Computer Forencics

S Jaya Prakash

2016115033

Assaignment on lesson 1

Security Threats

The Focus of This Paper

The purpose of this white paper is to help administrators, computer security officials, and others to understand the importance of computer security and the responsibilities it involves. The document provides a discussion of general security threats and how to plan and implement security policies and controls for often-performed computer security activities.

Security Overview

Background

Any organization that has a computer system and sensitive information wants to protect that information.

This section of this paper focuses on the background of security. It also looks at the importance of planning for possible threats and defining policies to limit the vulnerabilities that exist in a system and its security policies.

The greatest threat to computer systems and their information comes from humans, through actions that are either malicious or ignorant. When the action is malicious, some motivation or goal is generally behind the attack. For instance, the goal could be to disrupt normal business operations, thereby denying data availability and production. This could happen between two rival companies or even as a hoax.

COMPUTER CRIME

Computerization significantly eases the performance of many tasks. For example, the speed and ability to communicate with people is fostered by the Internet, a worldwide network that is used to send communiqués and provide access to the world-wide web. But this same speed and ability to communicate also opens the door to criminal conduct. Computer crime plays a significant role in the criminal law of the information age. Accompanying the influx of computers is an increase in criminal

acts and, as a result, an increase in the number of statutes to punish those who abuse and misuse this technology.

Computer crime, sometimes known as cyber-crime, is a serious concern. The crime can be perpetrated instantaneously and its effects can spread with incredible quickness. Furthermore, the ever-increasing use of computers, especially in serving critical infrastructure, makes computer criminality increasingly important.

The computer as a communication tool presents the computer as the object used to commit the crime. This category includes traditional offenses such as fraud committed through the use of a computer. For example, the purchase of counterfeit artwork at an auction held on the Internet uses the computer as the tool for committing the crime. While the activity could easily occur offline at an auction house, the fact that a computer is used for the purchase of this artwork may cause a delay in the detection of it being a fraud. The use of the Internet may also make it difficult to find the perpetrator of the crime.

computer can also be the target of criminal activity, as seen when hackers obtain unauthorized access to Department of Defense sites. Theft of information stored on a computer also falls within this category. The unauthorized procuring of trade secrets for economic gain from a computer system places the computer in the role of being a target of the criminal activity.

The five stages of a cyber intrusion

When it comes to cybersecurity, the most egregious breaches often come down to human error, such as someone clicking on a link in a spoofed email. That’s why officials try to emphasize the importance of good cyber hygiene and educating the work force on best practices.

With October being National Cyber Security Awareness Month, defense and civilian agencies have been trying to bolster public and personnel understanding of cyber risks. The Navy took a similar step recently, releasing a detailed list of the five stages of a cyber intrusion.

STAGE 1: RECON

During this stage, adversaries will begin to learn as much as possible on the potential target, its network, systems, personnel, logistics and warfighting capabilities. Through various virtual techniques – which have proven the most effective and the least risky – adversaries will begin to deploy measures aimed at acquiring information.

Social engineering and complacency. Attackers rely on human laziness to trick unsuspecting victims to surrender personal or confidential information enabling access to data without inside knowledge. This can be achieved by getting them to visit a bunk Web page or plug an unauthorized device with malicious code into a computer connected to the network.

Phishing:

The most common tactic, in which adversaries send emails to victims masking their identity to appear to be from a trustworthy source. The email often contains information that requires the user to click on a link or open an attachment that contains malicious code. Generally, the email will insist some type of urgency, thus further enticing victims to fall for the trap, again, relying on laziness and haste.

Watering-hole:

Adversaries profiling the websites and social media outlets typically used by certain individuals, then wait for targeted individuals to visit, upon which they will be redirected to another site with implanted malware. Often, victims do not even know that their computer, and their network, is infected.

STAGE 2: INTRUSION AND ENUMERATION

At this point, the adversary has already gained access to the network and will now blend in with the network's traffic and look for desirable information to exploit or deploy cyber tools that might inflict greater, more destructive damage.

