Use Case Based Access Control[1]

By
Coimbatore S. Chandersekaran
William R. Simpson
Institute for Defense Analyses, 4850 Mark Center Dr.
Alexandria, Virginia 22311

[1] The publication of this paper does not indicate endorsement by the Department of Defense (DOD) or IDA, nor should the contents be construed as reflecting the official position of these organizations.

ABSTRACT

This paper describes a use case based access control architecture developed by the authors that is extensible and provides a systematic approach to access control within the Air Force enterprise, DOD interest groups and coalition partners. The architecture leverages COTS products that separate the administration of access control from its use of access to data and resources. A prototype was implemented using the enclave model that allows for extensibility. The results from a pilot implementation support the use of Use Case Based Access Control to facilitate security administration and review for the Air Force. The use case based approach provides the sophistication of the attribute based access control with the simplicity of the group based access control.

Keywords: Access Control, enterprise, information security, information sharing.

Introduction

The development of access control within the Air Force is being approached from the ground up, starting at the functional development, the computing environment, and then enterprise. If the access control process is orderly, sufficiently simple, and straightforward, it can be extended to cross-enterprise, coalition and dynamic interest groups. If the access control process is complex and/or violates IA tenets then the result will be expensive, high maintenance, and an inconsistent application of policy and agreement with unsatisfactory results and high maintenance costs.

Operational Tenets

Any access control solution for the enterprise--and indeed, any solution for any component of enterprise IA--should be tested against a set of fundamental evaluation criteria or tenets. These tenets are separate from the "functional requirements" of a specific component, e.g., access control needs to be defined. They relate more to the attributes of the solution that make it able to be implemented, extensible, cost-effective, and supportive of the fundamental objectives of IA. Our proposed top-level tenets are the following:

- The **zeroth** tenet is that the *enemy is embedded*. Current threat evaluation indicates that at the unclassified and NIPR level, attacks are often successful, and discovery and ferreting out the results of these attacks is difficult and problematic at best. In many cases attackers may get inside of the exploit discovery and patch loop. In others, successful Phishing and Spear Phishing attacks have been launched. Rogue agents may be present and to the extent possible, we should be able to operate in their presence, although not exclude their ability to view some activity. The tenets below together with the architecture embody this approach.

- The **first** tenet is *simplicity*. This seems obvious, but it is notable how often this principle is ignored in the quest to design solutions with more and more features. However, at a certain point (usually a lower point than you would suspect), these added features come at the cost of greater complexity, less understandability, greater difficulty in administration, higher cost, and/or lower adoption rates that are unacceptable to the organization. Therefore, simplicity absolutely must be a primary goal of any access solution. Supporting cross-enclave and enterprise scenarios will automatically add a certain degree of complexity that will be challenging enough to handle in any case. Extension to coalition adds yet another level of complexity. That being said, there is a level of complexity that must be handled for security purposes and implementations should not overly simplify the problem for simplicity's sake.

- The **second** tenet, and closely related to the first is *extensibility*. Any construct we put in place for an enclave should be extensible to the forest and the enterprise, and ultimately to cross-enterprise and coalition. It is undesirable to work a point solution or custom approach for any of these levels.

- The **third** tenet is *information hiding*. Essentially, information hiding involves only revealing the minimum set of information to the outside world needed for making effective, authorized use of a capability. It also involves implementation and process hiding so that this information cannot be farmed for information or used for mischief. For example, a user of a service needs to know the input parameters required to call it, and the output it gets in return. It does not need to know the algorithms or internal variables the service uses to implement the capability. Hiding this information keeps these details secret from the consumer of the capability, makes it harder to exploit and increases implementation flexibility. Any information that is not shielded from inadvertent discovery may be used in later attacks.

- The **fourth** tenet is *accountability*. In this context, accountability means being able to definitively identify and track what entity in the enterprise performed any particular operation (e.g. accessed a file or IP address, invoked a service). To enable accountability, it is necessary to prohibit online "impersonation", in which principals share their credentials with another actor rather than delegating their authority. Without a delegation model, it is impossible to establish a chain of custody or do effective forensic analysis to investigate security incidents.

