Components- Based Access Control Architecture

Adesina S. Sodiya and Adebukola S. Onashoga
Department of Computer Science, University of Agriculture, Abeokuta, Nigeria

sinaronke@yahoo.co.uk; bookyy2k@yahoo.com

Abstract

Ensuring adequate security of information has been a growing concern of individuals and organizations. There is then the need to provide suitable access control mechanism for preventing insider abuses and ensuring appropriate use of resources. This paper presents an access control scheme that adopts the techniques of Role-Based Access Control (RBAC), Purpose-Based Access Control (PBAC), Time-Based Access Control (TBAC) and History-Based Access Control (HBAC) as components to form an integrated Components-based Access Control Architecture (CACA). In CACA, an Access Control Score (ACS) is computed from the combined access control techniques. CACA also combines ACS with the sensitivity nature of system resources before a level of access is granted. The architecture was implemented within a payroll system developed using JAVA and SQL. Using usability testing, the evaluation of CACA showed 92% reduction in insider abuses and misuse of privileges. This shows that CACA can provide higher level of security access as against what used to exist.

Keywords: Security, Access Control, Resource Sensitivity, Insider Abuses, Privileges

Introduction

The most widely used mechanism for preventing unauthorized access to systems is Identification and Authentication. Identification is the process where a user gives a valid and recognized identity to the system and authentication is the process whereby the system verifies the supplied identity. Access control, which is the concept of authorization, is concerned with determining the allowed activities of legitimate users (Scott-Chapman, 2006). The major aim of access control systems is to protect system resources against inappropriate and undesired user access. To reduce the security risks on computer systems as much as possible, there is a need to define who is allowed to access the stored information, which system resources the user is allowed to access, and what type of actions he/she is allowed to perform on those resources. Access control is one of the most important security mechanisms in the network environment and web services. Access control consists of policy, model and mechanism. The policy is the statement of what is, and what is not allowed, while the model is the formal representation of the security policies enforced by the system and is useful for proving the theoretical limitations of a system. The mechanism is a method, tool, or procedure for enforcing the Access Control Policy (NISTIR, 2006).

Access control systems are generally classified as Discretionary Access Control (DAC) and Non-Discretionary Access Control (NDAC). In DAC, the object owner or anyone else who is author-

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

ized to control the object's access specifies who have access to the object or specifies the policies. All access control policies other than DAC are categorized as NDAC. In NDAC, policies are rules that are not specified at the discretion of the user. Some examples of NDAC are:-

i. Mandatory Access Control (MAC):- This technique specifies that access control policy decisions are made by a central authority and not by the individual owner of the object. For example, the individual owner of an object can not specify whether an object is Top Secret and so on.

ii. Role-based Access Control (RBAC):- This describes the technique in which categories and duties of users are considered before permissions are granted to invoke an operation. The different categories are predefined, and have varying amount of privileges. The users will be placed in these categories. A user may be assigned many roles, but may not execute all his roles at the same time.

iii. Purpose-based Access Control (PBAC):- In this case, access is granted based on the intentions of the subjects. Each user is required to state his or her access purpose when trying to access an object. For example, in a school environment, data is collected for registration, checking of results, and so on. The system validates the stated access purpose by the user to make sure that the user is indeed allowed for the access purpose.

iv. History-based Access Control (HBAC):- This describes an access control technique in which access is granted based on the previous records. A subject is granted access to an object if logical the subject have previous access to the object to some reasonable threshold.

v. Temporal Constraints Access Control (TCAC):- This involves access control policies in which time restrictions are attached resource access. For example, some activities must be performed within a reasonable period.

vi. Rule-based Access Control (RuBAC):- This describes the technique that allows subjects or users to access objects based on pre-determined and configured rules. RuBAC is a general term for access control system that allows some form of organization-defined rules.

However, most of the current access control techniques are not completely adequate to ensure effective access control to computer resources because they are still faced with some problems. Some of the problems are:-

- the difficult to tailor access based on various attributes or constraints
- the difficulty in encapsulating all possible job functions and requirements to access objects
- inadequate capability of the administrator to compose all rules that covers the necessary access constraints and permission between subjects, operations and objects because of dynamic nature of operation
- non-prevention unauthorized access
- denial of authorized access because of complicated rules, etc.

This work presents an integrated access control architecture that combines or integrates four access control techniques (called components) for effective access control. It also relates access control to the sensitivity of the object or resource.

