Differentiating Between Access Control Terms

Understanding User and Role Based Access Control, Policy Based Access Control, Content Dependent Access Control, Context Based Access Control, View Based Access Control, Discretionary and Mandatory Access Control

Table of Contents

1. INTRODUCTION

Access control, by the broadest definition, is the ultimate goal of all network security – granting access when appropriate and denying when inappropriate. Access control tools help accomplish this purpose, as do firewalls, encryption, and intrusion detection. In light of the true mission of network security, however, having the right access control tool is absolutely essential.

Almost every network has some form of access control, even if it is merely that of the native operating systems – some networks have several access control tools that protect different resources. The various types of access control tools enforce security policy and/or users' privileges by protecting mail servers, web applications, database systems, fileservers, applications, or some combination of these resources. In fact, there are so many types of access control tools and so many non-standardized terms to describe them that it is difficult to determine which solution is right for a given network. An understanding of the more common usages of these terms can make it easier to discern between the basic types of solutions available.

Whatever the access control solution, there are only a few approaches to configuring access permissions, regardless of the terms different companies use to represent those approaches. These approaches strive to achieve the common goals of fineness of granularity and management simplification, usually sacrificing one goal to some extent, in order to meet the other.

The finer the granularity, the less likely any user is to be granted unnecessary access – or to be denied necessary access. Likewise, the coarser the granularity, the more likely unnecessary privileges will be granted. The supreme goal of refining access privileges is to attain the level of least privilege - something generally thought in security to be unattainable. Least privilege is the concept of defining user privileges to precisely what users need, with no extraneous privileges. The need to achieve this level of security for a variety of resources (fileservers, applications, and web applications) has been the driving force in developing new approaches to access control.

Ultimately manual configuration of permissions schemes is the stumbling block that prevents the various approaches from achieving the common goals of access control. It is unlikely that any solution requiring manual configuration will be able to refine access privileges to the level of least privilege or to substantially minimize management overhead. If informed executives can easily distinguish between the basic types of solutions available, they will be better equipped to make the right decisions in purchasing and implementing the optimal technological solutions.

2. User Based Access Control (UBAC)

UBAC, which has also been called identity-based access control, requires a system administrator to define permissions for each user, based on the individual's needs. UBAC has the potential to result in more finely grained permissions, although an effective UBAC system would be so labor intensive as to be cost-prohibitive. It is impossible for security management to know precisely what access each and every user needs, accordingly configure permissions, and update them daily to avoid build-up of outdated permissions.

For example, an assistant to the CEO may need specific permissions that an assistant in the marketing department doesn't need. Rather than assigning all assistants to a group that can access the same resources, the CEO's assistant is assigned permissions to the CEO's files, and other resources that he/she will need on a day-to-day basis; all assistants are defined individually and not grouped together.

In actual practice, UBAC has largely been implemented by designating extremely coarse user groups, giving each user far more access privileges than he/she could possibly need.

3. Role Based Access Control (RBAC)

Role Based Access Control (RBAC) is popular because it purports to advance permissions configurations towards the common goals. RBAC entails mapping the different "roles" (a "role" is a user group with access to a specific group of resources) in an organizational hierarchy and defining a profile of access permissions to the network's resources for each role. Then each user is assigned one or more roles, providing him/her with the access permissions defined by those roles. A user with super-user access may be classified in every role, whereas someone with less need-to-know may be classified in only one or two roles.

RBAC has the potential for refining granularity by assigning specific types of privileges to specific resources. Users in a certain "role" may be granted access to "read" but not to "write" certain files. However, with a finite number of pre-defined roles, "we do not expect to have many read-only or write-only roles. Yet it is really only within the read-only roles that we get much differentiation in security levels."[1]

Pre-defined roles are overly broad. For example, a secretary in marketing may need "write" privileges for marketing documents, but permissions assigned to the secretarial "role" may be "read only." As there are exceptions like this in every role and security clearances differ from department to department even in the same "role" or position, new "roles" must be constantly defined in the system. The more meticulous the manual definition of roles, the finer the granularity; however this multiplies the management effort exponentially.

