC	• Policy-neutral/flexible • Principle of least privilege • Separation of duty • Easy administrative features • Able to express DAC, MAC, and user-specific policies using role hierarchy and constraints • Can be easily incorporated into current technologies • Good for multidomain environments when policies are expressed using role hierarchies and constraints
Access control Tasks/Workflow	• Task-oriented authorization paradigm • RBAC is highly beneficial for WFMS • TBAC is at an initial stage of development (no formalism yet) • A key component for success of transaction-intensive e-commerce, medical applications, and so forth
Hypertext-based authorizations	• Approach based on hypertext model or document characterization-infancy stage • Essential for providing formal base for the security of Web objects including links and nodes; access modes include browsing and viewing
Certificate-based	• Utilization of existing PKI facilities • Complements the host's access control model • Can use trust centers in the Web
Agents	• Adaptability and mobility • Mobile agents introduce new security issues • Can be considered a complementary system-building paradigm, rather than a model or mechanism for specific security implementation • May be useful in multidomain environments (for example, for policy negotiation)

Agent-based Approach

With the increase of Internet applications, software agents are becoming popular as an emerging system-building paradigm. This paradigm can be effectively used to provide security features for Web applications. An agent is a process characterized by adaptation, cooperation, autonomy, and mobility. Some agent communication language can be used to negotiate policies during conflicts for secure interoperation among participating policy domains. Agents can be assigned security enforcement tasks at the servers and client machines. Although mobility and adaptability are essential to the efficient use of Internet resources, they pose several security threats. For example, an agent can engage in malicious behavior, thus disrupting normal operation of the host. Similarly, a host may be able to affect the activity of an agent by denying required access to local information resources.

Certificate-based Approach

Public-key infrastructure technology is maturing, and the use of PKI certificates is expected to be ubiquitous in the near future. Certificates issued by a PKI facility can be used for enforcing access control in the Web environment. An example is the use of an extended X.509 certificate that carries role information about a user [7]. These certificates are issued by a certification authority that acts as a trust center in the global Web environment.

The use of public-key certificates is suitable for simple applications. These techniques can be used to either support a host's access control method by carrying access control information or provide a separate access control mechanism based on trust centers.

Discussion

We have discussed several access control models and approaches that can be used to disseminate and

exchange information securely, and allow secure execution of WFMSs. However, comprehensive frameworks are needed to address the multifaceted security issues related to Web-based applications. In particular, robust access control models are needed to allow: controlled access, dissemination and sharing of information based on content, context, or time; secure execution of tasks and workflows; secure interoperation in a dynamic distributed enterprise environment; and efficient management and administration of security.

The table summarizes the key features of each access control model and approach discussed here. The DAC and MAC models lack capabilities needed to support security requirements of emerging enterprises and Web-based applications. Newer models such as SPM, TAM, and DTAC have the potential to support Web-based applications. In particular, DTAC's feature of using safety invariants in a dynamic environment is highly desirable for dynamic and transaction-intensive workflow-based applications. Hypertext-based authorization models are essential for secure composition and distribution of complex Web documents. However, these security models are yet to be fully developed and assessed for their efficacy and viability to support Web-based applications.

Achieving secure interoperation in a heterogeneous Web environment is a difficult task, because of the inherent dynamism and evolving security requirements of the underlying autonomous administrative domains. Using RBAC models and software security agents are suitable approaches for such environments. The RBAC models have several desirable features such as flexibility, policy-neutrality, better support for security management and administration, the principle of least privilege, and other aspects that make them attractive candidates for developing secure Web-based applications. In addition, they can represent traditional DAC and MAC as well as user-defined or organization-specific security policies. Furthermore, an RBAC model provides a natural mechanism for addressing the security issues related to the execution of tasks and workflows. A key advantage of RBAC models is the ease of their deployment over the Internet. The use of RBAC in conjunction with PKI facilities can provide a pragmatic approach to addressing issues related to security of distributed Web-based applications and WFMSs. The TBAC models represent efforts toward finding effective security solutions for the unique needs of task-based systems. However, they are still in the early stages of development.

Conclusion

We have presented a comparative assessment of existing security models in terms of supporting Web-based applications and WFMSs. Although there has been phenomenal growth of Web-based applications on the Internet, access control issues related to Web security have largely been neglected. The RBAC models are expected to provide a viable framework for addressing a wide range of security requirements for large enterprises. However, several extensions to the existing RBAC models are needed to develop workable solutions to adequately address such needs. C

References

1. Bertino, E., Ferrari, E., and Atluri, V. The specification and enforcement of authorization constraints in workflow management systems. *ACM Trans. Info. Syst. Security 2*, 1 (Feb. 1999), 65–104.
2. Bertino, E., Pagani, E., Rossi, G.P., and Samarati, P. Protecting information on the Web. *Commun. ACM 43*, 11 (Nov. 2000), 189–199.
3. Ferraiolo, D.F., Barkley, J.F., and Kuhn, D.R. A role-based access control model and reference implementation within a corporate intranet. *ACM Trans. Info. Syst. Security 2*, 1 (Feb. 1999), 34–64.
4. Garfinkel, S. and Spafford, E.H. *Web Security and Commerce.* O'Reilly and Associates, Sebastopol, CA, 1997.
5. Harrison, M.H., Ruzzo, W.L., and Ullman, J.D. Protection in operating systems. *Commun. ACM 19*, 8 (Oct. 1976), 461–471.
6. Power, R. *Tangled Web: Tales of Digital Crime from the Shadows of Cyberspace.* Que/Macmillan Publishing, Aug. 31, 2000.
7. *Proceedings of The Fifth ACM Workshop on Role-based Access Control.* Berlin, Germany, Jul. 2000.
8. Sandhu, R. Lattice-based access control models. *IEEE Computer 26*, 11 (1993).
9. Tari, Z. and Chan, S. A role-based access control for intranet security. *IEEE Internet Computing* (Sept.–Oct. 1997), 24–34.
10. Thomas, R.K. and Sandhu, R.S. Task-based authorization controls (TBAC): A family of models for active and enterprise-oriented authorization management. In *Proceedings of the IFIP WG11.3 Workshop on Database Security* (Lake Tahoe, CA, Aug. 1997).
11. Wing, P. and O'Higgins, B. Using public-key infrastructure for security and risk management. *IEEE Communications Magazine*, (Sept. 1999), 71–73.

James B.D. Joshi (joshij@ecn.purdue.edu) is a graduate student in the School of Electrical and Computer Engineering at Purdue University in West Lafayette, IN.

Walid Aref (aref@cs.purdue.edu) is an associate professor in the Department of Computer Science at Purdue University in West Lafayette, IN.

Arif Ghafoor (ghafoor@ecn.purdue.edu) is a professor in the School of Electrical and Computer Engineering at Purdue University in West Lafayette, IN.

Eugene H. Spafford (spaf@cerias.purdue.edu) is a professor and the director of the Center for Education and Research in Information Assurance and Security (CERIAS) at Purdue University in West Lafayette, IN.

This work has been supported by a grant from CERIAS, Purdue University.