Global Information Assurance Certification Paper

A Cliff Notes Guide to the History of Information Security: Past, Present, and Future

By David J. Jackson

GIAC Security Essentials Certification (GSEC)
Practical Assignment Version 1.4b – Option 1

August 25, 2003

TABLE OF CONTENTS

Summary

References:

Abstract

Like many technologies that have advanced rapidly in our society to safeguard Federal computer systems and networks from improper use, the U.S. Government created computer and network security standards. Through the years this information has been disseminated to the public and to vendors and manufacturers of networking and security devices. This in turn has provided a means for companies with computer networks to maintain a wide range of security levels in the business environment.

This paper will try to examine and describe key points in history that critical events occurred in information security. It will also attempt to cover key aspects of Information Security and provide information that applies to information security in the past, present, and future.

The Beginnings of the Internet: The Arpanet

Computer networking and the Internet of today have progressed rapidly in 34 years. In 1958 the Department of Defense established an agency known as the Advanced Research Projects Agency (ARPA), which was initially given responsibility for the direction or performance of advanced projects given to it by the Secretary of Defense. One of those first projects was to create a communications medium, which consisted of 4 devices for electronic communication.

A research and development company called Bolt, Beranek and Newman or most commonly known as BBN, was given the task to design and implement the first node of the Arpanet in January of 1969 after bidding on the contract from ARPA. Many companies thought this would be impossible to design or implement. The original Request for Quotation (RFQ) was initially designed for four IMP (Interface Message Processors) devices, but by April of 1972, the Arpanet network soon grew to 23 sites. The first ever communication across the ARPANET was the word “LOGIN” typed from the first IMP installed at UCLA to the second IMP, which was installed at Stanford Research Institute (SRI).

The TCP/IP Suite

Vint Cerf and Bob Kahn created TCP/IP based on collaborative efforts. It was originally designed to make up for shortcomings in NCP (Network Control Protocol), which was the protocol that was originally used with the IMPs created by BBN. TCP/IP was designed to include error detection, packaging, and routing. Although it took some time, TCP/IP was finally accepted as the protocol of the Internet. On January 1, 1983, the ARPANET was fully switched over to

using only TCP/IP. The IPv4 version of TCP/IP created in the 1970s is still the standard protocol on the Internet to this day.

IPv6 fixes a number of problems in IPv4, such as the limited number of available IPv4 addresses. It also adds many improvements to IPv4 in areas such as routing and network auto-configuration.

Throughout the 1980s, several large companies were created and many of the critical parts of the Internet that we use today were designed. By 1987, Cisco, which was founded in 1983, shipped the first IP router. Additionally, by 1987, 25 million personal computers had been sold in the United States, and there were approximately 28,000 hosts on the Internet. By 1989 that number had grown to 133,000 hosts.

Viruses

A virus is a program that reproduces its own code by attaching itself to other executable files in such a way that the virus code is executed when the infected executable file is executed.

On November 2, 1988 a student from Cornell University named Robert Tappan Morris launched what many consider the first worm virus. Unlike the devastating worm viruses we battle today, this virus was initially written to spread using known vulnerabilities in UNIX. Unfortunately, a bug in the virus code itself caused more harm than initially anticipated by Robert Morris. A computer that was infected by the Morris Worm could be infected multiple times. Every additional process that was infected slowed the computer down to the point that it could not be used. The Morris Worm spread so rapidly, that approximately 6,000 UNIX computers were infected.

During that same month of November, 1988 the Computer Emergency Response Team Coordination Center (CERT/CC) was established by the Defense Advanced Research Projects Agency (DARPA) after the Morris Worm demonstrated the Internet's susceptibility to attack. Robert Morris was found guilty of violating the 1986 Computer Fraud and Abuse Act and was sentenced to three years probation, 400 hours of community service, and a fine of $10,050. The damages the Morris Worm caused were estimated between $10 million and $100 million.

