Computer access control

en.wikipedia.org/wiki/Computer_access_control

In computer security, general access control includes identification, authorization, authentication, access approval, and audit. A more narrow definition of access control would cover only access approval, whereby the system makes a decision to grant or reject an access request from an already authenticated subject, based on what the subject is authorized to access. Authentication and access control are often combined into a single operation, so that access is approved based on successful authentication, or based on an anonymous access token. Authentication methods and tokens include passwords, biometric scans, physical keys, electronic keys and devices, hidden paths, social barriers, and monitoring by humans and automated systems.

Software Entities

In any access-control model, the entities that can perform actions on the system are called *subjects*, and the entities representing resources to which access may need to be controlled are called *objects* (see also Access Control Matrix). Subjects and objects should both be considered as software entities, rather than as human users: any human users can only have an effect on the system via the software entities that they control.

Although some systems equate subjects with *user IDs*, so that all processes started by a user by default have the same authority, this level of control is not fine-grained enough to satisfy the principle of least privilege, and arguably is responsible for the prevalence of malware in such systems (see computer insecurity).

In some models, for example the object-capability model, any software entity can potentially act as both subject and object.

As of 2014, access-control models tend to fall into one of two classes: those based on capabilities and those based on access control lists (ACLs).

- In a capability-based model, holding an unforge-able reference or *capability* to an object provides access to the object (roughly analogous to how possession of one's house key grants one access to one's house); access is conveyed to another party by transmitting such a capability over a secure channel
- In an ACL-based model, a subject's access to an object depends on whether its identity appears on a list associated with the object (roughly analogous to how a bouncer at a private party would check an ID to see if a name appears on the guest list); access is conveyed by editing the list. (Different ACL systems have a variety of different conventions regarding who or what is responsible for editing the list and how it is edited.)

Both capability-based and ACL-based models have mechanisms to allow access rights to be granted to all members of a *group* of subjects (often the group is itself modeled as a subject).

Services

Access control systems provide the essential services of *authorization*, *identification and authentication* (*I&A*), *access approval*, and *accountability* where:

- authorization specifies what a subject can do
- identification and authentication ensure that only legitimate subjects can log on to a system
- access approval grants access during operations, by association of users with the resources that they are allowed to access, based on the authorization policy
- accountability identifies what a subject (or all subjects associated with a user) did

Authorization

Authorization involves the act of defining access-rights for subjects. An authorization policy specifies the operations that subjects are allowed to execute within a system.

Most modern operating systems implement authorization policies as formal sets of permissions that are variations or extensions of three basic types of access:

- Read (R): The subject can
  - Read file contents
  - List directory contents
- Write (W): The subject can change the contents of a file or directory with the following tasks:
  - Add
  - Update
  - Delete
  - Rename
- Execute (X): If the file is a program, the subject can cause the program to be run. (In Unix-style systems, the "execute" permission doubles as a "traverse directory" permission when granted for a directory.)

These rights and permissions are implemented differently in systems based on *discretionary access control* (*DAC*) and *mandatory access control* (*MAC*).

Identification and Authentication (I&A)

Main article: Authentication

Identification and Authentication[1] (I&A) is the process of verifying that an identity is bound to the entity that makes an assertion or claim of identity. The I&A process assumes that there was an initial validation of the identity, commonly called identity proofing. Various methods of identity proofing are available, ranging from in-person validation using government issued identification, to anonymous methods that allow the claimant to remain anonymous, but known to the system if they return. The method used for identity proofing and validation should provide an assurance level commensurate with the intended use of the identity within the system. Subsequently, the entity asserts an identity together with an authenticator as a means for validation. The only requirements for the identifier is that it must be unique within its security domain.

Authenticators are commonly based on at least one of the following four factors:

- *Something you know*, such as a password or a personal identification number (PIN). This assumes that only the owner of the account knows the password or PIN needed to access the account.
- *Something you have*, such as a smart card or security token. This assumes that only the owner of the account has the necessary smart card or token needed to unlock the account.
- *Something you are*, such as fingerprint, voice, retina, or iris characteristics.
- *Where you are*, for example inside or outside a company firewall, or proximity of login location to a personal GPS device.