- This **fifth** tenet is *minimal detail* (to only add detail to the solution to the required level). This combines the principles of simplicity and information hiding, and preserves flexibility of implementation at lower levels. For example, adding too much detail to the access solution while all of the other components are still being elaborated may result in wasted work when the solution has to be adapted or retrofitted later.
- The **sixth** is the emphasis on a *service-driven* rather than a product-driven solution whenever possible. Using services makes possible the flexibility, modularity, and composition of more powerful capabilities. Product-driven solutions tend to be more closely tied to specific vendors and proprietary products. That said, COTS products that are as open as possible will be emphasized and should produce cost efficiencies.
- The **seventh** tenet is that *lines of authority* should be preserved and decisions should be made by policy and/or agreement at the appropriate level.
- The **eighth** and final tenet is the *Need to Share* outweighs the need to know.

Access Control

Access control commonly involves two areas, the physical and the logical. When dealing with the physical access control, an individual presents himself and his mission (reason for being there) at a facility and to a guard or clearance individual, together with his credentials that identifies him as an individual. These may be supplemented by a data base of attributes that support his identity and his mission. Identity-based access control in these cases is perfectly reasonable. A human interpretation based upon experience, the individual facility and its missions provide a clear basis for decision making. Restricted information policies can be over-ridden or enforced by the nature of the problem (for example, too close an inspection of covert operatives is not desirable – supporting the tenet of *information hiding*). In this case, time is a commodity that can be used to elaborate upon decisions and suspect cases can be examined in greater detail.

This is not the case referred to by this white paper. The logical access control occurs when a documented (certificated) individual presents himself to a service and the service needs information upon which to base a decision of providing service or not (and if so, which kind of service). This decision often involves transfer of information and that transfer may be in quantity. If the service examines only an attribute set, it may not be sufficient for a decision or the decision may be a complex Boolean algebra that is subject to complexity and error and based upon GOTS developed processes rather than COTS process (this does not support the tenet of *Simplicity*). The process invites wholesale grabbing of attributes (which does not support the tenet of *information hiding*).

Authorization and Access in General

Access is granted through an authorization process that is generally reviewed through the appropriate level of authority for the problem being analyzed. Thus, local access to local data bases may be resolved at the local level. Air Force wide access to Air Force data will be resolved at a much higher level. Coalition access may be resolved at the State Department, or at least Department of Defense level. The authorization model in general requires the user, either directly or through an authenticated source, to supply credentials and authorization information sufficient for the service to make an access decision. Many models exist and they include:

- **Default access** (anyone pre-registered can have access) – this lacks the discrimination and flexibility necessary for an effective IA policy. Revocation is done by removing pre-registration. Delegation difficult in this model (basic tenet of *accountability*).

- **Identity based access (IBAC)** – Access is based solely upon identity, and is usually accompanied by a weak authentication such as password only. The identity has some valuable properties and should be included in many of the other access control methods.

- **Group based access (GBAC)** – all members for a specific access privilege (under a registered group) are enumerated. Individual revocation is done by removing group link to identity. Group revocation is done by removing the group from the registration authority. All privileges for an individual must be accomplished by an enumeration service, and is given by the enumeration of the privilege groups held by an individual. Delegation and/or inheritance are both possible.

- **Role based access (RBAC)** – a special form of the group where privileges are accrued by virtue of job requirements. Revocation is done by removing role from the individual entity (in reality removing attributes that make up the role). Difficulties come when individuals assume multiple roles (often solved by assigning separate identities). When treated as group access then membership in multiple groups is not a problem. Delegation and/or inheritance are both possible. Roles may be credentialed and the use of role credentials may be used in place of individual credentials. Roles are less dynamic than groups in general, but do change over time. Roles may exist independent of membership, but in any instance of time the collection of members serving a role may be treated as a group.

