

Chapter 6
Security Architecture

Jack Suess

**Computer and Network Security
in Higher Education**

Mark Luker and Rodney Petersen, Editors

A Publication of EDUCAUSE

Copyright 2003 Jossey-Bass Inc.

Published by Jossey-Bass, A Wiley Company. Reprinted by permission of John Wiley & Sons, Inc. For personal use only. Not for distribution.

Security Architecture

Jack Suess

The focus of this chapter will be on how institutions can use an IT security architecture to “build in” security as we plan, design, and deploy the networks, computers, middleware, and applications that make up our IT infrastructure.

It is important to acknowledge at the beginning that there is no single solution for an IT security architecture that will work across the thousands of higher education institutions in existence today; however, there are common elements of an IT security architecture that each campus should consider when developing its security plan. These common elements include network security, computer (or “host”) security, middleware and directory services, and application-based security. An IT security architecture should be integrated with the broader IT plan for the campus and support those IT initiatives proposed in the plan. In fact, many aspects of IT security architecture, such as the use of a central directory for authentication, can be enabling technologies that facilitate the development of a broad range of IT initiatives (Barton and others, 2001).

A second acknowledgment is that our IT infrastructure is constantly evolving. As a result, our security architecture must be adapted to keep pace. This is a curse in that our work is never complete, but also a blessing in that we can opportunistically replace technology in accordance with our IT plan and at the same time enhance security.

The remainder of this chapter discusses each element of an IT security architecture. Purposely, this chapter is written at a high level and is not directed to network engineers and system administrators. An excellent primer for technical personnel is RFC 2196—“Site Security Handbook” developed by the Internet Engineering Task Force (Fraser, 1997).

Network Security

Network security architecture is the planning and design of the campus network to reduce security risks in accordance with the institution’s risk analysis and security policies. It focuses on reducing security risks and enforcing policy through the design and configuration of firewalls, routers, and other network equipment.

Network security is important because it is one of the means to enforce the policies and procedures developed by the institution to protect information. It is often referred to as the “front door” in broader discussions of IT security. To the extent that you can block network access to a computer, you “lock” the door and provide better protection for that computer and its contents.

Traditional network design has focused on creating a secure network perimeter around the organization and strategically placing a firewall at the point where the network is connected to the Internet. For higher education, this traditional design is problematic; our constituents need access from off campus to a large number of machines and services on campus. In addition, because we have many computers on our campus that we cannot implicitly trust, we also must be concerned about security threats from inside the perimeter protected by a traditional firewall. These design issues require a different approach to network security. Although it is impossible to do justice to the topic of network design in a few pages, there are some best practices that I feel universities should focus on in terms of network design:

Step 1: Eliminate Network Components That Still Use Shared Ethernet

Shared Ethernet switches (or hubs) were developed more than a decade ago to interconnect multiple computers and networks. These hubs retransmit all network traffic to all computers connected to that hub. The security implication is that if one computer has its security compromised it can be used to monitor network traffic coming from any other computer that shares the same hub. This could expose passwords and other sensitive information. Today, switched Ethernet, which isolates traffic intended for one computer from the view of others on the same switch, is very inexpensive and, hence, it is worth the cost of replacing older hubs.

Step 2: Embrace and Implement the Concept of Defense and Use Multiple Firewalls Within Your Network

Commercial and Linux-based firewalls are inexpensive enough that you can deploy these in multiple locations as needed. It is still beneficial to have a firewall separating your institutional network from the connection to the Internet. This firewall, called a *border firewall*, will provide a minimal level of protection for all computers on your network. The major benefit of this firewall is that it allows your network and security staff to quickly block external access should a threat arise, such as when the “SQL worm” was launched in January 2003 (“Safe SQL Slammer Worm Attack Mitigation,” 2003). In addition to the border firewall, consider adding internal firewalls to protect areas that require different levels of security. For example, placing a firewall between the network segments containing the computers that operate the institutional business systems allows the institution to provide more restrictive security for those computers. Other areas that firewalls can strengthen include residential networks and research labs. Each firewall can have different access controls, support different security policies, and allow for distributed administration—all of which are essential to success in academia (Gray, 2003).