The Cascade virus was originally written to trigger between October 1 and December 31, 1988. After an infected program was executed, characters on the screen dropped down into a pile at the bottom of the display.

By September of 1989, IBM commercially released version 1.0 of their virus-scanning program. Since the Cascade and Lehulpe viruses had infected IBM in

1988, their technical engineers thought it was a good idea to prevent customer's computers from crashing from these viruses. An extensive and intense research process began and several other companies followed IBM's lead and released anti-virus software of their own. Toward the end of 1989, there were only a couple dozen viruses that researchers knew about. By 1990, however, ideas in the virus creation world began flourishing. Mark Washburn created the first polymorphic virus. By definition, a polymorphic virus changes its code in an attempt to avoid detection by antivirus scanners. Essentially, the polymorphic virus encrypts itself in a different manner each time it infects, meaning that specific signature codes must be developed to search for each variety.

As the years have gone on, there have been several implementations of viruses on a wide-scale basis throughout the world. Anti-virus companies have invested billions of dollars in research and development to protect the common consumer from the growing insecurity of virus infection. As of the completion of this research, Symantec Norton Anti-Virus is currently able to detect and is protecting against 63,439 different viruses.

Although the Morris Worm is commonly considered the first worm virus, there have been many more since. The Melissa Worm was a computer worm that attacked vulnerabilities in Microsoft Outlook and Microsoft Word. On March 26, 1999 Internet email systems started to become overloaded with mail. The virus was initially posted to a Usenet newsgroup by means of a Microsoft Word document that contained a list of 80 pornographic websites. When launching the document, the worm was activated. It then sent a copy of itself to the first 50 entries in the Outlook address book of the infected computer.

As different variants spread throughout the Internet and were eventually contained, many more worms would come after. On July 19, 2001, a worm virus that exploited vulnerability in the indexing software distributed with Internet Information Server (IIS) from Microsoft made its way across the Internet in record time. It tried to spread itself by looking for other IIS servers connected to the Internet. Additionally, it defaced the infected website to display **"HELLO! Welcome to http://www.worm.com! Hacked By Chinese!"** It also waited 20-27 days after it had infected the host system, and then attempted to launch Denial of Service attacks against several target IP addresses. Among them was the IP address of the White House web server.

As we look at the world of viruses today we fast forward to 2003. January 24, 2003 a new worm made its way rapidly across the world infecting 75,000 computer systems in 10 minutes. CERT Advisory CA-2003-04 describes that the SQL Slammer worm virus exploits two known buffer overflow bugs in Microsoft's SQL Server, which is a database server. The SQL Slammer, although initially thought to just be a worm virus, turned into a Denial of Service attack as well. It's key process, after infecting a computer system without the appropriate patches, was to continually send traffic to randomly generated IP addresses, attempting to

send itself to hosts that were also running Microsoft SQL Server. The bandwidth consumed by such an attempt was devastating to the Internet and to companies all over the world including Microsoft themselves. On July 24, 2002, Microsoft released Microsoft Security Bulletin MS02-039. MS02-039 provides a description of the SQL vulnerability as well as a patch to fix what Microsoft describes as "Buffer Overruns in SQL Server 2000 Resolution Service" that "Could Enable Code Execution." Almost six months after this vulnerability and patch were released, still tens of thousands of computers were infected with the worm because users failed to patch vulnerable systems.

On August 11, 2003 CERT Advisory CA-2003-20 was issued in response to reports of widespread virus activity related to vulnerability in the Microsoft Remote Procedure Call (RPC) Interface. Hundreds of thousands of computers around the world were infected within hours with a virus that wreaked havoc on many networks. The W32/Blaster worm scans a random IP range to look for vulnerable systems on TCP port 135. The worm attempts to exploit vulnerability in the DCOM RPC Interface. The worm was designed to launch a TCP SYN flood denial-of-service attack against windowsupdate.com.