[1] "Mandatory Access Control and Role-Based Access Control Revisited" Sylvia Osborn p.10

In order to attain a finer degree of granularity than found in most UBAC group permissions schemes, RBAC requires incredible management effort and guesswork.

4. POLICY BASED ACCESS CONTROL

Policy Based Access Control is also known as Rule Set Based Access Control (RSBAC). An access control policy is a set of rules that determine users' access rights to resources within an enterprise network (e.g., files, directories, sites, web pages). A typical example would be a policy regulating employees' access to corporate internal documents via an intranet. A policy, in that case, may impose a limit on the number of documents that can be downloaded by an employee during a certain time period, limit his/her access to certain sites or to certain web pages on the intranet.

One prevailing mechanism for enforcing enterprise policy is the use of access control lists (ACLs). ACLs associate with every resource, lists of users or groups of users and their access rights (i.e., the type of access attempts that should be allowed).

ACLs are ineffective in enforcing policy. When using ACLs to enforce a policy, there is usually no distinction between the policy description and the enforcement mechanism - the policy is essentially defined by the set of ACLs associated with all the resources on the network. Having a policy being implicitly defined by a set of ACLs makes the management of the policy inefficient, error prone, and hardly scalable up to large enterprises with large numbers of employees and resources. In particular, every time an employee leaves a company, or even just changes his/her role within the company, an exhaustive search of all ACLs must be performed on all servers, so that user privileges are modified accordingly. It follows that such policies are usually defined with a very low granularity, and hence tend to be overly permissive.

In contrast, policy-based access control makes a strict distinction between the formal statement of the policy and its enforcement. Making rules explicit, instead of concealing them in ACLs, makes the policy easier to manage and modify. Such a mechanism is usually based on a specification language which should be expressive enough to easily formulate the policy rules.

Coupled with the policy description language there should be an enforcement mechanism capable of intercepting access attempts, evaluating them against the policy, and accordingly granting or denying the access requests.

In all the examples (see Matrix) one can maintain an explicit access control policy, and use the appropriate technology to enforce it. The firewalls example stands out in that the enforcement technology is based on the existence of an explicit table of rules that defines the policy, and hence it is in a sense a strict policy-based access control mechanism.

Policy Based Access Control, alone, like ACLs or the access control of native operating systems, isn't designed with fine granularity or management simplification in mind. However, these approaches can be used in combination with other access control tools.

5. Content Dependent Access Control (CDAC)

Content Dependent Access Control is a method for controlling access of users to resources, based on the content of the resource. CDAC is primarily used to protect databases containing potentially sensitive data. A good example would be a patient record management system that allows different people to access records depending on what they contain. A nurse may have access to blood tests, for example, unless the blood test is an HIV test (the system has to check which test it is in order to determine if the access is allowed). Only certain people can access such a record.

One can argue that anti-virus software is a content-based access control system - as it allows access only to files that do not contain viruses. Resource attributes may also be viewed as part of its content - though usually they are not regarded as part of it. For example, each file in an operating system of the Windows™ family has a "Read Only" attribute. "Write" access to such a file is denied regardless of what the permissions for this file are, if the flag is On. If the attribute is considered to be part of the file, then this would in theory be a content-dependent access control system, but it's not considered as such.

Content Dependent Access Control involves a lot of overhead resulting from the need to scan the resource when access is to be determined (in some implementations it may really slow down the users, even if the security policy doesn't utilize the content-dependent capabilities). High levels of granularity are only achievable with extremely labor-intensive permissions configuration and continuous management.

6. Context Based Access Control (CBAC)

CBAC is most commonly used to protect traffic through firewalls. Since Context and Content sound similar, many people confuse them - but these are actually two completely different approaches to access control, which may be used simultaneously. Context Based Access Control means that the decision whether a user can access a resource doesn't depend solely on who the user is and which resource it is - or even the resource content, as in the case of Content Dependent Access Control - but also in the sequence of events that preceded the access attempt.