The rest of this paper is organized as follows. The second section discusses related works. The Components-based Access Control Architecture (CACA) is presented in the third. The implementation and evaluation are presented in the fourth section. The fifth section presents conclusion and future work.

Related Works

Many researchers have contributed in the past in designing effective access control systems. Typical Role-Based Access Control (RBAC) systems were presented in Ferraiolo et al. (2003) and in Cavale and McPherson (2003). They presented architecture for ensuring separation of duties in order to control access to computer resources. The problem with RBAC is that it is difficult in some cases to encapsulate all permissions to perform a job function. In fact, role engineering has turned out to be a difficult task (NISTIR, 2006).

Lattice-based access control models were described in McCue (2000) and Pleeger and Pfleeger (2003). In Lattice-based models, subjects and objects are assigned security labels from a partially ordered universe, which is a lattice. Nowadays, lattice-based access control is not widely used because the practical implementation is difficult as the size of the security lattice increases (Obedkov et al., 2009).

Scott-Chapman (2006) in his thesis proposed a perimeter based community-centric, access control system that makes use of an access control tree to represent privilege. The tree is rendered in such a way that the location based relationships of the objects in their respective security perimeters are preserved. Each node of the tree represents an object, and each branch represents an access operation. The access control tree is able to dynamically determine capability by consolidating security information from external data sources, software agents, and location based sensors. This access control was typically based on physical access control.

Menzel et al. (2007) proposed a Two-Level Access Control (2LAC) architecture for cross-organizational federated service composition independent from local access control models. The architecture helps to prevent information leakage and allowing authorization-based cross-organizational service invocation. This architecture provides a list of access control and authorization requirements for federated composite web service frameworks, and an evaluation and categorization of existing Service Oriented Architecture (SOA) security frameworks and their capabilities to support cross-organizational federated composite services.

Obedkov et al. (2009) described the building of access control models using attribute exploration. The attribute exploration, which is a concept from formal language, was adopted for improving lattice-based access control models. But, the real implementation of access control model using attribute exploration has not been realized.

Yang et al. (2008) presented the division of purpose into intended purpose and access purpose corresponding to data object and the data access, which makes access control clearer. The intended purpose specifies the intended usage of the data object. An access purpose, on the other hand, specifies the intentions for which a given data object are accessed. Each user is required to state his or her access purpose along with the data request. The system validates the stated access purpose to make sure that the user is indeed allowed for the access purpose. In addition, only when an access purpose is compliant with its intended purpose that access is allowed. Their work was only attributed to medical care scenario and did not also consider the changing nature access purpose.

Components-Based Access Control Architecture (CACA)

As shown in Figure 1, CACA components are divided into two main categories:-

a. **Combined access control techniques**

As mentioned earlier, four known access control techniques are combined within CACA. The techniques are RBAC, PBAC, HBAC and TBAC.

b. **Resource Sensitivity**

CACA also considers the sensitivity of an object in granting access to an object. Since all objects are not equally sensitive, in CACA, objects are classified as extremely sensitive, sensitive, and insensitive.

