Security in a Distributed Computing Environment

Today users expect to reach beyond their desktops to take advantage of computing resources spread throughout their organizations. Working in distributed computing environments made up of hardware and software from different manufacturers, users are able to access data wherever it is stored on the network and bring it to the desktop—where it can be used to tackle business problems.

Distributed computing makes the most of computing resources, making them available to any user linked to those facilities by a network. With this expanded computing capability, however, come security risks. Data stored and transmitted on local and wide area networks (LANs and WANs) is vulnerable to unauthorized access, whether intentional or accidental, that could wreak havoc for businesses that rely on the integrity and confidentiality of their files. For that reason, security is a critical component in a distributed computing environment.

This paper describes the need for security in a distributed computing environment and presents a security architecture that addresses that need. In addition, it discusses an implementation of this architecture, the security component of the OSF™ DCE (the Distributed Computing Environment from the Open Software Foundation). DCE is the industry's only set of integrated distributed computing components based on open systems.

THE NEED FOR SECURITY IN A DISTRIBUTED COMPUTING ENVIRONMENT

Office computers connected in networks enable people to easily and inexpensively share resources. Unfortunately, the technology that makes these networks cost-effective also makes them susceptible to eavesdropping and intrusion. Computer networks are implemented by connecting two or more computers with a common communications channel, usually a piece of wire. Because all computers in such schemes are connected to one common cable, a user of a computer connected to the network can eavesdrop easily on communications intended for another computer. Likewise, a computer can be made to "impersonate" another computer and grant a user unauthorized access to data.

The purpose of a computer security system is to allow authorized computer users to access resources in a distributed computing environment securely and confidentially. Controlling such access to information is a tricky problem. Anyone can gain access to the computer network; therefore, a method of distinguishing between computer services and users that may be trusted and those that may not is critical. For this reason both users and services must be capable of proving their identities to each other. Any entity on a computer network that is capable of proving its identity—such as a user, client, or server—is called a *unit of trust* or a *principal*.

In a distributed computing environment, security mechanisms are used to solve these problems. The security system provides *authentication systems* that allow users and resources to identify themselves so they may trust one another. It also ensures data integrity by providing mechanisms that can detect when data has been altered during transmission, and data privacy by providing mechanisms to prevent eavesdropping on network data. An *authorization mechanism* grants authorized users access to resources and rejects requests from unauthorized users.

The OSF DCE provides an architecture that addresses the needs for a secure computing environment. The security component of DCE is integrated with the DCE remote procedure call mechanism. Together they offer application programmers and end users several levels of security including high-level data privacy, in which all network data is protected and can be viewed only by the intended recipient, as well the option of choosing no security.

The OSF DCE Security Service makes use of *Kerberos authentication,* which allows principals to identify one another. Kerberos uses *secret key encryption*, which allows confidential data to be sent through the network without being compromised. DCE security makes use of *cryptographic checksum* techniques that ensure data integrity by allowing corrupted data to be detected easily. In addition, DCE security provides the *DCE Registry Service,* which allows system administrators to manage the database of valid principals easily, and a distributed *access control mechanism* that allows users and administrators to control access to resources. The remainder of this paper discusses the DCE security architecture and its implementation.

DATA ENCRYPTION TECHNIQUES

Network data is vulnerable to unauthorized access and tampering because it is shipped through a communications link, such as a piece of wire, or microwave transmissions. Without much difficulty, someone could tap into the communications link and monitor information being transmitted. One way to protect network data is to scramble it in a way that can be reversed only by those parties intended to receive it.

The scrambling technique used in secure communication is called *encryption*. A message is encrypted, or encoded, in a way that can be reversed only with the appropriate password, or *secret key*. Simple encryption techniques were first employed during the early days of the Roman Empire for the secure communication of battle plans between generals. Simple encryption techniques are used in everyday life. In the secret decoder rings offered by advertisers in early radio programs, for example, a number was substituted for each letter in the alphabet. The set of numbers needed to decode a message made up the key. The operation of substituting the numbers for the letters is the *encryption algorithm*. Consequently, computer encryption can be described as a mathematical method of locking information in an impenetrable box that can be opened only with the correct mathematical key.