Access approval

Access approval is the function that actually grants or rejects access during operations.[2]

During access approval, the system compares the formal representation of the authorization policy with the access request, to determine whether the request shall be granted or rejected. Moreover, the access evaluation can be done online/ongoing.[3]

Accountability

Accountability uses such system components as *audit trails* (records) and *logs,* to associate a subject with its actions. The information recorded should be sufficient to map the subject to a controlling user. Audit trails and logs are important for

- Detecting security violations
- Re-creating security incidents

If no one is regularly reviewing your logs and they are not maintained in a secure and consistent manner, they may not be admissible as evidence.

Many systems can generate automated reports, based on certain predefined criteria or thresholds, known as clipping levels*. For example, a clipping level may be set to generate a report for the following:*

- More than three failed logon attempts in a given period
- Any attempt to use a disabled user account

These reports help a system administrator or security administrator to more easily identify possible break-in attempts.

Definition of clipping level:[4] a disk's ability to maintain its magnetic properties and hold its content. A high-quality level range is 65–70%; low quality is below 55%.

Access control models

Access control models are sometimes categorized as either discretionary or non-discretionary. The three most widely recognized models are Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role Based Access Control (RBAC). MAC is non-discretionary.

Discretionary access control

Discretionary access control (DAC) is a policy determined by the owner of an object. The owner decides who is allowed to access the object, and what privileges they have.

Two important concepts in DAC are

- File and data ownership: Every object in the system has an *owner*. In most DAC systems, each object's initial owner is the subject that caused it to be created. The access policy for an object is determined by its owner.
- Access rights and permissions: These are the controls that an owner can assign to other subjects for specific resources.

Access controls may be discretionary in ACL-based or capability-based access control systems. (In capability-based systems, there is usually no explicit concept of 'owner', but the creator of an object has a similar degree of control over its access policy.)

Mandatory access control

Mandatory access control refers to allowing access to a resource if and only if rules exist that allow a given user to access the resource. It is difficult to manage, but its use is usually justified when used to protect highly sensitive information. Examples include certain government and military information. Management is often simplified (over what can be required) if the information can be protected using hierarchical access control, or by implementing sensitivity labels. What makes the method "mandatory" is the use of either rules or sensitivity labels.

- Sensitivity labels: In such a system subjects and objects must have labels assigned to them. A subject's sensitivity label specifies its level of trust. An object's sensitivity label specifies the level of trust required for access. In order to access a given object, the subject must have a sensitivity level equal to or higher than the requested object.
- Data import and export: Controlling the import of information from other systems and export to other systems (including printers) is a critical function of these systems, which must ensure that sensitivity labels are properly maintained and implemented so that sensitive information is appropriately protected at all times.

Two methods are commonly used for applying mandatory access control:

- Rule-based (or label-based) access control: This type of control further defines specific conditions for access to a requested object. A Mandatory Access Control system implements a simple form of rule-based access control to determine whether access should be granted or denied by matching:
  - An object's sensitivity label
  - A subject's sensitivity label
- Lattice-based access control: These can be used for complex access control decisions involving multiple objects and/or subjects. A lattice model is a mathematical structure that defines greatest lower-bound and least upper-bound values for a pair of elements, such as a subject and an object.

Few systems implement MAC; XTS-400 and SELinux are examples of systems that do.

Role-based access control

Role-based access control (RBAC) is an access policy determined by the system, not by the owner. RBAC is used in commercial applications and also in military systems, where multi-level security requirements may also exist. RBAC differs from DAC in that DAC allows users to control access to their resources, while in RBAC, access is controlled at the system level, outside of the user's control. Although RBAC is non-discretionary, it can be distinguished from MAC primarily in the way permissions are handled. MAC controls read and write permissions based on a user's clearance level and additional labels. RBAC controls collections of permissions that may include complex operations such as an e-commerce transaction, or may be as simple as read or write. A role in RBAC can be viewed as a set of permissions.

