Volume 2, No. 7, July 2011

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

SECURITY ISSUES IN COMPUTER NETWORK ARCHITECTURE

Dr. M. Lilly Florence*[1], D.Swamydoss[2]

[1]Adhiyamaan College of Engineering Hosur, Tamilnadu
lilly_swamy@yahoo.co.in
[2] Adhiyamaan College of Engineering Hosur, Tamilnadu.
Swamy_asir@yahoo.co.in

Abstract: The architecture of a computer network has evolved with advances in technology. The design of secure computer network architecture to protect the integrity of information exchange is pursued by the commercial and financial sectors and at all levels of government agencies. Active networks represent a new approach to network architecture. It provides a much more flexible network infrastructure. The network security is mainly based on the network architecture. The purpose of this paper is to provide a broad survey on security in network system architecture. The first goal is to discuss various network architecture. The second goal is to highlight security issues in network architecture. Thus an inclusive presentation of network architecture, security issues is given.

Keywords: Peer – to – Peer Network, Client – Server Model, Network Security, authentication, NCSC.

INTRODUCTION

Organizations are increasingly driven toward integrated interaction throughout their value chain: the ability to buy and sell using seamless electronic technologies is changing from a competitive advantage to a pre-requisite in many sectors. Current systems and communications standards are reducing the distinction between Local Area Networking and the remote operation of information systems using third-party telecommunications services. It is therefore becoming more realistic for smaller companies to connect their systems with those of their trading partners irrespective of organization size, structure, computing platforms or geographic location.

Traditionally, the function of a network has been to deliver packets from one endpoint to another. There was a distinct boundary between what is done within the network and what is done by the users. Processing within the network was limited basically to routing, congestion Control and quality of service (QoS) schemes. This kind of a network can be regarded as "passive". Several problems with "passive" networks have been identified: the difficulty of integrating new technologies and standards into the shared network infrastructure, poor performance due to redundant operations at several protocol layers, and difficulty accommodating new services in the existing architectural model. An additional shortcoming is that, recently, applications which sometimes require computations within the network have emerged, such as firewalls, Web proxies, multicast routers, and mobile proxies. In the absence of architectural support for doing so, these applications have adopted a variety of ad hoc services for performing user-driven computations at nodes within the network. A need was felt to replace the numerous ad hoc approaches to network-based computation, with a generic capability that allows the users to program their networks. This innovative idea of imparting the user the ability to program the network is called active networking.

Active networks represent a new approach to network architecture. These networks are “active" in two ways : routers and switches within the network can perform computations on user data flowing through them; and users can "program" the network, by supplying their own programs to perform these computations. In the extreme case, there will be no difference between internal nodes and end user nodes since both will be able, if needed, to perform the same computations.

When designing complex systems, such as a network, a common engineering approach is to use the concepts of modules and modularity. In this approach, the design of the system evolves by breaking the big task into smaller tasks. Each module is responsible for a specific task and provides services to the other modules to accomplish their tasks. We can interact with a module as a black box that provides certain functionality without knowing the details of how it works. We only need to know how to interface with the module. Someone can remove the module and update it with a newer one, and we would still be able to continue our work in the same way. Moreover, modularity is important to simplify tasks (divide and conquer). For instance, in a network, reliability of message delivery and routing of messages can be treated separately by different modules. Changing one would not impact the other. If a better routing procedure is employed, it only affects the module responsible for it. Certainly, we do not desire a change in routing to affect our ability to reliably deliver messages. Modules often interact in a hierarchy. A network is designed as a hierarchical or layered architecture in which every module or layer provides services to the upper layer. Users, sitting at the top layer of the network, communicate as if there is a *virtual link* between them, and need not be aware of the details of the network.