STAGE 3: MALWARE INSERTION AND LATERAL MOVEMENT

Attackers will begin to open additional channels to access the compromised network, deploying software such as remote access Trojans, or RATs, also called backdoors. Adversaries will move laterally on the network, implanting software that can give them more privileges and then access mission-critical information, sensitive data, intellectual property and/or warfighting/platform control systems. It can take years to discover the scope of these intrusions, the Navy said.

STAGE 4: DATA EXFILTRATION

Having gained deep access to the network, the adversary can now remove data from systems. The Navy notes that most information is encrypted, but that it can be decrypted. Breaking encryption is generally a time-consuming and challenging undertaking, but hackers who have gotten this far are likely up to the task.

STAGE 5: CLEAN UP

Lastly, adversaries will leave, sometimes cleaning up after themselves before they go. If they're not concerned about the hack being detected, they might just disconnect from the network. More sophisticated actors, however, might erase their presence on the network, leaving behind back doors they could use later. Or, they could delete or manipulate data – something the director of national intelligence has warned could be the next big cyber incident.

There are plenty of reasons to educate the Defense Department's workforce on cyber best practices. The Navy has said that the Defense Department faces 41 million scans,

probes and attacks per month, with Lt. Gen. Alan Lynn, director of the Defense Information Systems Agency, saying at an event hosted by Defense Systems last month that, “Out of 700 million emails we’ll get in a month, only about 98 million are actually good emails.”

DEFINITION

Telecommunication fraud is the theft of telecommunication service (telephones, cell phones, computers etc.) or the use of

telecommunication service to commit other forms of fraud. Victims include consumers, businesses and communication service

providers.

TELECOMMUNICATION FRAUD

TYPES OF TELECOMMUNICATIONS FRAUD

A) **IDENTITY THEFT-** The misuse of information that is specific to an individual in order to convince others that the imposter

is the individual, effectively passing one self off as someone else.

B) **INTERNET FRAUD** - Any type of fraud scheme that uses one or more components of the internet - such as chat rooms, email,

message boards, or web sites to present fraudulent solicitations to prospective victims, to conduct fraudulent transactions,

or to transmit the proceeds of fraud to financial institutions or to others connected with the scheme.

C) **TELEMARKETING FRAUD** - Any scheme to defraud in which the persons carrying out the scheme use the telephone as

their primary means of communicating with prospective victims and trying to persuade them to send money to the scheme.

D) **AUCTION AND RETAIL SCHEMES** - These schemes typically attract consumers by purporting to offer high-valve

merchandise ranging from expensive jewelry to computers to sport memorabilia at attractive prices. After persuading victims

to send money in the form of a personal check, money order, or cashier's check, schemers either send an inferior item or

nothing at all.

E) **NIGERIAN MONEY OFFER SCAMS** - Potential victims receive, either through e-mail or fax, a request from a purported

high ranking Nigerian government official (with the title of Doctor, Chief, or General) seeking permission to transfer a large

sum of money out of Nigeria or some other African country into the victim's bank account.

F) **ATM FRAUD** - Use a special information storage device to secretly copy the magnetic strip on the back of credit and debit

cards during normal transaction such as an ATM withdrawal or in-store purchase (this is called skimming)

G) **INVESTMENT SCAM**S- Market manipulation scams are the forefront of this type of scheme. Two methods are used, the

first is commonly known as the pump - and dump, attempts to drive up the price of thinly traded stocks or stocks of shell

companies by sending out e-mails that inflate the value of the company. When new purchases of the stock push its price to a

high enough level, the scammers sell off the stock to realize a significant return which in turn drives down the stock price. The

second method, referred to as short-selling or scalping, tries to decrease a stock value.

