Managing Security Vulnerabilities in a Networked World

Alan Rudd, Joel McFarland, and Scott Olsen

DRAMATIC GROWTH in networked computing led to vulnerabilities in operating systems and network applications which were traditionally identified by a system administrator, who would apply countermeasures as appropriate. The workload of a typical administrator was so substantial that following up to verify if the countermeasure was functioning properly—or if even still in place after a month or two—was often neglected. Learning new vulnerabilities was a time-consuming process of reviewing security bulletins from varying organizations, and it was rare that the individual in charge of fixing security holes had enough time to be thorough in his review of these documents. It was virtually impossible to ensure the security of all hosts on a network using this technique. As the number of Internet connections, remote access modems, and “trusted” business partners proliferated, so did the risks. Because of the peer relationship among network hosts and the broadcast nature of LAN technology, this meant that if a malicious party gained access to one networked computer, the security of the entire network was at risk. Firewalls were designed to be a single point on the network that would regulate all incoming and outgoing traffic, breaking down its own connections into an “untrusted” and a “trusted” side, acting as a gatekeeper for all traffic flow between the two. Typically, firewall implementation gave management and system administrators a false sense of security regarding their networks. This resulted in a general dismissal of the value of fixing security problems on individual hosts and led to an increasingly vulnerable internal network susceptible to attacks via modems on desktop computers, insider attacks, and penetrations through holes in misconfigured firewalls. Performance limitations also prevented firewalls from being a complete scalable solution on the internal network. Today’s security expert recognizes that there is no one simple solution to providing strong security for increasingly valuable networks. There is no “silver

bullet.” For effective security, organizations should invest in a robust security architecture and a multi-step, continuous process. This process includes the strategic use of security “resources” and the deployment of enterprise-scale technology that delivers the performance and management characteristics to optimize security operations.

Managing the Vulnerability Process

This new dynamic approach to security management—the Vulnerability Management Process (represented in Fig 1 by the Security Wheel; Wheelgroup Corp) is a process by which organizations can incorporate existing security applications and investments into an on-going, operations-based process for addressing vulnerability. Security policy should be a practical, living document outlining the organization’s security objectives and documenting the resources to be protected. Secondly, strong security devices should be deployed controlling access to network resources. Third, organizations should, in real time, monitor their networks for misuse and violations of security policy including on-going testing of network resources using automated tools to look for vulnerabilities. Unlike the old paradigm, which required companies to closely follow security bulletins and scramble to apply patches, organizations should utilize sophisticated security tools that have comprehensive security vulnerability databases, thereby eliminating the tedious and expensive labor element required to track everything. The final step before repeating the cycle is to “manage and improve” the network security posture. This requires detailed information from the monitoring and testing applications so that the information professional can make sound decisions about where to spend limited resources to improve security. This stage demands a testing device, which not only identifies security problems on the network, but can also present this data in a format that can effectively communicate risks to both administrators and management.

A FOUR-STEP PLAN FOR A HEALTHY NETWORK

Phase I: Network Mapping

Attempting to identify any device on the network that can be reached with the TCP/IP will result in a

From Storage Technology Corp, Charlotte, NC; and Wheelgroup Corp.

Address reprint requests Alan Rudd, Storage Technology Corp, 5500 77 Center Dr, Suite 110, Charlotte, NC 28217.

*Copyright © 1998 by W.B. Saunders Company
0897-1889/98/1103-1072\$8.00/0*

Fig 1. Security wheel.

comprehensive network map. It can be obtained using numerous techniques such as ICMP Request protocol, commonly known as a “ping”, to query hosts and network devices on a given LAN. This is the method used to generate an “electronic map” of the tested network shown in Fig 2. An electronic map is a profile of all live hosts at the particular time that a scan is run. It is important that the scanning tool be flexible enough to schedule and run at any particular time of day and also be set up to run recurring scans every day, week, month, or even at random. Useful information can also be obtained by examining router and switch tables using SNMP and other network management applications. In the future, more information will be gathered from other devices on the network such as routers and switches by generating SNMP queries for routing tables. Many organizations store a great deal of information about their network topology in their Domain Name Servers (DNS), such as host name, domain name, hardware, operating system, and occasionally some application-specific data. This data is used to generate a “registered map” of the tested network. This map shows what is advertised by a given DNS server and can be useful in

Fig 2. Identifying live hosts.

determining what information outsiders can gather about a protected network. Revealing too much information in public DNS servers makes a hacker’s job much easier. Identifying systems by names such as “finance.organization.com” or “research.organization.com”, offers hackers specific internal targets. This type of information should be restricted to protected internal DNS servers and not advertised to the Internet.

Phase II: Data Collection:

Once a network map has been generated, a baseline becomes available for further data collection and testing. The primary goal of this phase is to gather, and store in a database, data on all live hosts and network devices identified in the previous phase to determine operating system, version numbers, network applications, and numerous potential vulnerabilities. “Port sweeps” identify services and collect data about electronic resources on a tested network. Flexibility must allow users to customize their scan for a unique environment. For instance, they may want to do full scans on their key servers and perform a fast scan for the rest of their network.

Phase III: Data Analysis:

After the data has been collected and stored in a database, a sophisticated vulnerability analysis engine needs to identify network devices, operating systems, version numbers, available applications, and potential vulnerabilities. These include routers, switches, firewalls, and network servers with operating systems and version numbers wherever possible. This identification analyzes all collected data in a batch mode and compares this data against known characteristics of particular operating systems and versions. During the data analysis phase, security-focused information will be identified. First, a database of all applications available on network hosts is created. This is critical to identifying adherence to security policy and to identify potential weaknesses in the network. Then all potential vulnerabilities on the network are identified and collected in a Network Security Database (NSDB). The NSDB is a comprehensive repository of vulnerability “signatures,” which identify known security risks, both “potential” and “confirmed” vulnerabilities, from security “holes” in Windows

NT or UNIX systems to mis-configured routers. During this phase, most identified potential vulnerabilities are public services, applications with known holes, and system mis-configurations. Public services are identified by the general port sweep. Services such as tftp, rlogin, and rsh are typical missed vulnerabilities. Even though they are legitimate services for a given operating system, their public use can introduce unwanted holes in a system. We must also find specific applications that have security risks associated with them. The most notorious application for vulnerabilities is sendmail. The final type of vulnerability to look for in this phase is a system mis-configuration. This can include an NT host that is sharing its files without strong user access restrictions, as well as a UNIX host using NFS to share its file systems with the world.

Phase IV: Vulnerability Confirmation:

Unfortunately, there are a good number of vulnerabilities that cannot be identified by simply looking at collected network data. In those instances, “confirms” of those vulnerabilities must be done with great care not to disrupt or damage the host being tested. These tests identify vulnerabilities by taking

advantage of their weakness to show that a host is actually vulnerable.

Phase V: Data Presentation and Reporting:

Sophisticated data presentation capabilities incorporating a hyper-linked spreadsheet, or “grid browser”, and a powerful report wizard allow vulnerabilities to be displayed based on address ranges or operating systems to allow an organization to identify trends of security weaknesses across their enterprise. This allows the user to fix identified problems on their network without having to wade through huge reports of complex information by clearly defining:

- a description of the problem
- the potential damage that can be caused using a particular vulnerability
- links to web sites for patches and further information, and
- A link to a comments page that allows users to customize information about a particular vulnerability.

This enables organizations to document their fix procedure and to record on-going problems with their network.