Why did this happen? Why were so many hosts infected? Microsoft released a Security Bulletin (MS03-026) on July 16, 2003, which describes the vulnerability and offers a patch for users of Windows NT 4.0, Windows NT 4.0 Terminal Services Edition, Windows 2000, Windows XP, and Windows 2003 Server. It wasn't however until August 11, 2003 that an exploit raised it's ugly head and attacked users of these operating systems that were not patched. The vulnerability and patch was published and 26 days later 145,264 computer systems from around the world were still infected.

On August 18, 2003, a new worm virus was unleashed exploiting the same vulnerability that the W32/Blaster worm exploited. The W32/Welchia worm was someone's attempt at helping patch computers with this vulnerability. The W32/Welchia worm kills and removes the msblast.exe file that was created by the W32/Blaster worm, and attempts to patch the vulnerable computer using the patch available in the Microsoft Security Bulletin MS03-026. However, because this worm does an ICMP scan of network computers to determine vulnerability, some experts believe this worm did more harm than good because it created a denial-of-service flood of ICMP traffic on the network.

On August 19, 2003, a new variant of a worm that was created back in January 2003 hit the web. W32/Sobig.F hit mailboxes fast. The CERT Incident Note IN-2003-03 describes the incident as "an email-borne malicious program with a specially crafted attachment." Oddly enough, the worm is believed to have a programmed shut down date of September 10, 2003, at which time it is expected to stop propagating. As Virus Definitions were released blocking this virus, many users did not update and subsequently were infected.

As viruses have grown more powerful over the years, it has been increasingly important not only for companies and business owners to have virus protection, but with the increase number of high-speed data lines installed inside homes today, it is just as important to have anti-virus software installed and updated constantly. This is where most companies and consumers have gone wrong in the past. They've purchased the anti-virus software with the hope that it will protect their data from being infected or in some cases destroyed, but they have not been dedicated to keeping virus definitions updated. This, in many instances, is because up until recently Anti-virus software manufacturers did not make it easy enough for a normal consumer, let alone a business owner with several hundred to thousands of computers, to update the virus definitions with ease.

However, with the increasingly sophisticated nature of virus detection software, so too comes extremely sophisticated viruses. In less than 10 days in August 2003, the Internet and hundreds of thousands of computers were maliciously attacked and infected and there was nothing anyone could do besides pull the plug.

Security in the Federal Government

The Computer Fraud and Abuse Act (CFAA) of 1984 was initially written to exclusively cover computer crimes that crossed state boundaries. The key areas of the act originally provided that it was a crime to access classified information or financial information in a federal system without authorization. It also established that it was unlawful to access a computer used exclusively by the federal government without authorization, use a federal computer to commit fraud, cause malicious damage to a federal computer system in excess of $1000, or to modify medical records in a computer that could impair proper treatment of an individual.

The Computer Security Act of 1987 laid out guidelines for fulfilling the responsibilities of maintaining a Federal computer system. Two of the key purposes of the Computer Security Act of 1987 were to identify computer systems that held sensitive information, and to create a Security Plan, which outlined the operation of each computer system under its control and to calculate the risk of loss, misuse or unauthorized access to the information on each system. Each federal agency was given the task to report their findings within 6 months. Additionally, they were given 1 year after the enactment of the Computer Security Act to submit their security plan to the National Bureau of Standards and the National Security Agency so that those agencies could utilize their expertise in evaluating each plan and give advice as to the functionality of the plan.

The NSA/NCSC (National Computer Security Center) Rainbow Series is a set of books, which give the basis of evaluating "Trusted Computer Systems" according to the National Security Agency. The main book of the series is the Orange Book, which was created on September 30, 1987. The Foreword of this book states "The guidelines defined in this document are intended to be used by computer hardware and software designers who are building systems with the intent of meeting the requirements of the Department of Defense Trusted Computer System Evaluation Criteria, DoD 5200.28-STD."