Step 3: Implement Intrusion Detection Systems at Key Points Within Your Network to Monitor Threats and Attacks

An *intrusion detection system* (IDS) looks at the incoming network traffic for patterns that can signify that a person is probing your network for vulnerable computers. The IDS can also look at traffic leaving your institution for patterns that might indicate that a computer's security has been compromised. This probing from off campus is usually the first step in attempting to compromise the security of a computer on your network. IDSs historically have produced daily reports showing what security vulnerabilities were being targeted the day before.

Some vendors are now integrating the IDS with the firewall and renaming these *intrusion prevention systems*. When a threat is identified, the IDS automatically works with the firewall to adjust the firewall rules to protect the computers on the network. IDS products are broadly available through commercial vendors and the open-source community. At my institution, we use an open-source product named Snort (Grimes, 2002; Roesch, 2003).

Step 4: Implement a Virtual Private Network Concentrator for Off-Campus and Wireless Access

A *virtual private network* (VPN) uses special software on each computer, called a VPN client, to encrypt network traffic from that computer to a VPN concentrator on the institution's network. Using a VPN allows a member of your institution to securely connect to campus computers from an off-campus computer. The VPN will establish an encrypted connection that allows the off-campus computer to appear as if it were part of your internal campus network, thereby granting access to resources that may be blocked by a border firewall (Frasier, 2002).

Many institutions are actively implementing wireless networks on campus. Wireless networks can create many security considerations because their signals typically are shared over a broad area. In particular, wireless networks are very much akin to shared Ethernet and may be susceptible to surreptitious monitoring of network traf-

fic. You should encrypt your wireless network traffic to eliminate the risk of others on that same network viewing your network traffic. Because a VPN does this, it is very effective in improving security on wireless networks (“Wireless Security and VPN,” 2001).

Step 5: Measure and Report Network Traffic Statistics for the Computers on Your Network That Are Using the Most Bandwidth

Measuring the number of bytes a computer sends and receives to the Internet can help you identify computers that have been compromised. Often, computers that are compromised on campus are used to store large data files (for example, copyrighted music, videos, or software) for others to download. When this happens the computer that was compromised will normally experience a much higher volume of network traffic than normal and will often become one of the largest users of the network. Reviewing the list of top “talkers” for computers that are not normally so active can offer indications that a machine has suffered a security incident (Dunn, 2001).

Although none of these steps by themselves will guarantee security, collectively they provide a good starting point for improving campus network security. As we shall see next, once a computer has been compromised it can be used for a variety of dangerous practices.

Host-Based Security

A computer, often referred to as a *host*, is often the target of hackers. Once a computer has its security compromised, a number of bad things can happen: the computer can be used as file storage for groups sharing illegal material, sensitive information stored on the computer (such as Social Security numbers or credit card information) can be accessed and released, the host may be used as an intermediary to probe other machines for security flaws, or the machine may be used to launch an outright attack on other systems. Because

they are often targets, securing the computers—that is, host-based security—is an important part of our IT security architecture.

Universities are often required to have networks that are much more open than other types of organizations to allow collaboration and access by students and faculty from off campus. As a result, computers connected to our campus networks are often more susceptible to hackers than computers in corporate networks. In tests at the University of Maryland, Baltimore County (UMBC), that have been confirmed in similar tests by other universities, we have attached machines running standard versions of Linux and Windows 2000 on our network and timed how long it took for the machine to have its security compromised. In all of the tests, the machines had their security compromised within the day; in fact, often this happened within hours! This occurred because hackers believe higher education institutions are easy targets and probe university networks for computers with security vulnerabilities.

Fortunately, host-based security can be accomplished through good system administration practices, such as maintaining up-to-date virus protection, making certain that the operating system software is configured properly, and ensuring that all of the latest security patches are installed. The challenge is that most campuses have thousands, if not tens of thousands, of computers on campus—most controlled by individuals outside of the central IT organization with little or no training in good system administration practices. I next discuss practices that institutions should promote to enhance host-based security.