- **Attribute based access (ABAC)** – a form of access control that is based on enumerated attributes rather than identity. Revocation is done by modifying a particular attribute or creating an exception list. All members who have a particular access privilege must be enumerated through a separate service. A separate use case for each type of access must be developed. Enumeration of all privileges for an individual is very problematic. Delegation and/or inheritance are both difficult and problematic.

- **Policy Based (PBAC)** – privilege is based upon policy and is described by the context, currency, and priorities of the command structure together with the other attributes that the entity may have. One group of policies may be deemed attribute control, others

may be context driven. The policy based privilege is the most difficult to automate because the semantics and grammar of an arbitrary policy based authorization has not been defined.

- Others – basically combinations of the above.

- **Use case based access control (UBAC)** – described in this paper.

COTS in general is proficient at handling group based (GBAC) and role based access (RBAC) along with Identity, so that products exist that can use these features in a general access approach. This is not true for attribute based or policy based access control both of which require specialized rule sets.

Establishing Authorization

Authorization for access is developed for software services from the relevant use case. One or more Communities of Interest (COIs) are tasked with establishing the use case. The use case can then map attributes to a group with prospective membership (this is the same process provided by ABAC, but it is done statically, not at service access time). This use case may be internal to an enclave or derived from a Trust agreement[2] that is either federated across enclaves within the enterprise, enterprise-wide, cross-enterprise (as with a DoD/AF Enterprise trust agreement), or across a coalition. Steps required for authorization are the following:

1. Put in place the controls developed by the trust agreement, which may consist of audit requirements, scope limitations, and other factors.

2. COI establishes use case(s) for access to a given service. The use case may be simple or arbitrarily complex, including white list and black list elements as well as exception conditions.

3. The Use Case is moved up the chain to the appropriate command level for approval and details of the use case may be negotiated prior to approval (this maintains *lines of authority*).

4. COI establishes a group name and registers it with the enterprise server.

5. A policy by the appropriate command level is promulgated.

6. Commands and/or COIs establish group membership in accordance with the use case and add group privileges to the Active Directory (AD) or LDAP of the members it controls. The latter may be done by script or robot or by using ABAC for determination.

   a. If a trust agreement is the basis of the access arrangement, other members of the trust agreement establish group membership and add group privileges to the members it controls. A mapping of identity and/or groups may be required.

   b. The trust agreement, the group names and membership are forwarded to the appropriate policy organization for approval. (This may be SECAF for Air Force enterprise issues, DoD or JCS for cross-enterprise or coalition issues).

7. Service administrators respond to the policy by adding the appropriate groups to their access lists for the services.

8. At execution time, access groups are verified through the enterprise server to be sure that the access group has not been revoked. This allows termination of a Trust agreement with only an action at the Enterprise level; individual terminations occur through the modification of group access privileges.

UBAC as a Group Access Model

If we allow that role based access may be included in the group model (this can be done as a minimum by devising a group for each role), then we will restrict this discussion to groups. In the group representation, access is controlled to services through group identity associated with an individual. Standing groups could be established with certain attribute sets, such as "all O-3 and above with personnel supervisory responsibility shall have access to personnel files". As you can see, ***the use cases are built right into the group definitions***. Thus the process supports the *Simplicity* tenet. Ignoring the start-up issues of establishing an initial set of groups, and allocating individuals group status (this start-up exists in all of our access model approaches), an individual is assigned to a group by his/her supervisor, a joint access control group, or by top-level policy. Thus the process supports the *lines of authority* tenet. The requirements for group membership may include the attributes in the ABAC model, but additional data may also be pre-requisite for group membership, such as temporary assignment, or other parameters. The group membership hides the details of its establishment, which may be pulled out by a separate service, but with appropriate authorization. Thus the process supports the *Information Hiding* tenet.