For example, a system that doesn't allow a user to access a certain resource more than 100 times a day, is a context based access control system, since it counts the number of accesses performed and blocks everything beyond the first 100 accesses, regardless of the fact that the user is the same user and that the resource is the same resource. As an example, a quota control is a program that allows system administrators to control collective use of corporate fileservers, i.e. users may save up to a certain volume of

data to shared fileservers, but may not exceed a predefined quota. Such a system may be viewed as a context-based access control system, because once a user has reached his/her quota, he/she will not be permitted to save any additional data, regardless of any other policy.
A firewall typically performs "stateful inspection" which means it updates the "state" of every connection when a new packet arrives, and drops packets or allows them to continue, based not only on their content, but also on the current state, the context in which the packets arrive. Checkpoint coined the term “stateful inspection”, an essential element of their patented firewall technology, to contrast with the industry term “stateless”; with stateless inspection every information packet is considered individually out of context.

Configuration of permissions for specific users is not required. Strict security policy rules are set and form the basis of decisions to permit or deny access.

7. View Based Access Control (VBAC)

View Based Access Control primarily protects database systems; thus for files and other applications, VBAC is not a functional solution. As opposed to other notions of access control, which usually relate to tangible objects, like files, directories, printers, etc., VBAC perceives the resource itself as a collection of sub-resources. For example a hospital’s patient record could be considered a resource. Employees in the finance department might need to see sub-resources of a patient record, while nurses administering drugs might need to see physician’s instructions on that record. All users access the same record, but different sub-resources are viewable by different users.

The view-based approach is naturally applicable when enterprise information is maintained in databases. In that case the access control policy could be based on a set of predefined interfaces (views), so that users are allowed to interact with the resources only via these interfaces. This can be used, for instance, to restrict modification rights of the salary field in a payroll table to the chief finance officer only, for salaries above a given threshold.

Granularity can be very fine, but permissions configuration is incredibly labor-intensive. In particular with VBAC, defining the resources is extremely complex. Such configuration requires intimate knowledge of the data structures and mutual relationships between users and data. Although most users only view a portion of any given resource, that resource must still appear complete, within context. The CFO who sees the salary field still may not view other fields, but the document must appear normal and complete, as if there were nothing missing.

8. Mandatory and Discretionary Access Control (MAC and DAC)

One of the attributes by which an access control policy is classified is whether it is a Mandatory Access Control (MAC) policy, or a Discretionary Access Control (DAC) policy.

As the name implies, a MAC policy is obligatory – that is, it dictates whether an operation should be permitted or denied without letting a user override the policy. Most mail servers, for example, enforce a policy that disallows messages larger than a predetermined size to be sent through them. Many mail servers also reject any incoming messages that are suspected of containing a computer virus. Both policies are MAC because they cannot be overridden by a decision of an end-user – neither the sender nor the recipient of a message can ask the mail server to disregard the policy for a specific message. The term "access control" is not in common use with respect to mail servers, but effectively, what the policy controls is access to the mailboxes managed by the mail server.

A DAC policy, on the other hand, leaves final decision in the hands of the end-user. The most common example is a computer file system. The owner of a file can grant or deny access rights at his/her discretion. For example, a company may decide on a policy that prohibits employees from disclosing expense reports to each other, and requires that every manager be able to inspect the expense reports of his/her employees. However, as long as the file system policy is discretionary, so is the adherence to the company policy, and it is up to every employee to adhere to it or violate it by granting or restricting access to his/her files.