Step 1: Establish Virus Protection with an Automated Update Service on All Critical Systems

Computer viruses and worms were the most common security problem during 2000–2002 (Briney, 2002). Although viruses can be written for any operating system, most are written to reach the widest audience and thus exploit security flaws in Microsoft prod-

ucts (Word, Excel, Internet Explorer, and the various versions of Windows). Because these products are among the most heavily used at universities, establishing virus protection on computers using Microsoft products is critical.

New viruses can spread very rapidly; it is important to select a virus product that will allow you to get frequent, automated updates to the virus protection software. Most virus protection products provide a version of their product that can be centrally managed by the institution. This allows the institution to automatically update all computers running the virus protection software at one time. Although this option is more expensive, without this automatic update a virus may strike and do considerable damage before people have updated their virus protection software. Because today's viruses spread through the Internet, by e-mail, and through the Web, they can quickly spread on campus. During one particular virus and worm outbreak, the NIMDA worm, UMBC measured 200,000 NIMDA virus probes from off-campus in one day ("CERT Advisory," 2001).

Step 2: Perform a Risk Assessment to Identify the Most Important Computers to Protect

Almost all institutions have more computers than they can properly protect. In designing a host-based security plan, the first step is to perform a risk assessment (see Chapter Three) to determine which hosts are the most important to protect and to focus first on those computers. In general, this will include computers that provide critical IT functions such as administrative systems, course management systems, e-mail, and Web servers. It should also include computers that contain sensitive information that needs to be protected, such as staff computers used in departments such as the bursar or registrar's office.

Finally, safeguarding research computers used by faculty may be very important as well. Prioritize the computers to protect by risk to the institution.

Step 3: Use a Network Scanning Utility to Create a Profile for Each Computer Identified in Step 2

In this step you create a profile of each computer you identified in step 2, showing the operating system and the different services accessible through the network.

Generally, each network service on a machine is associated with a specific TCP/IP port number (for example, Telnet is port 22, e-mail is port 25, and so on) (Postel et al., 2003). At a small institution it may be possible to examine the machines individually and get this information, but most campuses will want to use an automated tool to detect this information.

Commercial tools such as the Internet Scanner from ISS (“Internet Security Systems,” 2003) or public domain software such as Nmap (“Nmap—Network Mapping Software,” 2003) can be used to classify machines by operating system and the network services they are running. These tools work by scanning your network and looking for computers that respond. For each computer that responds, they check to see what network services are running and attempt to identify the version of the software. They can also be configured to look for and report known vulnerabilities for each computer.

Step 4: Disable the Network Services That Are Not Needed on the Computers Identified in Step 3; Consider Running a Host-Based Firewall on Your Computer to Block Unwanted Network Traffic

The default configuration for many operating systems is to have the most-common network services enabled. As a result, most machines are running network-based services such as a Web server, database server, or file sharing services that might not be necessary. One good tool for analyzing your system is the CISEcurity toolkit developed by the Center for Internet Security (“The Center for Internet Security,” 2003). This toolkit is easy to use and analyzes your system for potential security concerns against different baseline configurations. By disabling unnecessary network services on a computer, you

eliminate potential security problems associated with that service that could jeopardize the entire computer.

One newer solution that is gaining favor is to implement host-based firewalls. A *host-based firewall* is software that runs on each computer and is analogous to a network firewall, but it protects a single computer. It requires network traffic coming to the computer to meet certain rules before it is processed (Gwaltney, 2001).

During the next few years, many predict that host-based firewalls will play an important role; however, at present they can be problematic in that they can generate many time-consuming false alarms. Until vendors provide better configuration management capabilities so these can be run from a central place, they will be difficult to deploy across the enterprise. However, using these judiciously for machines that require additional protection may be a viable choice today.