As these guidelines and policies changed over the years, the National Institute of Standards and Technology (NIST) and the National Security Agency (NSA) created CCEVS. The Common Criteria Evaluation and Validation Scheme is meant to be used as the basis for the evaluation of security properties of IT products and systems. The following are the objectives of this scheme:

- To meet the needs of government and industry for cost-effective evaluation of IT products;
- To encourage the formation of commercial security testing laboratories and the development of a private sector security testing industry;
- To ensure that security evaluations of IT products are performed to consistent standards; and
- To improve the availability of evaluated IT products.

As new products are created in the IT industry, we will see many more conforming to Government approved security standards as many businesses base their networks on a similar structure.

Encryption

The National Bureau of Standards, as well as the National Security Agency, developed data Encryption Standard or DES in the 1970s. IBM designed the first submission of the DES algorithm in 1976. LUCIFER as they called it, became a standard in the federal government. Because it uses a 56-bit key to encrypt and decrypt information, it can be brute force attacked and decrypted fairly quickly. As a replacement of DES, Advanced Encryption Standard or AES, has become a popular standard of many organizations but most notably, the Federal Government. AES operates with 128, 192 or 256 bit keys, which seem almost impossible to break with the computing power we have today. It could take millions of years to crack this level of encryption.

Pretty Good Privacy or PGP is a program used to encrypt and decrypt e-mail over the Internet. It can also be used to send encrypted digital signatures that let the receiver verify the sender's identity and know that the message was not changed during routing. It was written by Phillip Zimmermann as a response to

Senate Bill 266, which had a measure contained within it that stated that all encryption software must have a back door built in. He interpreted this to mean that private citizens would be banned from secure communication, so he wrote PGP and gave the program to a couple of friends to test and use.

Suddenly it appeared on Bulletin Board Systems (BBS) and was downloaded from everywhere including outside the United States, which began a legal battle. U.S. Export Laws prevented such programs from being exported out of the United States, so an investigation began into export laws that Zimmermann had allegedly violated. After 3 years, on January 11, 1996, the U.S. Attorney in San Jose announced the closing of the grand jury investigation of Philip Zimmermann. Nobody knows exactly why the government dismissed the case against Zimmermann, however some speculate that the government either had too much difficulty in proving that Zimmermann had done anything to violate the export laws, or that the NSA had cracked PGP and didn't need to worry about it anymore.

Passwords

Passwords are used to restrict access to systems or to protect sensitive or important data. Unfortunately, many companies do not enforce strong passwords and use passwords as their only line of defense. Passwords should be part of the Defense-in-Depth concept where they are used as part of the defensive strategy for the security in your network. Requiring and enforcing Strong Passwords only helps protect that level of defense.

On April 12, 1985, the Department of Defense released a document on Password Management Guidelines that was referred to as the "Green Book." This book detailed suggested steps that should be taken to assure that user-based password authentication policies should be followed and that certain requirements should be met when setting and enforcing passwords. Some of the ideas advocated in this document are:

- Users should be able to change their own passwords and not have the passwords changed for them to maintain the sensitivity of the password.
- Passwords should be generated by the machine rather than being created by the user.
- Certain audit reports like the date and time of the last login should be provided by the system directly to the user. Auditing user authentication is a great way of tracking when users accessed the system.

It is also recommended that the system security officer should be permitted to change the password of any user by generating a new one. This is especially important if the user has forgotten their password. Some simple guidelines to follow when creating password policies are:

- Password length should be set to a specific range of characters such as 6 to 8 characters.
- Setting a password expiration period, like 90 days, helps keep the password fresh.
- Enforce strong passwords so the password cannot be easily guessed.
- Make clear the sensitivity of each user's password. Passwords should not be given out to anyone. A system security officer should not need to login as the user. They should have sufficient rights to perform whatever task needs to be performed for the user.
- Enforce the history and uniqueness of passwords. Passwords should not be able to be repeated each time password changes are enforced. Using the Defense-In-Depth strategy, password history should prevent users from using the same passwords over again. This can be set at a minimal setting like the previous 4 passwords, but to increase the strength and reliability of user-based authentication, it should be set to a much higher number like 26 passwords.