UBAC Group access is by policy. At each element in the information sharing hierarchy, a policy is established which is the access agreement between players. In the enclave, this policy may be established locally, for the forest this policy must be established at a higher level, in the enterprise it must be established Air Force wide. In coalition it must be established between coalition partners, etc. A new group is formed to tackle a problem and requires access to a variety of information elements either within the enclave, forest, enterprise, or across a coalition. The group must be established, named, registered (so that

[2] **International Multi-Conference on Engineering and Technological Innovation**, *"Information Sharing and Federation",* July 2009.

everyone recognizes it) and then it must be pushed through the policy chain to establish an authorization. Thus the process supports the ***lines of authority*** tenet.

Once policy is established, an individual requested service can then be checked against the policy list (through registration services) and granted access based on group membership and policy. Thus this part of the process also supports the ***Simplicity*** tenet. Administration of which individuals have group identities is an ongoing process, much as maintaining attributes is an ongoing process. Termination of a coalition or other agreement is as simple as removing the group identity, and the groups, by policy may be filtered for situational parameters such as DEFCON or Threat Condition. Thus this part of the process also supports the ***Simplicity*** tenet. Group identities must be unique, thus a registration process is recommended. In the original example:

> *Access to the NATO salary data base will to be granted to US Military Personnel holding the rank of O-3 and above, which are not in foreign liaison positions with non-NATO countries nor otherwise excluded because of prior history.*

The individuals that qualify are assigned group G_x privileges. The judgment model proceeds along a process that goes something like this:

> *IF (individual has group credential G_x) then provide service (to the appropriate level).*

This algebra (group membership), will also be worked out by agreement between coalition partners and or cross-enterprise dignitaries (groups are designated by trusted agents, and reviewed for policy by the appropriate policy personnel) and is available in COTS software, many of which work with groups and roles, but not attributes. Thus this aspect of the process supports both the ***Service-Driven*** and ***lines of authority*** tenets. Revocation of an individual's access is done by removal from the group, and elimination of the access is by elimination of the group. Entire groups may be suspended by revoking access to services based upon that group membership.

Authorization within the Enterprise

For now we will concentrate on a particular attribute of groups. If the attribute were groups (identity and roles), and the groups were unique and known to all of the members of the enterprise, then authorization could be accomplished. One way to insure that all groups were unique, and that all groups were known to the enterprise, is through a registration service that belongs to the enterprise. As groups are formed, their names are registered and promulgated by policy. Individuals would have their group credentials stored in the Active Directory, LDAP UDDI, or other data stores as appropriate. Enterprise policy could push the privileges associated with each group so that administrators of services could place the authorizations in their applications as required by the policy.

One way to insure that all groups were unique, and that all groups were known to the enterprise, is through a registration service that belongs to the enterprise. If authentication were made via standard web services, the group membership could be placed in SAML token and the group content could be verified against a federation service where the groups are registered.

Finally, we note that the group model is extensible to the cross-enterprise and coalition situation where trust agreements negotiate not only privilege and group membership, but the group access tags that are registered at each of the enterprise levels. In fact, since the an STS is in place, the simplest way to administer the access control is to use the web services and SAML tokens even at the cross-enclave and intra-enclave level. This will provide a truly extensible method that provides access controls for cases from intra-enclave to coalition.

Naming of Groups for Access

With naming services, the default names that are hardwired should not be used (reserved word list may be appropriate) and these should be used for clearing up problems when files are corrupted, etc. The naming of groups for access will be the job of the Community of Interest (COI) for the enterprise, however we need to ensure common elements where required by history, policy or other Air Force and DoD requirements. An example would be the names for classification groups (Unclassified, Confidential, Secret, Top Secret…). We need also to ensure uniqueness and hierarchy. The following requirements apply:

- The IA COI will be responsible for naming and registering enterprise wide group names and hierarchies.
- The Enclave COIs will be responsible for naming and registering groups that deal with their services with the proviso that they reuse groups were possible, and place groups in a hierarchy when possible.
- Even groups that are wholly for use within the enclave need to be registered to assure uniqueness and awareness.