Step 5: Monitor Security Alerts and Develop Mechanisms for Quickly Patching Systems

Dozens of security alert services are available to track security problems. At UMBC, we use the Bugtrak mailing list to track security alerts (“Bugtrak Mailing List Archive,” 2003). It is critical that some staff member(s) be assigned to monitor these security alerts. Once a security alert is announced, you can consult your computer profiles generated in step 3 to see what critical machines are vulnerable and work to get the security patch installed on those machines.

If the machines you are tracking number in the thousands, you must look at tools that can help automate the process of updating the machines. Many free as well as commercial tools are available that can assist with this task. The important thing is to make certain your staff has a plan for updating these machines rapidly when a security alert is announced.

One response to security alerts used at many schools is to reset their border firewall to block off-campus access to certain network services if it is believed that many machines will be vulnerable to a new threat until the staff can patch all of the machines susceptible

to that problem. Although this may have an impact on some off-campus usage, it may be preferable to letting the machines have their security compromised and dealing with all the consequences.

Step 6: Create a Centralized System Logging Service

All major operating systems provide support for system logging. These system logs record each time a network service is accessed and the success or failure of that access. Usually the record contains a time stamp, some identifying information, and the network service accessed. By default, these system logs are written to the local disk on the computer providing that network service; however, you can configure most systems to also write their logs to a central server via the network.

By centralizing the system logging service, a security officer can accumulate systems logs from hundreds of machines and look at patterns of unusual activity across those machines. An additional benefit of central logging is that if a machine is compromised, the log entries leading up to that compromise will not be lost. This can be very important when examining the cause of a security compromise and looking for other computers that might be affected. Clear policies and procedures regarding the capture, retention, and use of system logs are essential to protect the privacy of those using the systems.

Step 7: Develop a Central Authentication Service to Replace Host-Based Password Files

Host-based password files are notoriously insecure. Invariably users choose passwords that are associated with words or people, things often found in a dictionary. Although most operating systems encrypt the password files, the encryption algorithms are well known. Simple tools are available that allow hackers to go through a dictionary of words and compare the results of encrypting that word until a match is found against the encrypted password. These tools, such as L0phtCrack (Semjanov, 2003), make it easy to gain many user passwords once a machine is compromised.

Developing a centralized security service, such as Kerberos (Kohl and Neumann, 1993), removes user passwords from each

machine and eliminates the ability of someone to decrypt the password files stored on the local computer. Kerberos is available for most versions of UNIX, Linux, Macintosh OS/X, and Windows 2000/XP and is free.

The Role of Middleware and Campus Directories

The Internet2 Middleware working group defines *middleware* as a layer of software between the network and the applications (Klingenstein, 2003). This software provides services such as identification, authentication, authorization, group membership, and security. Middleware provides the linkage, or “glue,” among individuals, hosts, networks, and the applications deployed. In this section, I discuss how middleware facilitates security and is a key component in campus security architecture.

In the past year magazines such as *InfoSecurity*, *Information Week*, and *Network Computing* have all listed “identity management” as one of the key challenges facing organizations (Yasin, 2002). *Identity management* provides automated mechanisms for managing accounts: creation, deactivation, and deletion. Identity management also supports the varied roles that people have in higher education. For instance, I can be a staff member teaching a course and also taking a course and thus be a member of the staff, faculty, and student groups. The key to identity management is building an enterprise directory linked to your campus business systems: student, human resources, alumni, and admissions. The enterprise directory provides authentication services (Am I person X?) and facilitates authorization information (Am I a member of group Y that has the authority to use service Z?). Often the authentication component of the enterprise directory is linked to an existing authentication service, such as Kerberos, if one is available for use. If not, the directory can provide authentication services. It is critical that the security of the campus directory itself be managed very carefully.

The Internet2 Middleware initiative developed a business case for implementing middleware in higher education. This document

identified twenty-four uses and applications that were facilitated by the existence of middleware (Barton and others, 2001). More than half of the applications were related to network security, authentication, or controlling authorized use of resources, including portals, VPN access, wireless authentication, and self-service network registration for residential students.