Firewalls

There are typically 3 types of firewalls in use today. They are Stateful Packet Inspection, Packet Filter, and Application Proxy.

A packet filter firewall is the simplest type of firewall. Dealing with each individual packet, the firewall applies its rule set to determine which packet to allow or disallow. A packet filter firewall uses Source IP address, TCP/UDP source port and the Destination IP address, TCP/UDP destination port to fire off rules to allow or deny specific criteria. This type of firewall is extremely fast, however, not extremely reliable since it doesn't detect if other data intended to do harm to a system tries to exploit a specific port number that is wide open on the firewall. It just simply lets the data come through since it's only criteria is the source and destination IP addresses and ports.

A Stateful Packet Inspection Firewall examines the contents of packets rather than just filtering them. Firewalls of this type use an inspection method that understands data in the packet intended for other layers, from the network layer to the application layer. They also take into account the state of the connections they handle so a legitimate incoming packet can be matched with the outbound request for that packet and allowed back in through the firewall. Not the greatest performance firewall available, Stateful Packet Inspection Firewalls make up for performance with reliability of security and enhanced logging capabilities. Because each packet is inspected, the firewall must be required to perform many more duties than a simple packet filtering firewall. Many times, these firewalls are a more expensive implementation because of the enhanced hardware needed to operate and perform at a higher speed.

An Application Proxy is a program running on the firewall that emulates both ends of a network connection. Acting as a gateway, each computer communicates with the other by passing all network traffic through the Application Proxy. It then analyzes the data from the sender and decides which to pass on and which to drop. Because this type of firewall does not allow communication directly to endpoints through it as Packet Filtering and Stateful Packet Inspection firewalls do, it is a much more secure method of transmitting data. However, being that it needs a proxy setup for each protocol it is filtering, it can not only be extremely time consuming to setup, but also could require more significant changes to how an end-user gets through it and out to the web.

There are also many personal software firewalls available today. These, however, are not as reliable as a hardware implementation because they rely on an operating system to function. If a hacker has the ability to exploit something in the OS, then the abilities of the software firewall can be dramatically altered or removed entirely. The use of firewalls is increasing daily by consumers, but for many it took a system compromise to put one in place.

Vulnerability Scanning

Vulnerability scanning tools have been around for many years. One of the most notable tools was the Security Administrator Tool for Analyzing Networks or SATAN. A Security Bulletin dated April 3, 1995 published by the CERT/CC warns that even though the creators of SATAN state it's use is for System Administrators, it potentially could be used to help individuals obtain unauthorized access to networked systems. SATAN was most notably the first widely available security vulnerability scanning tool. With its WWW client interface, it was very simple for a user to enter in a machine address or domain to begin scanning and probing hosts for services and programs being used on the network. Depending on the level of reporting the Administrator chose (Light, Normal, or Heavy,) the host would be scanned more aggressively for vulnerabilities and services. A Light level would simply report which hosts were available on the network. A Normal level would probe the host targets by trying to establish communication using finger, telnet, FTP, WWW, gopher, and SMTP. A Heavy level of reporting, in addition to the Normal level, would search for other known vulnerabilities like trusted hosts available to each computer as well as anonymous FTP directories that were writeable.

War Dialing began in the 1980s and was an extremely popular form of penetration testing. A hacker would use his computer to dial randomly generated numbers in an attempt to determine if a modem responded. If so, many options existed that would allow the hacker to keep track of whether or not a human answered the phone or if it was indeed a handshake from a modem. If a modem

did answer, this would allow the hacker a potential entry into an otherwise protected network.

Many considered a program called ToneLoc the Swiss Army Knife of war dialers. Short for Tone Locator, it was an extremely popular tool to determine what phone carriers were in a specific prefix or to simply determine if a hole existed in a network of computers that could be exploited and “owned” by the hacker. Among the many features of ToneLoc was the ever-popular “G” Key during a scan. This would allow you to log the current number as a "Girl.” Today there are still reasons to use a War Dialer to perform Vulnerability Scanning on networks. Many system administrators are busy watching Firewall and Router log files, but very few think of a modem as a potential threat anymore. Also, since most computers do not come pre-shipped with modems anymore, it is almost impossible to imagine a modem being used in your office.