Cyber terrorism:

• Terrorism is no longer bound by the means of creating harm in the physical world;

• Terrorism holds an agenda often, though not limited to, religious, cultural, social, economic and political;

• By definition cyber terrorism means to damage information, computer systems and data that result in harm against non-combatant targets;

• The boundaries between acts of cyber terrorism, cyber crime and ‘Hacktivism’ are often interlinked;

• Society faces a number of threats without our cyberspace, particularly to industrial control systems operating power grids and nuclear stations;

• Terrorist organisation are promoting the use of computing expertise to implement cyber attacks against targets;

• Though there are many organisations built to respond to cyber terrorism, a large amount of society is still unaware of the potential threat cyber terrorism poses;

• Systems are often developed without security in mind;

• Continually developing identification, tracing and mitigation methods to cyber terrorism is essential.

There is often a large amount of confusion as to what cyber terrorism is. More specifically, what cyber attacks can we actually define as acts of terrorism? The internet has allowed for a vast exchange of information. Thus has created a cyber space in which both criminals and terrorists can implement attacks/communications. This use of cyber space results in there no longer being simply a physical threat of terrorism. When we consider what cyber terrorism actually is, we must first understand the motivations behind cyber attacks. Cyber attacks can come in many differing forms, and it is these forms that help us understand whether the attack is of crime or terror. Figure 1 shows the distribution of cyberattacks across cultural, social, economic and political motivations. Gandhi et al. (2011) discusses that often these dimensions of motivations can often cross over and the motivating factors behind cyber attacks are needed to be carefully considered when we discuss cyber terrorism.

Need for Security

OS Security:

The Operating System can be described as the life of a computer system. It is the primary software component that is loaded into the system which allows the system to become operational and controllable. It manages all the programs and applications on the computer. Being the control centre of the computer, its role in the overall security of the system is paramount.

Computer Security is basically maintenance of system integrity, availability and confidentiality. The security within a computer system can be divided into various layers such as maintaining the physical security of the system, the security of the information the system holds and the security of the network in which it operates. In all these areas, the operating system plays a vital role in keeping the security.

How Operating System Maintains Security

Information Security

This entails security of all data, application and the operating system itself from attack. Threats may occur deliberately or due to error by humans, malicious programs or persons, or existing system vulnerabilities. The following measures highlight the role of the operating system in maintaining security of the information.

Authentication

Authentication is one of the protective method used by OS to ensure that the user accessing a program is authorised or legitimate. OS provides authentication using a number of techniques:

User names and passwords - these are names and passwords registered with the operating system to whom it allows access at the time of login.

Key Cards - these are physical cards programmed by the OS with unique identifying information that allows the user to login to the system.

User attributes - The operating system registers unique physical characteristics of the user (called attributes) to identify him at login. These may include fingerprints, signatures and eye retina patterns.

Backup and Restore

The OS has software modules that allows the user to take backups or make a copy of data and facilitate successful restoration of these backups whenever needed. These files may be stored off-site (which is the best security policy) or on-site. Backups can be:

File backups that entail backing up of data, files and folders associated with applications and programs. These can be saved within the system or to an external data storage.

System Image Backups that entail backing up of the OS along with programs, applications and files. It facilitates a successful restore of the entire system to its original state. This is necessary in the event the OS itself is corrupted or crashes or some irreversible disaster occurs due to physical threat. Image backup will ensure a complete roll back of the whole machine.

Database Security:

Use Web Application and Database Firewalls

Your database server should be protected from database security threats by a firewall, which denies access to traffic by default. The only traffic allowed through should come from specific application or web servers that need to access the data. The firewall should also protect your database from initiating outbound connections unless there is a specific need to do so.

In addition to protecting the database with a firewall, you should also deploy a web application firewall. That's because attacks such as SQL injection attacks directed at a web application can be used to exfiltrate or delete data from the database. A database firewall won't necessarily prevent this from happening if the SQL injection attack comes from an application which is an allowed source of traffic, but a web application firewall may. For more on SQL injection attacks, see How to Prevent SQL Injection Attacks.