One of the most basic and important security challenges every institution faces is managing user accounts and passwords. Without a directory, a member of the institution can end up with numerous usernames and passwords. When people have multiple accounts, this creates frustration and often leads to poor passwords (passwords that can be easily guessed through a dictionary attack as discussed earlier). For the institution, removing access for an individual when he or she leaves the campus is a tremendous challenge because you have to remove that individual from dozens of application-specific password files. Having these applications use the enterprise directory for authentication provides a single authoritative source for authentication across applications. In the event the individual leaves the institution or you must disable an account for some reason, this can be done in one place, the enterprise directory.

From these examples it should be apparent that middleware is an essential piece of our security architecture. It can also greatly facilitate the development of portals, enterprise resource packages, Web-based services, and so forth by centrally managing identities. Campuses beginning these projects should look at creating a middleware environment that furthers their security architecture in addition to meeting the needs of that project.

Applications and Central Services

A common, but critical security problem today is that many applications and services still send usernames and passwords unencrypted over the network, where they may be captured by hackers who have broken into another computer on the network. As a

result you have to assume that any person that used that service has a compromised password. At the least, all users must be contacted and required to change their passwords (“San Diego Super Computer Advisory,” 1997).

Common, everyday services that send unencrypted passwords include e-mail, Telnet (provides user access to remote computers), and FTP (transfers files from one computer to another). For most institutions, e-mail is the most heavily used application on campus. If your central e-mail servers have their security compromised, the passwords of thousands of people can be found in just one day.

Solutions have been available for a few years that provide these services by sending encrypted passwords over the network. Although changing software configurations is a major effort in user education, every campus should be working toward replacing these common applications with their “secure” counterparts, as shown in Table 6.1. (A good example is the “University of Colorado Encrypted Authentication Standards,” 2003.)

Another source of security problems is Web-based applications that maintain separate usernames and accounts for each user or that don’t utilize encryption for sending information from the users’ browser to the Web server (“The OpenSSL Project,” 2003). In some cases these Web-based applications use the same username that is used by campus servers but maintain separate password files.

Unfortunately, many people will use the same password for all of these applications without understanding that many of these applications don’t have strong security. The best solution is associated with middleware: develop a campus-based Web authentication

Table 6.1. Unencrypted Versus Encrypted Applications.

Unencrypted Application	Encrypted Application
Telnet	Secure Shell (SSH)
E-mail	E-mail over Secure Sockets Layer (SSL)
FTP	Secure Copy (SCP)

system that uses the enterprise directory, referred to as a Web initial sign-on (WebISO). By developing a WebISO, Web-based application developers can leverage the enterprise directory and use one central source for authentication. The Internet2 Middleware initiative has software available for institutions that want to develop a WebISO on campus (Dors, 2003).

As we look to the future, we can see that distributed security across multiple institutions will become increasingly important. This is already an issue for scientists using the national supercomputer centers funded by the National Science Foundation and will become an issue for access to online content providers used by our libraries. We are reaching the limits of what we can expect people to handle when it comes to accounts and passwords. Much of the time all of the information we need is an assertion from a trusted party that someone is still an active member of the same community.

Two technologies coming out of the Internet2 Middleware initiative, Shibboleth (Cantor, 2003) and OpenSAML ("OpenSAML," 2003), are designed to help support assertions of trust between institutions without the risks of application-based passwords. Such tools will be central components in the emerging technology of Web services.

Conclusion

Although this chapter touched briefly on a number of issues, it should be clear that IT security affects almost everything we do at our institution. If your institution is connected to the Internet, you can never be 100 percent secure. IT leaders, especially chief information officers (CIOs), play a critical role in developing their campus IT security architecture. CIOs need to work with their IT staff and other campus leaders to understand the local security risks and define priorities for their management.

Another leadership role of CIOs is to strongly encourage their entire staff to take an active and consistent interest in security.

Every CIO needs to ask his or her staff to prove how well they are doing in securing the institutional IT infrastructure. If no one knows the answers or cannot provide corroborating data, it is time to pull together your team and implement plans to answer them. Ask your team questions such as these:

- Who tracks security vulnerabilities?
- Who is responsible for making sure that machines with vulnerabilities get fixed? How do we know they actually did get fixed?
- How do we plan to secure wireless access?
- How do we protect ourselves from attacks that occur within our campus network?
- How many accounts and passwords do people have? Do we feel that people use good passwords?