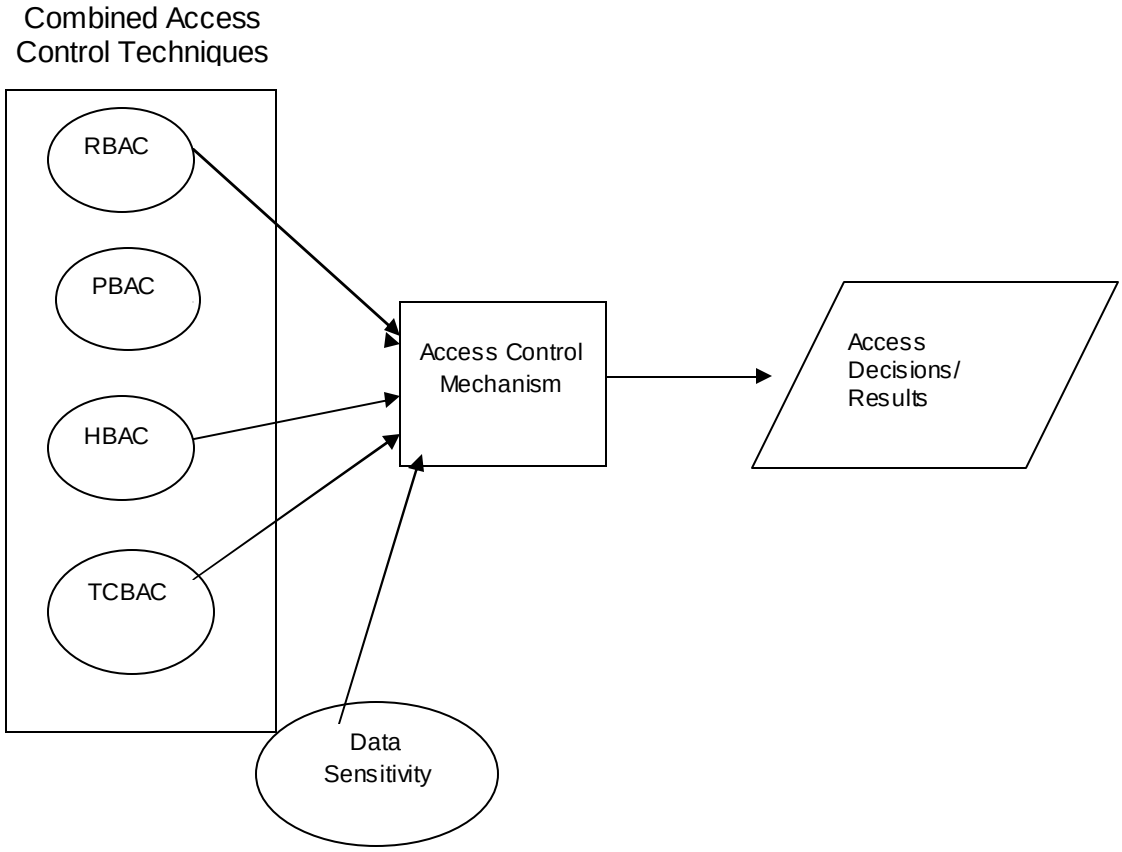

Figure 1: Components-based Access Control Architecture (CACA)

The access control mechanism involves the computation of Access Control Score (ACS), which determines the level of access of subjects or users.

Computation of Access Control Score (ACS)

Step 1: Computation of Capability Score (CS)

Capability specification describes how roles or subjects are mapped to operations and purpose. CACA capability Specification function (CSF) is given as

$A_{i(1..k)} \Longrightarrow R_{j(1..q)}: (O_1, O_2, \ldots O_n) : (P_1, P_2, \ldots P_3), \quad n \in S, m \in PP$

Where A_i = Subject i (i from 1 to k)

R_j = Object j (j from 1 to q)

O_i = Set of all possible operations on object i by subject i

S = Total number of all possible operations

PP = Total number of all possible purpose

k = Total number of users or subjects

q = Total number of objects

The CSF is used to compute access CS.

If there is match between the subject and an object as defined by the CSF,

Then CS = 1,

Otherwise, CS = 0

Step 2: Time-based Restrictions

CACA recognises that some activities within organisations are performed within specific periods or time. There are two types of time-based restrictions that are incorporated into CACA.

i. Subject-Based Restriction (SBR):- This states that a subject must request an object at a particular period of time. It is represented as

$A_i: (t_1..t_{11})$ --- for single period definition

and $A_i: (t_1..t_{11}, t_2..t_{22} \ldots t_n..t_{nn})$ --- for multiple periods definition

where $(t_1..t_{11}, t_2..t_{22} \ldots t_n..t_{nn})$ represents periods

ii. Object-Based Restriction (OBR):- This states that a particular operation must be performed on an object at a particular period. It is represented as

$O_i: (t_1..t_{11})$ --- for single period definition

and $O_i: (t_1..t_{11}, t_2..t_{22} \ldots t_n..t_{nn})$ --- for multiple periods definition

The computation of the Period Complaint Score (PCS) is described as follows:-

For SBR,	PCS = 0,	If there is no match
	PCS = 1,	If there is match
For OBR,	PCS = 0,	If there is no match
	PCS = 1,	If there is match

The Average Period Compliant Score (APCS) is

APCS = (PCS(SBR) + PCS(OBR))/2

Step 3: History-Based Check

CACA also has a technique for checking and confirming the extent of the previous activities of subjects. The History-Based Check Score (HBCS) is calculated as follows:-