NETWORK STRUCTURES AND ARCHITECTURE

Networks are usually classified using three properties: Topology, Protocol, and Architecture. Topology specifies the geometric arrangement of the network. Common topologies are a bus, ring, and star. A bus topology means that each computer on the network is attached to a common central cable, called a bus or backbone. This is a rather simple network to set up. Ethernets use this topology. A ring topology means that each computer is connected to two others, and they arranged in a ring shape. These are difficult to set up, but offer high bandwidth. A star topology means all computers on the network are connected to a central hub. These are easy to set up, but bottlenecks can occur because all data must pass through the hub.

Architecture refers to one of the two major types of network architecture: Peer-to-peer or client/server. In a Peer-to-Peer networking configuration, there is no server, and computers simply connect with each other in a workgroup to share files, printers, and Internet access. This is most commonly found in home configurations, and is only practical for workgroups of a dozen or less computers. In a client/server network, there is usually an NT Domain Controller, which all of the computers log on to. This server can provide various services, including centrally routed Internet Access, mail (including e-mail), file sharing, and printer access, as well as ensuring security across the network. This is most commonly found in corporate configurations, where network security is essential.

PEER TO PEER NETWORK ARCHITECTURE

Peer-to-peer (P2P) networks have become popular for certain applications and deployments for a variety of reasons, including fault tolerance, economics, and legal issues. Peer-to-peer systems have two dominant features that distinguish them from a more standard client-server model of information distribution: they are overlay networks that have unique namespaces. P2P systems link different, possibly heterogeneous systems as ‘peers,’ and allow them to interact on top of existing network configurations. It does this by defining relationships unique to that system, usually in the form of a topology by which systems are linked. Occasionally a system will piggyback atop an existing namespace, such as the IP/port labeling, but still treats these separately from the Internet-layer protocols. The combined effect of independent systems interacting through a unique namespace is *decentralization*. Not every system generally considered a P2P system is strictly decentralized in management—the infamous Napster actually ran through a centralized server—but the matter of decentralization brings out important concerns for system security.

Although they have won the most attention for their roles as (often illicit) file-swapping networks, peer-to-peer systems can serve many functions, and design considerations must reflect these. One proposed use of a P2P system was that of shared expertise. A Gnutella pioneer has suggested that the system could be used to propagate queries of any sort through a network until a peer felt that it could respond to the query, anything from a lexical look-up to a specialized calculation (Kan, 2000). In a network with varied resource availability, P2P systems have been used to distribute those resources to those who need or want them the most. The shared resources in question are usually computation or storage. This can create efficient resource usage, given a low marginal cost of using a resource not otherwise needed. Beyond sharing computation power for enormous tasks, P2P networks have been proposed as an escrow system for digital rights management, or for distributing searches across a wide range of other peers. A new class of business software systems known as ‘groupware’ uses many P2P principles. Groupware networks are closed networks that support collaborative work, such as the Groove network. Finally, the decentralized nature of peer systems offers many positive features for publishing and distribution systems, not least of which is their resilience to legal and physical attacks and censorship. P2P architectures have many different functions, and different functions lead to different design conclusions, with respect to overall system structure, and specifically to security issues.

SECURITY ISSUES IN PEER TO PEER ARCHITECTURE

Security expert Bruce Schneier is often quoted as claiming that “Security is a process.” As such, it matters very much whether security administration is centralized or decentralized. The basic model of the commercial Internet is the client-server relationship. As the network user base grew, less and less responsibility for administration was placed on the edges of the network, and more was concentrated in smart ‘servers.’ This model is most evident on the World Wide Web, but file servers, mail servers and centralized security administration mechanisms such as Virtual Private Networks have all grown apace. In a centralized system, security policy can be dictated from a single location, and a ‘that which is not permitted is forbidden’ policy can be enforced with firewalls, monitoring and intervention.

On the other hand, centralization offers a single point of failure. Both direct malicious attacks and lax or negligent administration at the heart of a centralized network can undermine security for an entire system. With a single breach, outgoing or incoming content can be corrupted or intercepted, or the system can be rendered inoperable with a denial of service attack. Critical infrastructure systems such as the Domain Name System (DNS) have redundant implementation exactly because a single point of control is vulnerability in and of itself. In a decentralized P2P system, bad behavior has a locality impediment. Malicious attacks need to occur at or near every part of the system it wishes to affect.