Finally, the IT leader must find ways to incorporate security into the funding and implementation of both new and existing projects. Portals, enterprise resource planning, or course management systems are all major projects. Look for opportunities in their funding and implementation to enhance the security of the entire campus.

References

- Barton, T., and others. "Middleware Business Case." [middleware.internet2.edu/earlyadopters/draft-internet2-ea-mw-business-case-00.pdf]. Oct. 2001.
- Briney, A. "CYBER-Menace: Special Report on Growing Virus Problem." [www.infosecuritymag.com/2002/may/cybermenace.shtml]. May 2002.
- "Bugtrak Mailing List Archive." [www.securityfocus.com/archive/1]. Mar. 2003.
- Cantor, S. "Internet2 Shibboleth Project." [shibboleth.internet2.edu/]. Feb. 2003.
- "The Center for Internet Security." [www.cisecurity.org]. Mar. 2003.
- "CERT Advisory CA-2001-26 NIMDA Worm." [www.cert.org/advisories/CA-2001-26.html]. Sept. 2001.
- Dors, N. "Internet2 Web Initial Sign-on Project." [middleware.internet2.edu/webiso]. Mar. 2003.

- Dunn, J. "Security Applications for Cisco Netflow Data." [www.sans.org/rr/software/netflow.php]. July 2001.
- Fraser, B. "RFC 2196—Site Security Handbook." [www.faqs.org/rfcs/rfc2196.html]. Sept. 1997.
- Frasier, M. "Understanding Virtual Private Networks." [rr.sans.org/encryption/understanding_VPN.php]. Mar. 2002.
- Gray, T. "Firewalls: Friend or Foe." [staff.washington.edu/gray/papers/fff-final.htm]. Jan. 2003.
- Grimes, S. "IDS in the Trenches." [www.infosecuritymag.com/2002/sep/roundtable.shtml]. Sept. 2002.
- Gwaltney, R. "Protecting the Next Generation Network—Distributed Firewalls." [www.sans.org/rr/firewall/next_gen.php]. Oct. 2001.
- "Internet Security Systems." [www.iss.net]. Mar. 2003.
- Klingenstein, K. "Internet2 Middleware Initiative." [middleware.internet2.edu]. Mar. 2003.
- Kohl, J., and Neumann, C. "RFC 1510: The Kerberos Authentication Network Service V5." [ftp://ftp.isi.edu/in-notes/rfc1510.txt], also [web.mit.edu/kerberos.www]. Sept. 1993.
- "Nmap—Network Mapping Software." [www.insecure.org/nmap]. Feb. 2003.
- "OpenSAML—Open Source Security Markup Language." [www.opensaml.org]. Jan. 2003.
- "The OpenSSL Project." [www.openssl.org]. Mar. 2003.
- Postel, J., and others. "Internet Assigned Numbers Authority Port Assignments." [www.iana.org/assignments/port-numbers]. Mar. 2003.
- Roesch, M. "Snort." [www.snort.org]. Feb. 2003.
- "SAFE SQL Slammer Worm Attack Mitigation." [www.cisco.com/warp/public/cc/so/neso/sqso/worm_wp.htm]. Feb. 2003.
- "San Diego Super Computer Security Advisory." [www.attrition.org/security/advisory/misc/sdsc/97.05.caltech]. Sept. 1997.
- Semjanov, P. "Russian Password Crackers." [www.password-crackers.com/]. 2003.
- "University of Colorado Encrypted Authentication Security Standards." [www.colorado.edu/its/security/encauth]. Jan. 2003.
- "Wireless Security and VPN." [www.intel.com/ebusiness/pdf/prod/related_mobile/wp0230011.pdf]. Oct. 2001.
- Yasin, R. "What Is Identity Management?" *InfoSecurity Magazine* [www.infosecuritymag.com/2002/apr/cover_casestudy.shtml]. Apr. 2002.