Three primary rules are defined for RBAC:

1. Role assignment: A subject can execute a transaction only if the subject has selected or been assigned a suitable role.
2. Role authorization: A subject's active role must be authorized for the subject. With rule 1 above, this rule ensures that users can take on only roles for which they are authorized.
3. Transaction authorization: A subject can execute a transaction only if the transaction is authorized for the subject's active role. With rules 1 and 2, this rule ensures that users can execute only transactions for which they are authorized.

Additional constraints may be applied as well, and roles can be combined in a hierarchy where higher-level roles subsume permissions owned by lower-level sub-roles.

Most IT vendors offer RBAC in one or more products.

Intent-based Access Control (IBAC)

Intent-based Access Control (IBAC),[5] [6] a novel access control model first proposed by Abdulaziz Almehmadi, is an access control system that detects the intention of the user requesting access answering the question "Why?" access is being requested as opposed to current access control systems that asks "Who?" is requesting

access. IBAC is designed to prevent the insider threat as opposed to the current access control systems that are designed to prevent the outsider threat. IBAC is a risk-based access control that assesses risk of access based on the detected intent and the motivation level towards executing that intent. IBAC takes advantage of the robustness of P300-based Concealed Information Test to detect an intent of access and uses the brain signals to detect the motivation level. The access control system has been used on 30 participants with 100% detected intentions of access and all mal-intent users being rejected access before they commit their mal-intended action.

Emotion-based Access Control (EBAC)

Emotion-based Access Control (EBAC),[7] a novel access control model first proposed by Abdulaziz Almehmadi, is an access control system that detects the emotion of the user requesting access in order to form an access decision. This form of access control adds the sensibility aspect to access control systems to further analyze the risk of granting an authorized user access. A user who has a high level of anger and might cause damage if granted access. As well as denying a malicious authorized user access can be useful, granting a non-authorized user who have good intentions of access can be useful as well (e.g. granting firefighters access to a facility in order to suppress damage).

In some cases, we would wish to deny access to authorized personals in the case that they request access to cause damage. On the other hand, we would wish to grant access to unauthorized individuals. who may suppress damage or prevent catastrophic incidents

EBAC uses emotion detection technology to supplement the access control systems by detecting the emotion of the person requesting access and using it as an additional authentication factor along with the recognized identity of the user as needed. The novelty of the approach is that access is granted based on the actual feelings of the users with regards to the requested resources. The approach is based on the detection of emotion based on the involuntary brain signals that are extremely hard to control or circumvent, and on using the detected emotion in the context of access control.

The EBAC system flow starts with the EEG signal acquisition. The EEG signals are sent via the Emotiv EPOC headset to a listener in the EBAC application. Signals are then analyzed and the emotion is detected with correspondence to the emotion level. The emotion and its rate are then categorized to be either positive or negative. Then data is sent to the decision maker to deny or grant access to the entity.

Attribute-based access control

In attribute-based access control (ABAC),[8][9] access is granted not based on the rights of the subject associated with a user after authentication, but based on attributes of the user. The user has to prove so-called claims about his attributes to the access control engine. An attribute-based access control policy specifies which claims need to be satisfied in order to grant access to an object. For instance the claim could be "older than 18". Any user that can prove this claim is granted access. Users can be anonymous when authentication and identification are not strictly required. One does, however, require means for proving claims anonymously. This can for instance be achieved using anonymous credentials. XACML (extensible access control markup language) is a standard for attribute-based access control. XACML 3.0 was standardized in January 2013.[10]

Break-Glass Access Control Models

Traditionally, access has the purpose of restricting access, thus most access control models follow the "default deny principle", i.e. if a specific access request is not explicitly allowed, it will be denied. This behavior might conflict with the regular operations of a system. In certain situations, humans are willing to take the risk that might be involved in violating an access control policy, if the potential benefit that can be achieved outweighs this risk. This need is especially visible in the health-care domain, where a denied access to patient records can cause the death of a patient. Break-Glass (also called break-the-glass) try to mitigate this by allowing users to override access control decision. Break-Glass can either be implemented in an access control specific manner (e.g. into RBAC),[11] or generic (i.e., independent from the underlying access control model).[12]

Access control based on the responsibility

In **Aligning Access Rights to Governance Needs with the Responsibility MetaModel (ReMMo) in the Frame of Enterprise Architecture**[13] an expressive Responsibility metamodel has been defined and allows representing the existing responsibilities at the business layer and, thereby, allows engineering the access rights required to perform these responsibilities, at the application layer. A method has been proposed to define the access rights more accurately, considering the alignment of the responsibility and RBAC.