An access control policy does not have to be strictly mandatory or strictly discretionary. Let us revisit the mail server example – it is often possible for the mail account owner to specify a rule that rejects messages from a specific origin. Combined with a mandatory size-limit and a mandatory virus-rejection policy, such a mail server employs a combined mandatory/discretionary access control policy – some decisions are mandatory, but others are left at the discretion of the end user. Naturally, for mandatory rules to actually be mandatory, they must take precedence over discretionary rules in a combined mandatory/discretionary policy.

Strict security standards such as the Trusted Computer System Evaluation Criteria (TCSEC) employed in military environments, require MAC policies to be in effect. In the business world, MAC policies are usually not used – not because they are not useful, but rather because they are practically impossible to implement using the operating system's standard tools.

The reader should be aware that both MAC and DAC stand for many other things. DAC also stands for Digital/Analog Converter. MAC is also used as an acronym for Message Authentication Code in secure transport systems, Medium or Media Access Control protocols determine rules for fairly sharing wireless bandwidth. With respect to security policies, however, they stand for Mandatory and Discretionary access control.

DAC can be very finely grained – in theory. In reality, DAC is extremely labor intensive, as each user must define permissions for all users to every resource he/she "owns" – if all users don't cooperate and define permissions to a minimum level of granularity, the system administrator or security officer will have to configure separately for all resource "owners". The fineness of granularity achievable with MAC is dependent on the ability to precisely define least privilege permissions for all users. Any real fineness in granularity requires painstaking configuration.

9. THE BOTTOM LINE

All of the approaches to access control have traditionally required manual configuration, but no manually configured approach has resulted in finely grained permissions, nor in drastically reducing the management workload in assigning permissions. "At present, most access to network information resources is not controlled on a fine-grained basis. There is a very real danger that by accommodating all of the needs for fine-grained access management into the basic access management mechanisms we will produce a system that is too complex and costly to see wide-spread implementation anytime soon,"[2] argues Clifford Lynch in his argument for RBAC. Within the framework of manually configured approaches to access control this is undoubtedly true. The fact that so many extraneous privileges are granted and remain in systems represents a substantial security risk.

Many analysts have predicted a development in automated configuration based on an artificial intelligence as the ideal solution. "Over time, other advances may help put IT managers' minds at ease. One technology on the horizon is a network scanner based on artificial intelligence that will analyze network behavior."[3] This development is a blessing because a clear understanding of actual network usage is necessary for writing logical and functional security policy – yet without intelligent technology, security policy must be based on insubstantial knowledge and guesswork.

Furthermore, such a development would virtually eliminate intrusions and breaches by alerting security management in real-time to suspicious activity on the network, including access attempts by unknown users. Irrelevant of where the access attempt is coming from, inside or outside the network, access attempts that system administrators would not want granted should be blocked and should trigger alerts.

Monitoring and real-time response is necessary to network security. "One way companies can protect themselves from insider abuse is to focus on what their networks can tell them about what is going on inside the company," advised Eric Friedberg, previously the computer/telcos crime coordinator at the U.S. Attorney's Office in New York. Responding a prominent FBI inside hacking incident, Friedberg

[2] "A White Paper on Authentication and Access Management Issues in Cross-organizational Use of Networked Information Resources" by Clifford Lynch, Coalition for Networked Information p.7
[3] "Firewalls Becoming Ineffective, Experts Say" by Karen Schwartz
http://www.planetit.com/techcenters/docs/security-firewalls/news/PIT20001222S0006?printDoc=1

recommended that companies "look into artificial intelligence-enabled security software that can tip administrators off to 'anomalous activity' on the network."[4]

Network Intelligence as applied to access control could produce the most finely grained user-based permissions, whilst simultaneously simplifying management far beyond any manually configured system.

Camelot's Network Intelligence technology utilizes advanced discovery algorithms to continuously monitor and analyze millions of network events, deduce the functional structure of an organization, extract and map the relationships between users and various network resources - and then respond to these events automatically, reducing management overhead and refining response granularity. Camelot's NI engine performs detailed and precision-based tasks which traditional manual methods are unable to achieve.