Basic Elements of GBAC

On the requestor side, every entity has one or more unique names (and may have aliases). Every entity is enrolled in zero or more groups. On the provider side, for every object a service can touch, service will be authorized by name and/or zero or more group affiliations (as promulgated by policy).

Summary

The architecture requires that a registration service for groups and group membership be established at the enterprise level. It requires that administrators periodically review group memberships that they have been deemed, by policy, to provide services to, or alternatively, they could write scripts or programs (or the vendors could provide services) to do this function. The rest is inherent in the enterprise architectures for most of the enterprise using Active Directory, and a few services away from other representations including LDAP and flat files. This

architecture is also extensible to cross-enterprise, coalition, and other authorization processes.

ACRONYM PRIMER

ABAC - Attribute Based Access Control
COI - Community of Interest
COTS - Commercial Off-The-Shelf
DOD - Department of Defense
GBAC - Group Based Access Control
IA - Information Assurance
IBAC - Identity Based Access Control
IDA - Institute for Defense Analyses
IP - Internet Protocol
LDAP - Lightweight Directory Access Protocol
NIPR - Non-secure Internet Protocol Router
PBAC - Policy Based Access Control
RBAC - Role Based Access Control
UBAC - Use Case Based Access Control
UDDI - Universal Description, Discovery and Integration

REFERENCES

[1]. *National Defense Strategy of the United States of America*, March 2005.

[2]. The *"Net-Centric Operations and Warfare Reference Model, OASD (NII)",* 17 Nov 2005.

[3]. *"Joint Concept of Operations for Global Information Grid NETOPS version 2.0",* USSTRATCOM, 10 Aug 2005

[4]. Liu, Ranganathan, Riabov, "*Specifying and Enforcing High-Level Semantic Obligation Policies*," policy, pp. 119-128, Eighth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'07), 2007

[5]. Vijayalakshmi Atluri , Avigdor Gal, "*An Authorization Model For Temporal And Derived Data: Securing Information Portals*", ACM Transactions on Information and System Security (TISSEC), v.5 n.1, p.62-94, February 2002

[6]. Jean Bacon , Ken Moody , Walt Yao, "*A Model Of OASIS Role-Based Access Control And Its Support For Active Security*", ACM Transactions on Information and System Security (TISSEC), v.5 n.4, p.492-540, November 2002

[7]. Ronald Fagin, "*On An Authorization Mechanism, ACM Transactions On Database Systems (TODS)*", v.3 n.3, p.310-319, Sept. 1978

[8]. Cheh Goh , Adrian Baldwin, "*Towards A More Complete Model Of Role*", Proceedings of the third ACM workshop on Role-based access control, p.55-62, October 22-23, 1998, Fairfax, Virginia, United States

[9]. Patricia P. Griffiths , Bradford W. Wade, "*An Authorization Mechanism For A Relational Database System*", ACM Transactions on Database Systems (TODS), v.1 n.3, p.242-255, Sept. 1976

[10]. Åsa Hagström , Sushil Jajodia , Francesco Parisi-Presicce , Duminda Wijesekera, "*Revocations-A Classification*", Proceedings of the 14th IEEE Workshop on Computer Security Foundations, p.44, June 11-13, 2001

[11]. JongSoon Park, YoungLok Lee, HyungHyo Lee, and BongNam Noh, "*A Role-Based Delegation Model Using Role Hierarchy Supporting Restricted Permission Inheritance*", In Proceedings of the International Conference on Security and Management, SAM '03, pages 294--302. CSREA Press, 2003.

[12]. Jacques Wainer, Paulo Barthelmess, and Akhil Kumar. "*WRBAC - A Workflow Security Model Incorporating Controlled Overriding Of Constraints*", International Journal of Cooperative Information Systems, 12(4):455--486, 2003.

