International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 11 Issue 2 –NOVEMBER 2014.

THREATS TO ENTERPRISE NETWORK SECURITY

S.SeshaTalpaSai

Lecturer in Computer Science, Department of Computers, K.B.N.College, Kothapet, Vijayawada

sai327@rediffmail.com

Abstract— The principle goal is secure blueprint for enterprise networks (SAFE) is to provide best practice information to interested parties on designing and implementing secure networks. SAFE serves as a guide to network designers considering the security requirements of their network. SAFE takes a defences-in-depth approach to network security design. This type of design focuses on the expected threats and their methods of mitigation, rather than on "Put the firewall here, put the intrusion detection system there." This strategy results in a layered approach to security where the failure of one security system is not likely to lead to the compromise of network resources. SAFE is based on products and those of its partners.This document begins with an overview of the architecture, then details the specific modules that make up the actual network design. The first three sections of each module describe the traffic flows, key devices, and expected threats with basic mitigation diagrams. Detailed technical analysis of the design follows, along with more detailed threat mitigation migration and techniques strategies.

I. INTRODUCTION

Architecture Overview First and foremost, SAFE is a security architecture. It must prevent most attacks from successfully affecting valuable network resources. The attacks that succeed in penetrating the first line of defense, or originate from inside the network, must be accurately network. However, in being secure, the network must continue to provide critical services that users expect. Proper network security and good network functionality can be provided at the same time. The SAFE architecture is not a revolutionary way of designing networks, but merely a blueprint for making networks secure

Performance needs are not great, this document uses a complex design as an example because designing security in a complex environment is more involved than in simpler environments. Options to limit the complexity of the design are discussed throughout this document.

At many points in the network design process, you need to choose between using integrated functionality in a network device versus or when performance needs require using specialized hardware. Make your decisions based on the capacity and functionality of the appliance versus the integration advantage of the device. For example, sometimes you can chose an integrated higher-capacity IOS™ router with IOS firewall software as opposed to a smaller IOS router with a separate firewall. Throughout this architecture, both types of systems are used. Most critical security functions migrate to dedicated appliances becauseof the performance requirements of large enterprise networks.

II. MODULE CONCEPT

Although most enterprise networks evolve with the growing IT requirements of the enterprise, the SAFE architecture uses a green-field modular approach. A modular approach has two main advantages. First, it allows the architecture to address the security relationship between the various functional blocks of the network. Second, it permits designers to evaluate and implement security on a module by module basis, instead of attempting the complete architecture in a single phase. Illustrates the first layer of modularity in SAFE. Each block represents a functional area. The Internet service provider (ISP) module is not implemented by the enterprise, but is included to the extent that specific security features should be requested of an ISP in order to mitigate against certain attacks.

III. ENTERPRISE COMPOSITE MODULE

The second layer of modularity, which is illustrated, represents a view of the modules within each functional area. These modules perform specific roles in the network and have specific security requirements, but their sizes are not meant to reflect their scale in a real network. For example, the building module, which represents the end-user devices, may include80 percent of the network devices. The security design of each module is described separately, but is validated as part of the complete enterprise design.

While it is true that most existing enterprise networks cannot be easily dissected into clear-cut modules, this approach provides a guide for implementing different security functions throughout the network. The authors do not expect network engineers to design their combined with

understanding that VLANs and VLAN tagging protocols were not designed with security in mind, makes their use in sensitive environments inadvisable. When VLANs are needed in security deployments, be sure to pay close attention to the configurations and guidelines mentioned above.

Within an existing VLAN, private VLANs provide some added security to specific network applications. Private VLANs work by limiting which ports within a VLAN can communicate with other ports in the same VLAN.

Isolated ports within a VLAN can communicate only with promiscuous ports. Community ports can communicate only with other members of the same community and promiscuous ports. Promiscuous ports can communicate with any port.

This is an effective way to mitigate the effects of a single compromised host. Consider a standard public services segment with a Web, FTP, and Domain Name System (DNS) server. If the DNS server is compromised, a hacker can pursue the other two hosts without passing back through the firewall. If private VLANs are deployed, once one system is compromised, it cannot communicate with the other systems. The only targets a hacker can pursue are hosts on the other side of the firewall.

Hosts Are Targets

A host is the most likely target during an attack and presents some of the most difficult challenges from a security perspective. There are numerous hardware platforms, operating systems, and applications, all of which have updates, patches, and fixes available at different times. Because hosts provide the application services to other hosts that request them, they are extremely visible within the network.

In part because of the security challenges mentioned above, another, an operating system from still another vendor, and a Web server that is either open source or from yet another vendor. Additionally, the same Web server might run applications that are freely distributed via the Internet, and might communicate with a database server that starts the variations all over again. That is not to say that the security vulnerabilities are specifically caused by the multisource nature of all of this, systems. Keep any systems up to date with the latest patches, fixes, and so forth. In particular, pay attention to how these patches affect the operation of other system components. Evaluate all updates on test systems before you implement them in a production environment. Failure to do so might result in the patch itself causing a denial of service (DoS).

Networks Are Targets

The worst attack is the one that you cannot stop. When performed properly, distributed denial of service (DDoS) is just such an attack. As outlined in Appendix B, "Network Security Primer," DDoS works by causing tens or hundreds of machines to simultaneously send spurious data to an IP address. The goal of such an attack is generally not to shut down a particular host, but rather to make the entire network unresponsive.

