Authentication, Access Control, and Audit

RAVI SANDHU

George Mason University ⟨*sandhu@isse.gmu.edu*⟩

PIERANGELA SAMARATI

Università degli Studi di Milano, Milano, Italy ⟨*samarati@dsi.unimi.it*⟩

Authentication, access control, and audit together provide the foundation for information and system security.

—Authentication establishes the identity of one party to another. Most commonly authentication establishes the identity of a user to some part of the system, typically by means of a password. More generally, authentication can be computer-to-computer or process-to-process and mutual in both directions.

—Access control determines what one party will allow another to do with respect to resources and objects mediated by the former. Access control usually requires authentication as a prerequisite.

—The audit process gathers data about activity in the system and analyzes it to discover security violations or diagnose their cause. Analysis can occur offline after the fact or online in real time. In the latter case, the process is usually called intrusion detection.

AUTHENTICATION

User-to-computer authentication can be based on one or more of the following:

—something the user knows, such as a password,

—something the user possesses, such as a credit-card-sized cryptographic token or smart card, or

—something the user is, exhibited in a biometric signature such as a fingerprint or voiceprint.

Password-based authentication is the most common technique but it has significant problems. Passwords can be surreptitiously observed or guessed. Password management is required to prod users to change their passwords regularly, to select good ones, and to protect them with care. Excessive password management makes adversaries of users and security administrators, which can be counterproductive. An intrinsic flaw of passwords is that users can share them with other users, which breaks down accountability. However, passwords can be effective and are cheap, so they are likely to remain in use.

The second technique authenticates the token rather than the user. Each token has a unique secret cryptographic key stored within it, used to establish the token's identity via a challenge-response handshake. The party establishing the authentication issues a challenge to which a response is computed using the secret key. Sometimes the challenge is implicitly taken to be the current time. The secret key should never leave the token. Attempts to break the token open to recover the key should cause the key to be destroyed. User-to-token authentication can be based on passwords in the form of a PIN (personal identification number).

Biometric authentication has been

used for some time for high-end applications. The biometric signature should be different every time (for example, a voice-print check of a different challenge phrase on each occasion), or require an active input (for example, the dynamics of handwritten signatures).

Technically, the best combination would be user-to-token biometric authentication, followed by mutual cryptographic authentication between the token and system services. This combination may emerge sooner than we imagine, although there are social issues in addition to technical ones.

Token-based authentication is a technical reality today, but it still lacks major market penetration. Many existing systems use the desktop workstation as a "token" for authentication with the rest of the network. A cryptographic key is computed from the user's password by the workstation, on the basis of which the workstation authenticates to the network. Kaufman et al. [1995] describe some of the techniques in current use.

ACCESS CONTROL

Access controls usually apply after authentication has been established. Access control can take several forms [Sandhu and Samarati 1994].

—Discretionary access control (DAC) is based on the idea that the owner of data should determine who has access to it. DAC allows data to be freely copied from object to object, so even if access to the original data is denied, access to a copy can be obtained.

—Lattice-based access controls [Sandhu 1993], also known as mandatory access controls (MAC), confine the transfer of information to one direction in a lattice of security labels (for example, low to high but not high to low). MAC emerged from confidentiality requirements of the military but has broad applications for integrity and separation objectives.

—Role-based access control (RBAC) requires that access rights be assigned to roles rather than to individual users (as in DAC) [Sandhu et al. 1996]. Users obtain these rights by virtue of being assigned membership in appropriate roles. This simple idea greatly eases the administration of authorizations.

Other forms of access control also exist, and this remains a fertile area for further research and development.

Existing systems often take a feature-based approach to access control in which multiple interacting access-control facilities are configured by security administrators to meet their policy objectives. Unfortunately, these access-control features are often poorly documented and their interactions poorly understood.

AUDIT

Audit has two components: the collection and organization of audit data [Jajodia et al. 1995], and an analysis of the data to discover or diagnose security violations [Lunt 1993; Mukherjee et al. 1994].

Audit data needs protection from modification by an intruder. Vast amounts of audit data can be recorded. Audit data tends to be captured at a low level of abstraction. Analysis of audit data is often performed only when violations are suspected. Even so, only audit data connected with the suspected violation are examined.

Intrusion detection systems seek to help carry out audit controls. *Passive intrusion* detection systems analyze the audit data, usually offline, and bring possible intrusions or violations to the attention of the auditor. *Active systems* analyze audit data in real time and may take immediate protective response, such as killing the suspected process and disabling the account.

The problem is what to look for in

audit data and how to determine automatically whether a violation has occurred or is being perpetrated. The following approaches have been tried: *anomaly detection*, which is based on the assumption that the exploitation of the vulnerabilities of the system involves abnormal use of the system, and *misuse detection,* which is based on rules specifying events, sequences of events, or observable properties of the system, symptomatic of violations.

Finally, we note that audit analysis is an empirical discipline in which we currently have little historical data.

REFERENCES

JAJODIA, S., GADIA, S., AND BHARGAVA, G. 1995. Logical design of audit information in relational databases. In *Information Security: An Integrated Collection of Essays.* Abrams, Jajodia, and Podell, Eds. IEEE Computer Society Press, Los Alamitos, CA, 585–595.

KAUFMAN, C., PERLMAN, R., AND SPECINER, M. 1995. *Network Security*. Prentice-Hall, Englewood Cliffs, NJ.

LUNT, T. F. 1993. A survey of intrusion detection techniques. *Comput. Security 12*, 405–418.

MUKHERJEE, B., HEBERLEIN, L. T., AND LEVITT, K. N. 1994. Network intrusion detection. *IEEE Network* (May/June), 26–41.

SANDHU, R. S. 1993. Lattice-based access control models. *IEEE Computer 26*, 11, 9–19.

SANDHU, R. S. AND SAMARATI, P. 1994. Access control: Principles and practice. *IEEE Communications 32*, 9, 40–48.

SANDHU, R. S., COYNE, E. J., FEINSTEIN, H. L., AND YOUMAN, C. E. 1996. Role-Based Access Control Models. *IEEE Computer 29*, 2.