As software becomes more sophisticated, so do the vulnerability scanning tools. Because many software developers do not take measures to ensure the code they write is secure, many vulnerabilities exist in applications that are sold for the first time. As the community of consumers and security advocates use the software, vulnerabilities are detected and reported. Although many of these vulnerabilities may be fixed simply by applying patches, many more are unable to be fixed without a complete re-write of the code. Many security companies today insist that security vulnerability testing begin with the software developer and that software should not be released to the public unless it has been certified to be secure. Very few software companies take this stance.

In June, 2003, Microsoft Corporation released a statement that they were initiating a strategy called the Trustworthy Computing Initiative. This initiative introduced the following concepts as a result of significant feedback by consumers:

- **Secure by Design** – Products have secure architectures and features, and developers take extensive steps to avoid introducing security vulnerabilities into their code.
- **Secure by Default** – Products limit their exposure to attack, and disable features that are not very widely needed; and features run with minimum privilege to minimize the impact of even a successful attack.
- **Secure in Deployment** – Products are complemented by tools, guides, and training that help customers operate their systems securely.
- **Communications** – Microsoft provides customers with the information they need to help them operate securely – including responses to vulnerabilities from the Microsoft Security Response Center

Microsoft has also provided technologies such as the Windows Update Service, Office Update, and the Microsoft Baseline Security Analyzer (MBSA). Another notable patch management software tool that Microsoft has released free to

consumers, but mostly directed to companies of medium size, is Software Update Services (SUS). SUS assists administrators with the distribution of Security Fixes and Critical Updates released for Microsoft Windows 2000 and Microsoft Windows XP computers, which include Servers and Desktops.

Recently on July 9, 2003, Dell Computer Corporation released a statement that they would begin shipping a more secure or "hardened" configuration on desktops and laptops. Included in this new security service from Dell, over 50 security settings will be activated on Microsoft Windows 2000 computers. Based upon benchmark standards established by the Center for Internet Security, Dell is taking big strides to maintain its edge in the mainstream consumer market and to help keep consumer computers secured.

Wireless Security

Wireless security is currently a rapidly changing and evolving standard. In 1970, the world's first wireless packet switching network was in use. ALOHANET was developed at the University of Hawaii and using Radio transmissions, developers were able to connect the mainland to the ARPANET. The ALOHANET was connected to the ARPANET on the mainland in 1972. It was the first time another network was connected to the ARPANET.

Many other attempts at wireless have been achieved over the years, however not many are left standing or are being used as a standard. There is 802.11, HomeRF, IrDA, and Bluetooth. 802.11 is the industry standard wireless protocol. It was adopted by most of the networking manufacturers, and since the inititial design has been through many revisions to the protocol. Currently, 802.11 is the most widely used implementation of wireless networking. With it's speed capabilities and low price per user, it has become the product of choice for many. However, there are still many security issues that have yet to be worked out in the 802.11 protocol.

One of the first attempts at securing the extremely popular 802.11b was called WEP. Short for Wired Equivalent Privacy, in simplistic terms it attempted to secure the packets being transmitted by embedding a secret key in every packet sent across the airwaves. Of course, being encrypted it was initially thought that WEP would be secure, but as it was discovered and well documented in an article entitled "Weaknesses in the key Scheduling algorithm of RC4," this was not the case. Soon hackers had all the information they needed to begin to create programs that would crack WEP keys across the wire.