Let aa_i be the number of times subject Ai has accessed object Oi

Let aa_{jn} be the total number of times that object j has been accessed

Then,

The Average number of Access to Subject R_j (AAR_j) = aa_{jn} / n ,

Where n represents the total number of users

The HBCS are then given as follows:-

HBCS = 0, if no previous record

HBCS = 1, if $aa_i <= (1/3) * AAR_j$

HBCS = 2, if $(1/3) * AAR_j < aa_i <= (2/3) * AAR_j$

HBCS = 1, if $aa_i > (2/3) * AAR_j$

Step 4: Computation of ACS

As stated earlier, the sensitivity levels that are attached to objects are:-

- Extremely sensitive
- Sensitive
- Insensitive

ACS is computed as

ACS = (CS + APCS + (HBCS)/3)/3

If ACS > 2/3, then access is granted to all objects

1/3 <ACS >= 2/3, then access is granted to sensitive and insensitive objects alone

ACS <= 1/3, access not granted

Implementation and Evaluation

CACA was implemented using JAVA and SQL. CACA was built into a commercial Payroll system designed by our team. The payroll system was implemented using JAVA and SQL. The evaluation of CACA was based on the recently conducted usability study conducted on the organization that recently implemented the new payroll system because of its security features. It was purposely acquired by the organization in other to correct insider abuses and misuse of privileges. The most frequent insider abuses were because of financial gain at the detriment of the organization. The evaluation result showed a reduction in insider abuses to about 92%. In fact, this organization is of those we are using as test cases before final commercial deployment of the system.

Future Works and Conclusion

It might be necessary for researchers to identify and consider more key security elements of computer systems in building efficient access control system. Researchers in computer security might need to combine authentication, access control and intrusion detection together so as to provide adequate security to computer-based systems. Also, most access control systems do not consider emergency. For example, a job might be needed immediately and the user who has the privildge to execute the job might be out of reach. What is normally done in most cases is to assign the

right to another user so as to get the job done. There is the need for an intelligent way of accomplishing this.

In this work, we have been able to combine some known access control techniques to develop an efficient access control system. To some degree, most access control models are not flexible; they either permit access or deny a subject completely. CACA considers key access issues in granting a level of access to subjects. It has also eliminated the problem of complexity in rules specification and overall administration of access control systems.

Access control research still requires a lot of effort despite the previous activities in this area. More research effort is still needed to achieve great success in designing access control systems.

References

Cavale, M., & McPherson, D. (2003). *Cu, Role-based access control using Windows Server 2003 Authorisation Manager*. Microsoft Corporation. Retrieved from http://www.microsoft.com/technet

Ferraiolo, D., Kuhn, D., & Chandramouli, R. (2003). *Role-based access control*. Artech House, Computer Security Series.

Hu, V. C., Ferraiolo, D. F., & Kuhn, D. R (2006). *Assessment of access control systems*. NIST Interagency Report 7316.

Manjhi, A., Ailamaki, A., Maggs, B. M., Mowry, T. C., Olston, C., & Tomasic, A. (2006). Simultaneous scalability and security for data-intensive web applications. *Proceedings of ACM SIGMOD*, June.

McCue, A. (2000). *LloydsTSB to offer smartcard security*. Retrieved from http://www.vnunet.com/vnunet/news/2118641/lloydstsb-offer-smartcard-security

Menzel, M., Wolter, C. and Meinel, C. (2007). Access control for cross-organisational web service. *Journal of Information Assurance and Security, 2*.

Naumovich, G., & Centonze, P. (2006). *Static analysis of role-based access control in J2EE applications*. Department of Computer and Information Science, Polytechnic University, Brooklyn, NY.

Obedkov, S., Kourie, D. G., & Eloff, J. H. P. (2009). Building access control models with attribute exploration. *Elsevier Journal of Computers and Security, 28*, 2-7.

Pfleeger, P., & Pfleeger S. T. (2003). *Security in computing*. Prentice Hall.

Scott-Chapman, A. (2006). *A dynamic, perimeter based, community-centric access control system*. Msc. Thesis, Florida State University.

Yang, N., Barringer, H., & Zhang, N. (2008). A purpose-based access control model. *Journal of Information Assurance and Security, 1*.

Biographies

Dr. A. S. Sodiya is a Senior Lecturer in the Department of Computer Science, University of Agriculture, Abeokuta, Nigeria. His research interests are computer security, artificial intelligence, network management, and information systems. He has publications in both local and international journals.

S. A. Onashoga is currently a Lecturer in the Department of Computer Science, University of Agriculture, Abeokuta, Nigeria. Her areas of specialization are data mining, computer security, and information systems. She has publications in both local and international journals