Camelot is devoted to researching Network Intelligence and applying it to network security. Powered by Camelot's Network Intelligence technology, Hark! is the world's first automated access control solution. Hark! continuously analyzes millions of network events, observing and learning the access patterns of individual network users. Hark! determines precisely which permissions users need, enforces them and updates them automatically. For more information on Camelot and and Hark!, please visit us at http://www.camelot.com.

[4] "FBI spy case highlights insider threat to corporate data" by Dan Verton
http://www.computerworld.com/cwi/story/0,1199,NAV47_STO57889,00.html

In order to help differentiate between access control terms, the matrix below matches the various terms with common settings, with examples. Several of the settings and examples are not applicable in practice; the matrix should be used as an aid for understanding, not as an authoritative definition of access control in the various settings.

Setting	User/Role Based	Policy Based/ ACLs	Content Dependent	Context Based	View Based	Mandatory/ Discretionary
ATM	User based; privileges assigned specifically to account owner.	Policy dictates withdrawal limit per day and per individual account.	Cannot withdraw more than account balance.	Cannot withdraw more than $2500 per 24 hours.	No appropriate example.	Mandatory; account owner cannot grant others right to access his account with their cards.
Magnetic Door Keys	User based; access assigned specifically to key owner.	ACL based because access is assigned through lists of authorized users.	No appropriate example.	Deny access if door opened more than 3 times in an hour, or during nightshift.	No appropriate example.	Mandatory; key owner cannot autonomously delegate his access rights to other key owners.
Firewall	Usually Role based; access granted by role, e.g. employee, subcontractor, customer.	Both policy and ACL based. You set policy in configuring a firewall.	Do not allow emails containing viruses to go through.	Do not allow access to internal network resources until the user has been authenticated.	No appropriate example.	Mandatory; an internal user cannot grant access to data through the firewall.
Hospital Database System	Usually Role based; access granted by role, e.g. Doctor, Nurse, Finance	There could be policy and ACLs, but historically, there has been no enforcement of such rules.	Deny access to anyone except a tending physician to V.I.P. patient records.	No appropriate example.	A finance user can receive a list of patient treatments and costs but not the associated diagnosis or details	Discretionary; a doctor or nurse may grant a colleague access to a patient's file to receive his opinion.
Operating Systems Native Access Control & File Servers	Both Role-based and User-based. Access to data granted either by Rule or specifically to a user.	Access controlled by ACLs; it is impossible to set a policy that would apply to files not yet created.	Before running an application program, run anti-virus check –deny if virus is present.	Users cannot save another file on the shared fileserver once they have exceeded their quota.	No appropriate example.	Discretionary; the file owner can usually grant access to her data to anyone, regardless of company policy.
Browsing a personalized community web site.	User based; personalization and access control are usually user-specific.	Policies can be set, such as webmasters get complete access. Users can be assigned limited ability to modify/ customize pages through ACLs.	Do not allow contributed content to have obscene content.	Only allow full access after the user has gone through an "introductory tour" of the web site.	Content editors get access to complete contribution details, including author history. Other users only get to see the contribution and author's nickname.	Mandatory; a user cannot usually delegate his rights to other users, or bypass the site's policy.
Sending an email	User based; email accounts are created for each individual user.	Impractical for ACLs or user-specific rules. Policies can be set such as: only designated users may send emails to the entire company.	Disallow sending emails longer than 5 Megs.	No appropriate example.	Email author has access to all recipient names, including bcc-line recipients. Other recipients can only see to-line recipients.	Mandatory: neither sender nor recipient can override policy.

10. LEGAL NOTICE

For more information about Camelot, visit http://www.camelot.com.

Camelot Israel
Matam, Advanced
Technology Center
Haifa 31905, Israel
Tel.: +972-4-813-2000
Fax.: +972-4-850-1060

Camelot USA
45 Broadway, 11th Floor
New York, NY 10006
USA
Tel.: +1-212-797-2980
Fax.: +1-212-797-2989