WPA, which stands for Wi-Fi Protected Access, is a new emerging standard. As some of the deficiencies of WEP keys were noticed, it became apparent that some other method of key distribution and management was needed. WPA uses a robust method of key distribution, which allows keys to be rotated at different

intervals so an attacker is not able to lock onto one specific key when sniffing the airwaves. AES or Advanced Encryption Standard was the government's replacement for an ailing DES (Data Encryption Standard). AES will be a built-in feature on many new wireless devices, but will also not work on some existing equipment. WPA has authentication support added to it by means of using a RADIUS (Remote Authentication Dial-In User Service) server. RADIUS is a protocol that is responsible for authenticating remote connections made to a system, as well as providing authorization to network resources and logging for accounting purposes.

Another form of built-in security to most Wireless Access points is MAC Address Filtering. This enables the Access Point to only allow computers with specific MAC addresses that it knows about, associate with it and use it's resources. Unfortunately, many do not realize that a MAC address is easily spoofed. Many experts say use all of the security features available in Access Points today to secure your wireless network.

Like War Dialing discussed earlier in this document, another prominent vulnerability scanning method has been invented specifically pertaining to wireless networks. War Driving consists of a few simple things:

- A computer, laptop, or pocket pc;
- A Wireless network card; and
- Specific software to help find Access Points and rogue wireless users.

Another term that is not as frequently used anymore, but was initially, is War Chalking. War Chalking consisted of someone scanning for an open wireless LAN and indicating that the wireless network was open or closed by writing with a piece of chalk on the sidewalk in front of the location, on walls at the building, or just simply in the street. This would be an indication to others in the area performing similar scanning whether a network was available to attach to or not.

Like War Dialing and modems connected to your network without your knowledge, wireless-networking vulnerabilities become even more burdensome and extremely difficult to detect. As a network administrator, it is your duty to know what devices are communicating on your network. There are many utilities available today, mostly freeware utilities written for the Linux operating system, that allow you to not only detect an access point or wireless user, but also determine what the SSID (Service Set Identifier) is and whether or not the AP is secured by WEP or not. War Driving has become an obsession for many. Using a laptop with a wireless card, external antenna, and GPS receiver allows many freedom of movement from one location to another by simply driving down the street and detecting open wireless networks. This could be extremely useful to a network administrator who is trying to do vulnerability testing on his wireless network to determine if there are access capabilities within a certain range.

Because of it's low cost and ease of use, wireless networking will be around for many years to come, and along with it will eventually come security that can be trusted and reliable.

Virtual Private Networks

A Virtual Private Network or VPN is a virtual, encrypted network that is built on top of an existing network. Using tunneling, the encrypted data stream is set up and maintained within a normal, unencrypted connection. It provides three basic functions; Sender Authentication, Message Integrity, and Message Confidentiality. A VPN extends the safe Internet network out to the remote user. A VPN is extremely versatile since it can be used over just about every connection method. In fact, prior to the creation of VPNs, companies used expensive point-to-point leased lines that were dedicated connections from one office to another.

Imagine needing to host an application on your internal network and requiring customers to have direct lines into your network from all over the world. That would be an extremely costly solution. With a VPN, you can virtually run cables all over the world to customers anywhere and still provide a similar secure method of transmission over an insecure line. Unfortunately, bandwidth is consumed very rapidly in low speed connections and is not always the best solution. VPNs, however, are an extremely useful resource and should be used whenever possible.

Summary

Information Security has had many faces throughout the years, many of which are not very pleasing to the public. Many news organizations throw the words "hackers" and "worms" around without knowing specifically what they have in common. As with many things in the Information Technology world, a little information can be extremely harmful in the hands of a novice. Viruses make their way out to the cyber world wreaking havoc on mail systems and firewalls, but, unfortunately, not all are made by hackers attempting to break in and steal your information. Disgruntled employees seeking revenge for being wronged write many viruses.

Movies tend to glamorize the ways of the hacker. With the information available on the web today, it's very easy for a 13 year old to become the next target of a federal investigation. Just as the epic battle between Black Hats and White Hats, it's what you do with the information that makes the difference.