Secret decoder rings used simple encryption techniques. In comparison, today's encryption algorithms are very sophisticated and cannot be decoded without a secret key.

KERBEROS AUTHENTICATION USING SECRET KEY ENCRYPTION

The OSF DCE makes use of the Kerberos authentication service from the Massachusetts Institute of Technology's Project Athena. Kerberos uses a sophisticated authentication mechanism that meets the requirements of open computer networks.

Kerberos requires principals to exchange secret messages that prove their identities to each other. A scenario from the world of international espionage, in which one secret agent must pass microfilm containing top-secret information to another agent, demonstrates how this authentication scheme works.

The two agents meet on a dark corner in Gorky Park. Before carrying out their mission, however, they must ascertain each other's identity. Unlike ordinary citizens, the agents do not depend on written credentials such as passports, which can be forged or stolen. Such documents have no credence in the spy world. For that reason, the spies exchange secret messages.

"The moon is in the fourth quarter," says one agent. The other responds, "Yes, it's a beautiful night for a dance." Through that exchange, the spies verify each other's identity, developing sufficient trust to allow them to execute their duties.

Computers and users in a network environment must address a similar problem. Before they can exchange information, they must be certain of each other's identity and confident that they can trust each other.

For example, a user might want to access information from a database service without compromising the integrity of his or her information. Before retrieving the information, the database service must trust the user. Likewise, users must trust a printer service before they will send it confidential information. (For example, they would not want to send private data to a phony printer service that collects confidential information and transmits it to a competitor.)

To foster the requisite trust between principals, Kerberos allows principals to identify themselves and verify their identities using secret key encryption technology. The unalterable credentials it generates allow user principals to identify themselves to network services. These credentials, called *tickets*, contain user information such as name and location. In addition, Kerberos provides the principals with a secret key, which they use to exchange secret messages. Possession of the secret key, which allows each party to encode and decode the secret messages, constitutes verification of the credentials.

DATA INTEGRITY

An important function of computer security systems is to ensure that information is not altered or corrupted as it flows through the computer network. One solution is to encrypt all information before sending it through the network. This prevents unauthorized individuals from reading or modifying data on the network.

There are two reasons users might not want to encrypt all their data. First, not all computer networks need such a high level of security. In many cases, a security system need ensure only that data arrives at its destination unaltered. For data that falls into this category, networks might be able to use higher performance data integrity mechanisms that operate faster than data encryption.

Second, some governments impose export and import restrictions on data encryption. For example, many governments restrict the use of certain data encryption mechanisms through their public communication networks. For these reasons, users may be limited to data integrity mechanisms to protect their data.

The OSF DCE provides a high level of data integrity through the DCE RPC component. The DCE RPC ensures data integrity by using cryptographic data checksums. This allows the receiver of data to determine whether it was modified or corrupted while passing through the network.

Data checksum mechanisms take advantage of the fact that computer data is always represented in a numerical form. They employ a mathematical operation that can detect whether data has been modified.

A simple example of a data checksum is adding all the data in a single transmission and then appending the checksum onto the transmission. The receiver then could recalculate the checksum and compare it with the original. If someone on the network were to alter the data or if the data were corrupted through interference, the two checksums would differ.

Cryptographic data checksums use more complicated algorithms that make use of encryption keys to calculate the checksum. This prevents unauthorized individuals from altering the data and then updating the checksum.

PRIVACY OF DATA

The OSF DCE, through its remote procedure call (RPC) component, uses trusted *third party secret key encryption* to ensure the privacy of data. Trusted third party secret key encryption provides a method for distributing secret encryption keys to both parties of a secure communication. This method makes use of a trusted third party, called the *key distribution center* (KDC),

which maintains a database of all secret keys in the distributed computing environment. Two principals may hold a secure communication by obtaining the same secret key from the KDC. This key can be used for encoding and decoding messages.