Harden Your Database to Fullest Extent Possible

Clearly it's important to ensure that the database you are using is still supported by the vendor or open source project responsible for it, and that you are running the most up-to-date version of the database software with all database security patches installed to remove known vulnerabilities.

But that's not sufficient. It's also important to uninstall or disable any features or services that you don't need to use, and ensure that you change the passwords of any default accounts from their default values - or better still, delete any default accounts that you don't need.

Finally, ensure that all database security controls provided by the database are enabled (most are enabled by default) unless there is a specific reason for any to be disabled.

Once you have done all this, you should audit the hardened configuration -- using an automated change auditing tool if necessary -- to ensure that you are immediately aware if a change to the hardened configuration is made that compromises your database security.

Encrypt Your Data

It is standard procedure in many organizations to encrypt stored data, but it's important to ensure that backup data is also encrypted and stored separately from the decryption keys. (Not, for example, stored in encrypted form but alongside the keys in plaintext.) As well as encrypting data at rest, it's also important to ensure confidential data is encrypted in motion over your network to protect against database security threats.

Minimize Value of Your Database

Attackers can only get their hands on what is stored in a database, so ensure that you are not storing any confidential information that doesn't need to be there. Actively manage the data so you can delete any information that you don't need from the database. Data that must be retained for compliance or other purposes can be moved to more secure storage – perhaps offline -- which is less susceptible to database security threats.

In a similar vein, ensure you delete any history files (such as the MySQL history file ~/.mysql_history) that are written by a server during the original install procedure. While these files are useful to analyze if the install fails, if installation is successful they have no value to you but can contain information which is valuable to attackers.

Manage Database Access Tightly

You should aim for the least number of people possible to have access to the database. Administrators should have only the bare minimum privileges they need to do their job, and only during periods while they need access. For smaller

organizations this may not be practical, but at the very least permissions should be managed using groups or roles rather than granted directly.

If yours is a larger organization, you should consider automating access management using access management software. This can provide authorized users with a temporary password with the privileges they require each time they need to access a database. It also logs the activities carried out during that period and prevents administrators from sharing passwords. While admins may find sharing passwords convenient, doing so makes proper database security and accountability almost impossible.

On top of this, it is wise to ensure standard account security procedures are followed:

Strong passwords should be enforced

Password hashes should be stored encrypted and salted

Accounts should be locked after three or four login attempts

A procedure should be put in place to ensure that accounts are deactivated when staff leave or move to different roles

Audit and Monitor Database Activity

This includes monitoring logins (and attempted logins) to the operating system and database and reviewing logs regularly to detect anomalous activity.

Effective monitoring should allow you to spot when an account has been compromised, when an employee is carrying out suspicious activities or when your database is under attack. It should also help you determine if users are sharing accounts, and alert you if accounts are created without your permission (for example, by a hacker).

Database activity monitoring (DAM) software can help with this by providing monitoring which is independent of native database logging and audit functions; it can also help monitor administrator activity.

Software Development Security

Security, as part of the software development process, is an ongoing process involving people and practices, and ensures application confidentiality, integrity, and availability. Secure software is the result of security aware software development processes where security is built in and thus software is developed with security in mind.[1]

Security is most effective if planned and managed throughout every stage of software development life cycle (SDLC), especially in critical applications or those that process sensitive information.

The solution to software development security is more than just the technology.

Some of the basic steps are

1. Assess the landscape

SDLC phase: Requirements gathering

Begin the cycle with a strong understanding of what the customer actually wants. Here's how to make that happen:

Establish the scope and boundaries

Identify stakeholders

Identify process gaps

Institute tailored security-centric processes scaled to the organization and project scope

2. Incorporate an industry-standard security model

SDLC phase: Requirements gathering

Secure the software you're building from the beginning. This is the most cost-effective way to minimize the 'test-patch-retest' cycle that often negatively affects budget and scheduling goals near the end of the life cycle.

Integrate a trusted maturity model into your SDLC to infuse best practices and solid security design principles into the organization. The Building Security In Maturity Model (BSIMM) acts as a measuring stick that pinpoints strengths and weaknesses in your current security initiative. A BSIMM assessment can help your firm create data-driven goals.