For example, consider an organization with a DS3 (45 Mbps) connection to the Internet that provides e-commerce services to its Web site users. Such a site is very security conscious and has intrusion detection, firewalls, logging, and active monitoring. Unfortunately, all these security devices do not help when a hacker launches a successful DDoS attack.

Consider 100 devices around the world, each with DS1 (1.5 Mbps) connections to the Internet. If these systems are remotely told to flood the serial interface of the e-commerce organization's Internet router, they can easily flood the DS3 with erroneous data.

Even if each host is only able to generate 1 Mbps of traffic, (lab tests indicate that a stock Unix workstation can easily generate 50 Mbps with a popular DDoS tool As a result, legitimate Web requests are lost, and the site appears to be down for most users. The local firewall drops all the erroneous data, but by then, the damage is done.

Thwart such an attack. An ISP can con rate limiting on the outbound interface to the company's site. This rate limiting can drop most undesired traffic when it exceeds a prespecified amount of the available bandwidth. The key is to correctly flag traffic as undesired.

Common forms of DDoS attacks are ICMP floods, TCP SYN floods, or UDP floods. In an e-commerce environment, this type of traffic is fairly easy to categorize. Only when limiting a TCP SYN attack on port 80 (http) does an administrator run the risk of locking out legitimate users during an attack. Even then, it is better to temporarily lock out new legitimate users and retain routing and management connections, than to have the router overrun and lose all connectivity.

More sophisticated attacks use port 80 traffic with the ACK bit set so that the traffic appears to be legitimate Web transactions. It is unlikely that an administrator could properly categorize such an attack because acknowledged TCP communications are exactly the one approach to limiting this sort of RFC 2827 filtering is discussed in the "IP Spoofing" section of Appendix B, "Network Security Primer." For inbound traffic on a router that is connected to the Internet, you could employ RFC 1918 and 2827 filtering to prevent unauthorized traffic from reaching the corporate network. When implemented at the ISP, this filtering prevents DDoS attack packets that use these addresses as sources from traversing the WAN Collectively, prevent DDoS attacks, it does prevent such attacks from masking their source, which makes trace back to the attacking networks much easier.

Applications Are Targets

Applications are coded by human beings (mostly) and, as such, are subject to numerous errors. These errors can be benign—for example, an error that causes your document to print incorrectly—or malignant—for example, an error that makes the credit card numbers on your database server

available via anonymous FTP. It is the malignant problems, as well as other more general security vulnerabilities, that intrusion detection systems (IDSs) aim to detect. Intrusion detection acts like an alarm system in the physical world. When an IDS detects something that it considers an attack, it can either take corrective action itself or notify a management system for actions by the administrator.

Some systems are more or less equipped to respond and prevent such an attack. Host-based intrusion detection can work by intercepting OS and application calls on an individual host. It can also operate by after thefact analysis of local log files. The former approach allows better attack prevention, while the latter approach dictates a more passive attack response role. Because of the specificity of their role, host-based IDS (HIDS) systems are often better at preventing specific attacks than Network IDS (NIDS), which usually only issue an alert upon discovery of an attack. However, that specificity causes a loss of perspective to the overall network.

"If you're going to log it, read it." So simple a proposition, that almost everyone familiar with network security has said it at least once. Yet logging and reading information from over 100 devices can prove to be a challenging proposition. Which logs are most important? How do I separate important messages from mere notifications? How do I ensure that logs are not tampered with in transit? How do I ensure my time-stamps match each other when multiple devices report the same alarm? What information is needed if log data is required for a criminal investigation? How do I deal with the volume of messages that can be generated by a large network?

You must address all these questions when considering managing log files effectively. From a management standpoint, a different set of questions needs to be asked: How do I securely manage a device?

From an architectural point of view, providing out-of-band management of network systems is the best first step in any management and reporting strategy. Out-of-band (OOB), as its name implies, refers to a network on which no production traffic resides.

Devices should have a direct local connection to such a network where possible, and where impossible, (due to geographic, or system-related issues) the device should connect via a private encrypted tunnel over the production network. Such a tunnel should be precond to communicate only across the specific ports required for management and reporting. The tunnel should also be locked of secure socket layer (SSL) or secure shell (SSH), it should be preferred. SNMP should be treated with the utmost care because the underlying protocol has its own set of security vulnerabilities.

IV. CONCLUSION

When a network is under attack, it is important to know the state of critical network devices and Creating a plan for change management should be a part of your comprehensive security policy, but, at a minimum, record changes using authentication systems on the devices, and archive configurations via FTP or TFTP.

This first version of the SAFE architecture is meant to address the security implementation of a generic enterprise network. I think that there are many areas that need further detailed research, exploration, and improvement. Some ofthese areas include, but are not limited to, the following:

- In-depth security management analysis and implementation.
- Specialized design information for smaller networks. In-depth identity, directory services, AAA technologies, and certificate authority analysis and implementation Scaled versions of VPN head-end and WAN design.

V. REFERENCES

[1] RFCs RFC 2196 "Site Security Handbook" http://www.ietf.org/rfc/rfc2196.txt
[2] RFC 1918 "Address Allocation for PrivateInternets"– http://www.ietf.org/rfc/rfc1918.txt
[3] RFC 2827 "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing"
[4] http://www.ietf.org/rfc/rfc2827.txt