P2P systems lack the tools available to a centralized administrator, so it can be much more difficult to implement security protections on a deployed P2P system. Moreover, the absence of a defensible border of a system means that it is hard to know friend from foe. Malicious users can actively play the role of insiders—i.e., they can run peers themselves, and often a great number of them. Doceur (2002) notes that this is incredibly hard for any open system to defend itself against this sort of attack, known as a Sybil attack (named after a famous Multiple-Personality Disorder case study).

CLIENT – SERVER NETWORK ARCHITECTURE

Client/server is an architecture in which a system's functionality and its processing are divided between the client PC and a database server. System functionality, such as programming logic, business rules and data management is segregated between the client and server. The distribution of services in client/ server increases the susceptibility of these systems to damage from viruses, fraud, physical damage and misuse than in any centralized computer system. With businesses moving towards multi-vendor systems, often chosen on the basis of cost alone, the security issues multiply. Security has to encompass the host system, PCs, LANs, workstations, global WANs and the users.

However, every level of system security requires dollars and additional steps for the users. The cost and inconvenience (to the users) associated with security must be balanced against the cost and inconvenience of corrupted or insecure data.

SECURITY ISSUES IN CLIENT – SERVER ARCHITECTURE

The network connecting clients and servers is a less than secure vehicle that intruders can use to break into computer systems and their various resources. Using publicly available utilities and hardware an attacker can eavesdrop on a network, or "sniff" the network to read packets of information. These packets can contain useful information, e.g. passwords, company details, etc, or reveal weaknesses in the system that can be used to break into the system.

Encryption of data can solve the problem of attackers sniffing the network for valuable data. Encryption involves converting the readable data into unreadable data. Only those knowing the decryption key can read the data. A problem here is that some network operating systems don't start encryption until the user has been authenticated (i.e. the password is sent unencrypted).

Most systems employ re-usable passwords for authenticating users which allows an attacker to monitor the network, extract the login information and access the system posing as that user. Even if the password is encrypted the intruder can just inject that packet into the network and gain access. The problem is compounded when, to maintain that single system illusion, only one login is required to access all servers on the network. Customers want a "single system image" of all networked computing resources, in which all systems management and administration can be handled within a single pool of system resources. To have a secure network it must conform to four basic principles of a trusted computing base (TCB):

- Identification and authorisation
- Discretionary control
- Audit
- Object re-use.

The fundamental aspect of the TCB is that if you can trust all of the security features, then the network can also be trusted. The TCB must be self protected against tampering and malicious, inadvertent altering and attempts to circumvent it.

The National Computer Security Centre's (NCSC) evaluation model specifies a C2 level of security that provides the above features. C2 is a defined level for operating systems requiring users and applications to be authenticated before gaining access to any operating system resource. All clients must provide an authenticated user ID, access control lists must protect all resources, audit trails must be provided, and access rights must not be passed to other users that re-use the same items.

User authentication can be managed by the Kerberos authentication mechanism. The Kerberos protocol, with add ons introduced by the Open Software Foundations Distributed Computing Environment (OSF DCE), fulfills the authentication requirement of C2. Kerberos is a secret key network authentication system that uses DES for encryption and authentication. It authenticates every user for every application. It consists of three distinct services that provide access to network resources, Registry, Authentication and Privilege Servers.