3. Educate personnel on software security

SDLC phase: Requirements gathering

Ensure that all personnel involved in the project are knowledgeable and up-to-date with software security standards to reduce insecure design and development practices. Investing in training your staff is scalable, and aligns with the overall organization and the scope of each software development project at hand. The benefits resulting in a well-trained staff span all software development projects and can be an enterprise-wide asset.

4. Assign responsibility of software security

SDLC phase: Requirements gathering

To ensure that software security is incorporated into the SDLC, formally assign responsibility for it. Depending on the size of your organization, creating a software security group (SSG) is an effective way to educate, assess, and enforce established security measures across the organization. This is key to

maintaining change and risk management as your organization scales up, without degrading or ignoring security all together.

The SSG should act as the subject matter experts in software security, facilitating and conducting third-party security assessments during critical stages within the SDLC.

5. Perform security-focused requirements gathering

SDLC phase: Requirements gathering

Tailor your organization's approach to generating security requirements as a part of the initial phase. This approach will aid in embedding a solid security mindset throughout the SDLC. Generate abuse and misuse cases and perform an initial risk analysis during the requirements gathering phase to promote security activities in additional phases within the SDLC. This will also drive focus on testability when generating requirements.

6. Establish and institute a comprehensive risk management process

SDLC phase: Requirements gathering

It is critical to your SDLC's success to identify major risks and execute a mitigation plan. These are also key aspects to:

Ensure proper security design

Ensure an effective guide in SDLC execution in terms of:

Controlling scope-creep

Staying within budget and schedule goals

Engaging with stakeholders

7. Perform architecture reviews and threat modeling

SDLC phase: Design

It is far more cost-effective to identify and remediate design flaws early in the design process than to patch flawed design implementations once the software is deployed. Along with threat modeling, architecture risk analysis is a critical tool to detect design flaws. Flaws are identified by:

Analyzing fundamental design principles

Assessing the attack surface

Enumerating various threat agents

Identifying weaknesses and gaps in security controls

8. Carry out code reviews during implementation

SDLC phase: Implementation

Along with secure coding standards and static code analyses, perform a secure code review as a condition to passing a release gate. This drastically reduces the number of bugs escaping into the finished product. An effective defect containment and management system also aids in prioritization and tracking defects to resolution.

9. Execute test plans and perform penetration tests

SDLC phase: Verification

Execute the test plans during the verification phase. This will verify whether the product performs as expected in runtime scenarios. Penetration tests assess how the product handles various abuse cases, including:

Malformed input handling

Business logic flaws

Authentication/authorization bypass attempts

Overall security posture

10. Deploy software product

SDLC phase: Deployment/maintenance

Generate a deployment plan. This is essential to a successful release to production once thorough QA and acceptance testing are complete. The plan should detail the environment in which the software will operate and the steps for configuration and launch.

Plans for software maintenance and a change management process should be in place at this stage to efficiently handle any bugs or enhancement requests that come out of production.

Rollback plans and disaster recovery requirements in this phase also help ensure continued customer confidence.

Software Architecture

Software architecture refers to the high level structures of a software system, the discipline of creating such structures, and system. Each structure comprises software elements, relations among them, and properties of both elements and relations.[1] The architecture of a software system is a metaphor, analogous to the architecture of a building.[2] It functions as a blueprint for the system and the developing project, laying out the tasks necessary to be executed by the design teams.[3]

Software architecture is about making fundamental structural choices which are costly to change once implemented. Software architecture choices include specific structural options from possibilities in the design of software. For example, the systems that controlled the space shuttle launch vehicle had the requirement of being very fast and very reliable. Therefore, an appropriate real-time computing language would need to be chosen. Additionally, to satisfy the need for reliability the choice could be made to have multiple redundant and independently produced copies of the program, and to run these copies on independent hardware while cross-checking results.