[13]. Walt Yao. Fidelis, "*A Policy-Driven Trust Management Framework. In Trust Management*", First International Conference, iTrust, volume 2692 of Lecture Notes in Computer Science, pages 301--317. Springer, 2003.

[14]. Longhua Zhang , Gail-Joon Ahn , Bei-Tseng Chu, "*A Rule-Based Framework For Role-Based Delegation And Revocation*", ACM Transactions on Information and System Security (TISSEC), v.6 n.3, p.404-441, August 2003

[15]. McGraw, Gary & Viega, John, "*Keep It Simple*", Software Development, CMP Media LLC, May, 2003.

[16]. NIST, "*Engineering Principles for Information Technology Security*", Special Publication 800-27, US Department of Commerce, National Institute of Standards and Technology, 2001.

[17]. Saltzer, Jerome H. & Schroeder, Michael D., "*The Protection of Information in Computer Systems*," 1278-1308, Proceedings *of the IEEE 63*, 9 (September 1975).

[18]. Schneier, Bruce. "The Process of Security," *Information Security Magazine*, April, 2000.

[19]. Viega, John & McGraw, Gary. "*Building Secure Software: How to Avoid Security Problems the Right Way",* Boston, MA: Addison-Wesley, 2002.

[20]. Andrew Trice and Chandersekaran Coimbatore, "Classification, Labeling, and Relationship to Crypto Binding", August 20, 2007

[21]. "Guide to Secure Web Services: Recommendations of the National Institute of Standards and Technology", NIST-US Department of Commerce Publication, August 2007.

[22]. *Trusted Computer System Evaluation Criteria (TCSEC).* United States Department of Defense. December 1985. DoD Standard 5200.28-STD

[23]. Ferraiolo, D.F. and Kuhn, D.R. (October 1992). "*Role Based Access Control*". *15th National Computer Security Conference*: 554-563.

[24]. Sandhu, R., Coyne, E.J., Feinstein, H.L. and Youman, C.E. (August 1996). "*Role-Based Access Control Models*". *IEEE Computer* (IEEE Press) **29** (2): 38–47. http://csrc.nist.gov/rbac/sandhu96.pdf.

[25]. Sandhu, R., Ferraiolo, D.F. and Kuhn, D.R. (July 2000). "*The NIST Model for Role Based Access Control: Toward a Unified Standard*". *5th ACM Workshop Role-Based Access Control*: 47-63.

[26]. Sylvia Osborn, Ravi Sandhu, and Qamar Munawer (2000). "*Configuring role-based access control to enforce mandatory and discretionary access control policies*". *ACM Transactions* on Information and System Security (TISSEC)*: 85-106.*

[27]. Priebe, T., Fernandez, E.B., Mehlau, J.I., and Pernul, G. "*A pattern System for Access Control*" Proc. 18th Annual IFIP WG 11.3 Working Conference on Data and Application Security, Stiges, Spain, July 2004, p. 235 -249.

[28]. William R. Simpson and Combinatore Chandersekaran, International Multi-Conference on Engineering and Technological Innovation, *"Information Sharing and Federation",* July 2009.

[29]. Priebe T, Dobmeier, W. Muschall B. and Pernul G. "*ABAC – Ein Referenzmodell Fur attributbasierte Zugriffskontrolle*," Proc. 2. Jarrestagung Fachbereich Sicherheit der Gessellschaft Fur Informatik (Sicherheit 2005) Regensburg, Germany, April 2005, pp. 285-296.

[30]. Yuan, E., and Tong J. *"Attribute Based Access Control (ABAC) for Web Services"* Proc. 3rd International Conference on Web Services (ICES 2005), Orlando USA, July 2005, pp. 561-569.

[31]. Mary Thompson, William Johnston, Srilekha Mudumbai, Gary Hoo, Keith Jackson, Abdelilah Essiari, *"Certificate-based Access Control for Widely Distributed Resources"*, Proceedings of the 8th USENIX Security Symposium, Washington, D.C., USA, August 23–26, 1999