Many experts in the industry believe that it is the software developer's job to make sure an application or operating system is secure before selling it to a consumer. As a systems analyst, my job is to make sure all of my servers are functional. Many think there is a black and white difference between functionality and security, but as my job is to make sure my servers are functional, some of the security is lacking in those systems. If you have 100% security you are more than likely going to have very little functionality. So what if a software developer creates a 100% secure operating system. Is it going to be functional? The problem is that the current idea of functional is based on existing applications and their abilities. If we start from scratch and build applications and operating systems that are extremely secure, consumers certainly will miss features in existing applications.

As viruses become commonplace, it is extremely important to make sure your network is secured by granting the least privilege possible for a user. The principle of least privilege requires that a user be given no more privilege than necessary to perform his or her job. Many times, this is why viruses and other hacking tools can be so damaging. If a user has full access to his or her computer, they become a prime target for any type of malicious activity. Users must be taught to be diligent and detect changes in normal activity.

So because the government creates standards and uses them, many people follow those standards and do nothing else. Unfortunately, the government is not the most secure organization in the world. There have been many documented examples of security protocols being broken or lack of security at specific government offices that would enable just about anyone to penetrate the security mechanisms put in place. It becomes more apparent that government security should be used as a baseline for security in any organization, but that depending on your structure you may have to modify your security model. This may include stricter security policies or the loosening of policy restrictions.

As the Internet evolves, there will be many more challenges ahead. From securing Internet access at your local fast food restaurant, to secure high-speed Internet access being installed in many more homes across the world, Information Security will always be a necessity. Dealing with mainstream Information Security for 45 years, it seems we've only scratched the surface.

References:

Brief Timeline of the Internet
http://www.webopedia.com/quick_ref/timeline.asp

Early Computer Security Papers
http://www.isse.gmu.edu/~csis/history/

About the Internet
http://www.ryerson.ca/acs/usersguide/internet.html

BBN Timeline
http://www.bbn.com/timeline/

Polymorphic Virus (Definition)
http://antivirus.about.com/library/glossary/bldef-poly.htm

Online Firewall Buyer's Guide
http://www.icsalabs.com/html/communities/firewalls/buyers_guide2001/

Wireless LAN Security: A Short History
Matthew Gast
(April 19, 2002)
http://www.oreillynet.com/pub/a/wireless/2002/04/19/security.html

FLUHRER, S., MANTIN, I., AND SHAMIR, A. Weaknesses in the key scheduling algorithm of RC4. Eighth Annual Workshop on Selected Areas in Cryptography. (August 2001)
http://citeseer.nj.nec.com/fluhrer01weaknesses.html

War Chalking
http://www.warchalking.org/

History of Wireless
http://wireless.jhsph.edu/history.html

ALOHANET (Definition)
http://www.wikipedia.org/wiki/ALOHAnet

Advanced Encryption Standard (AES) Information from NIST
http://csrc.nist.gov/CryptoToolkit/aes/

[CSC-STD-002-85]
Green Book
(April 12, 1985)
http://www.fas.org/irp/nsa/rainbow/std002.htm

NSA/NCSC Rainbow Series Related Documents
http://www.fas.org/irp/nsa/rainbow.htm

[CCIMB-99-032]
Common Criteria for Information Technology Security Evaluation
(August 1999) Version 2.1
http://commoncriteria.org/cc/cc.html

Microsoft Security Bulletin MS03-026
(July 16, 2003)
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS03-026.asp

What You Should Know About the Blaster Worm and Its Variants
(August 22, 2003)
http://www.microsoft.com/security/incident/blast.asp

CERT® Advisory CA-2003-04 MS-SQL Server Worm
(January 25, 2003)
http://www.cert.org/advisories/CA-2003-04.html

CERT® Advisory CA-2003-20 W32/Blaster worm
(August 11, 2003)
http://www.cert.org/advisories/CA-2003-20.html

CERT® Incident Note IN-2003-03
(August 22, 2003)
http://www.cert.org/incident_notes/IN-2003-03.html

Last Updated: July 25th, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event