Host-based access control (HBAC)

The initialism HBAC stands for "host-based access control".[14]

References

1. ^ *"Unifying identity management and access control". sourcesecurity.com. Retrieved 15 July 2013.*
2. ^ Dieter Gollmann. *Computer Security*, 3rd ed. Wiley Publishing, 2011, p. 387, bottom
3. ^ Marcon, A. L.; Olivo Santin, A.; Stihler, M.; Bachtold, J., "A UCONabc Resilient Authorization Evaluation for Cloud Computing," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 2, pp. 457–467, Feb. 2014 doi: 10.1109/TPDS.2013.113, bottom
4. ^ *"Definition of: clipping level". PC Magazine.*
5. ^ Abdulaziz Almehmadi and Khalil El-Khatib, "On the Possibility of Insider Threat Prevention Using Intent-Based Access Control (IBAC)", *Systems Journal*, IEEE , vol. PP, no. 99, pp. 1, 12, doi: 10.1109/JSYST.2015.2424677 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7103286&isnumber=4357939
6. ^ Abdulaziz Almehmadi "Insider Threats Meet Access Control", URL: https://www.amazon.com/gp/aw/d/1539772012/ref=mp_s_a_1_1?ie=UTF8&qid=1477837714&sr=8-1&pi=AC_SX280_SY350_FMwebp_QL65&keywords=almehmadi&dpPl=1&dpID=518-g2uTc0L&ref=plSrch
7. ^ Abdulaziz Almehmadi and Khalil El-Khatib. 2013. "Authorized! access denied, unauthorized! access granted". In *Proceedings of the 6th International Conference on Security of Information and Networks* (SIN '13).
8. ^ Jin, Xin, Ram Krishnan, and Ravi Sandhu. "A unified attribute-based access control model covering dac, mac and rbac." *Data and Applications Security and Privacy* XXVI. Springer Berlin Heidelberg, 2012. 41–55.
9. ^ *Hu, Vincent C.; Ferraiolo, David; Kuhn, Rick; Schnitzer, Adam; Sandlin, Kenneth; Miller, Robert; Scarfone, Karen. "Guide to Attribute Based Access Control (ABAC) Definition and Considerations" (PDF).*
10. ^ eXtensible Access Control Markup Language (XACML) V3.0 approved as an OASIS Standard , eXtensible Access Control Markup Language (XACML) V3.0 approved as an OASIS Standard.
11. ^ *Ferreira, Ana; Chadwick, David; Farinha, Pedro; Correia, Ricardo; Zao, Gansen; Chiro, Rui; Antunes, Luis (2009). "How to Securely Break into RBAC: The BTG-RBAC Model". Computer Security Applications Conference (ACSAC). IEEE. pp. 23–31. doi:10.1109/ACSAC.2009.12.*
12. ^ *Brucker, Achim D.; Petritsch, Helmut (2009). "Extending Access Control Models with Break-glass.". ACM symposium on access control models and technologies (SACMAT) . ACM Press. pp. 197–206. doi:10.1145/1542207.1542239.*
13. ^ *Feltus C. (2014). Aligning Access Rights to Governance Needs with the Responsibility MetaModel (ReMMo) in the Frame of Enterprise Architecture (PDF).*
14. ^ *Ballard, Ella Deon (2013). "Identity Management Guide: Managing Identity and Authorization Policies for Linux-Based Infrastructures". Red Hat. Retrieved 2014-01-06. "Any PAM service can be identified as to the host-based access control (HBAC) system in IdM."*

External links

- Wireless proximity computer access control system