In DCE, users, servers and client computers are all referred to as "principals". The Registry server creates user accounts for the network principals and unique user IDs are created and stored with other information in the security database. At the time of logon, the DCE security system uses encrypted "keys" that reflect private validation information, such as a principal's password, the server and services or data required. This process of validating a principal through credentials is known as authentication. At the time of the logon request, a logon request is submitted to the Authentication Server. [6, 7]

The Authentication Server responds to the logon service by forwarding a ticket that is encrypted using the user's secure key. Once received by the logon service, an attempt to decrypt the ticket is made using the password supplied by the principal. If a valid password has been supplied, a new ticket will be created by the logon service. This ticket is then forwarded to a Privilege Server, where a new ticket is created that will be used to provide access to a specific application server and the specific services required. Properties of the Kerberos tickets include encryption of the ticket to insure its authenticity. If any data within the ticket is modified, the ticket becomes invalid. Further, tickets are issued with a specific time duration to protect against an attempt to modify, copy, or utilize someone else's ticket. The DCE Security Service is designed to allow applications to authenticate not only users, but servers as well. This is provided with each application server having again its own secret key that can be used by the security service to send packages that only the true server can decrypt. Once the client receives the ticket, it presents the ticket and request to the targeted server. The server then compares the incoming ticket with the ticket from the Kerberos server to verify that the client has access.

Applications, files, and other DCE services then use Access Control Lists (ACLs) to identify specific access privileges of a principal, whose individual or group identity is provided via the ticket. An ACL might identify whether a principal has read, write, test, or other permissions against a file or records within a database. If the ACL permissions match the action requested, the access is granted; otherwise, the RPC call is denied and a result returned to the calling application.

The purpose of the DCE Security Service is to provide a security architecture that provides both users and servers the ability to communicate in a manner which is very difficult to impersonate. This allows servers to offer service only to authorised users, and it allows users to have confidence that they are communicating with the "real" server.

Audit services allow network managers to monitor user activities, including attempted logons and the file servers or files used. It is achieved by monitoring all user workstations and recording transaction activity. Most network operating systems support audit trails. Securing the network also requires securing the devices. This is particularly important on the devices that interconnect large parts of the enterprise, such as the backbone or campus routers. These devices should be in one spot or in secure rooms placed strategically around the enterprise. They can be engineered to generate an alarm (raise a management event) if the cases are opened, module removed or if anything is changed.

Additional levels of security are also provided by building in rules, filters and screening into the interconnecting devices. For example, a router can be configured to only allow certain IP addresses access to a segment of the backbone.

CONCLUSION

Engineering security in network architecture is not an easy task. Risk assessment and anticipated threats to any network should be examined and studied so the proper security policy can be adopted. Some security appliance vendors have acknowledge that the security solutions presently available are not equipped to handle all types of network and application layer attack. Network security is a system engineering discipline. In the end, secure computer network architecture is not enough, a personal commitment to security awareness and a dedication to a security policy might protect us in an insecure computer network environment.

REFERENCES

[1] IPAM – Security Cyberspace: Applications and Foundations of Cryptography and Computer Security, PP. 2.

[2] Introudction to System and Network Security, Learning Tree Internation Technology Training Course 468 Material.

[3] D . L . Tennen house et a / . , A Survey of Active Network Research, IEEE Commun. Mag., vol. 35(1).

[4] Christian F. Tschudin , Active Network Overlay Network (ANON), RFC Draft,.

[5] S . Bhattacharjee, K . L.Calvert, and E W.Zequra , An Architecture for Active Networking, Proc. IEEE INFOCOM.

[6] D . Scott et a/. , A Secure Active Network Environment Architecture, IEEE Network, special issue on Active and Programmable Networks.

[7] B. R. Smith, and J. J. Garcia-Luna-Aceves, “Efficient security mechanisms for the border gateway routing protocol," Computer Communications, vol. 21, no. 3, pp. 203-210.

[8] J.M. Doar, “A better model for generating test networks," in Proceedings of Globecom .

[9] Cisco Systems Inc, “Network security: An executive overview,". http://www.managednetworks.com/docs/networksecurityoverview.pdf

[10] M. Hendry, Practical Computer Network Security. Norwood, MA.: Artech house.

[11] B. Schneier, Applied Cryptography: Protocols, Algorithms and Source Code in C. NY.: John Wiley & Sons.