The RPC mechanism hides the complexity of the security system from the user by automatically encrypting and decrypting data as it is transferred through the network. The user need only select the level of security for each communication; the RPC does the rest of the work. A user may choose between two levels of security: private and non-private. The level determines whether all or none of the data is encrypted.

REGISTERING PRINCIPALS WITH DCE REGISTRY SERVICE

The OSF DCE Registry Service provides a principal registry system for creating and maintaining a database of valid principals. This distributed database system can be accessed from any computer participating in the distributed computing environment. Like all other DCE components, the registry service is highly available, extensible, efficient, and secure.

DCE AUTHORIZATION USING ACCESS CONTROL LISTS

Once users are authenticated, the service must verify which operations, such as read or modify, they are permitted to perform on the information they try to access. This process is referred to as *authorization*. A payroll office might control access to its payroll database on a per-individual basis, for example. Individuals might be allowed to look up their own salaries but not the salary of others. Financial managers might be the only ones allowed to modify salary entries.

The OSF DCE uses access control lists (ACLs) to control user access to distributed computer resources. ACLs are associated with each computer resource and contain a list of names, or principals, and the types of operations they are permitted to perform on these resources. Examples of these operations include permission to print on a printer or modify a data record in a database.

The names in ACLs can be individual users names or names that represent a group of users. Group names are useful for providing a user access to a large number of resources. For instance, when a user joins the engineering division of a company, he or she might be given access to all engineering resources such as printers, files, and databases. Rather than adding the user's name to each resource, the administrator can add the new user's name to the engineering group, thus providing the user access to all the engineering resources associated with a particular access group.

The OSF DCE access control lists are a superset of the access lists specified by the POSIX 1003.6 ACL working group. POSIX ACLs were designed to control the file access of users sharing a single computer. The extensions of DCE ACLs allow them to be useful in a distributed computing environment.

The DCE allows lists of users to have ownership and administrative rights to distributed resources. This allows more than one individual to control the types of access other users may have to these resources. In addition, the DCE allows ACLs to contain names of individuals from organizations other than the one that administers the resources the ACLs protect. For instance, a user from a company in Boston can be given individual access to a laser printer at a different company in Munich. POSIX ACLs, on the other hand, limit inclusion to members within one organization, and are applicable only to files.

CONCLUSION

As companies make greater use of distributed computing, the need to protect the privacy and integrity of data transmitted over networks becomes increasingly critical. The Open Software Foundation's Distributed Computing Environment addresses this need by protecting against unauthorized access to data—both intentional and inadvertent—as well as preventing the corruption of data.

The security component of DCE is fully integrated with other DCE technologies, such as DCE RPC mechanism, to provide a high level of security without exposing the underlying complexity to application developers and end users. These features make it possible for business, engineering, and research users to work in a global computing network without compromising their private resources and confidential information. In addition, the security component is used by all DCE services, such as the Cell Directory Service (CDS) and Distributed File Service (DFS), thus providing a unified security architecture. In this way, the DCE provides a safe environment in which information can be accessed and shared securely.

OSF and the OSF logo are trademarks of the Open Software Foundation, Inc.
©opyright 1992. Open Software Foundation, Inc. All Rights Reserved.

Open Software Foundation/North America
11 Cambridge Center
Cambridge, MA 02142
United States
617 621 8700

4 Main Street
Garden Level #7
Los Altos, CA 94022
415 948 1171

620 Herndon Parkway
Suite 200
Herndon, VA 22070
703 481 9819

Open Software Foundation/Europe
Avenue des Pléiades-laan 15
B-1200 Brussels
Belgium
32 2 772 88 88

Research Institute
2 Avenue de Vignate
38610 Gieres
France
33 76 63 4863

Open Software Foundation/Pacific
11-10, Kita-Aoyama 2-chome
Minato-ku, Tokyo 107
Japan
81 3 3479 4740

OSF-O-WP11-1090-3

OPEN SOFTWARE FOUNDATION

Security in a Distributed Computing Environment

A White Paper
January 1992