Vulnerability Analysis

Vulnerability Analysis / Vulnerability Assessment

- A vulnerability assessment process that is intended to identify threats and the risks they pose typically involves the use of automated testing tools, such as network security scanners.
- And this results are listed in a vulnerability assessment report.

- Organizations of any size, or even individuals who face an increased risk of cyber attacks, can benefit from some form of vulnerability assessment.
- But large enterprises and other types of organizations that are subject to on going attacks will benefit most from vulnerability analysis. Because security vulnerabilities can enable hackers to access IT systems and applications.
- It is essential for enterprises to identify and remediate weaknesses before they can be exploited.

Examples of Vulnerability:

Understanding different types of vulnerabilities is essential for protecting your organization from malicious attacks. An issue like this can expose the system to multiple threats. Common examples of network security vulnerabilities are:

1. Unencrypted data on the network
2. Unknown security bugs in software or programming interfaces
3. Hidden backdoor program
4. Automated running of scripts without malware/virus checks in web browsers
5. Super user or admin account privileges

Types of vulnerability assessments:

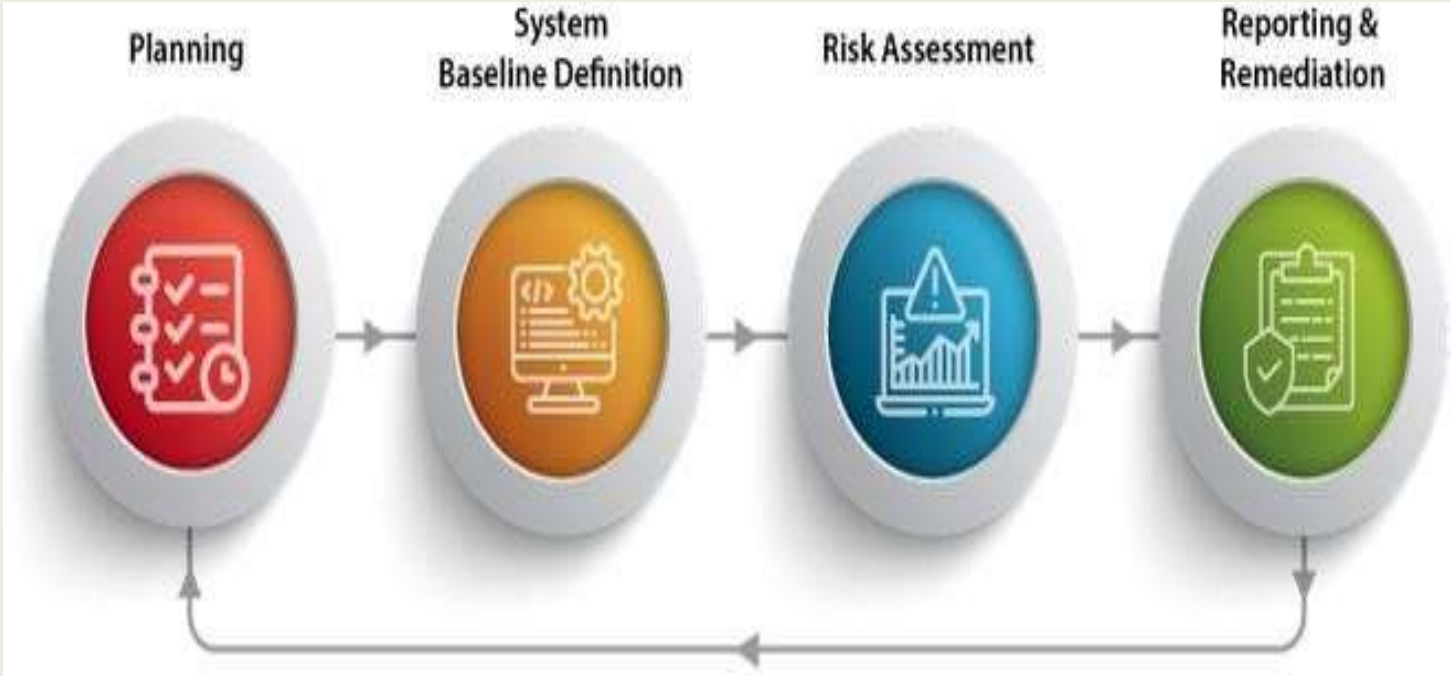
Vulnerability assessments depend on discovering different types of system or network vulnerabilities, which means the assessment process includes using a variety of tools.

- **Network-based scans** are used to identify possible network security attacks. This type of scan can also detect vulnerable systems on wired or wireless networks.

- **Host-based scans** are used to locate and identify vulnerabilities in servers, workstations or other network hosts. This type of scan usually examines ports and services that may also be visible to network-based scans, but it offers greater visibility into the configuration settings and patch history of scanned systems.

- **Wireless network scans** of an organization's Wi-Fi networks usually focus on points of attack in the wireless network infrastructure. In addition to identifying rogue access points, a wireless network scan can also validate that a company's network is securely configured.
- **Application scans** can be used to test websites to detect known software vulnerabilities and erroneous configurations in network or web application
- **Database scans** can be used to identify the weak points in a database so as to prevent malicious attacks, such as SQL injection attacks.

Four Steps to Vulnerability Assessment:



Step 1: Initial Assessment (Planning)

- Identify your assets and know the worth of the devices that are part of your network.
- Identify the risks and critical value device. It should also include a security analysis vulnerability scanner.
- Analyze if the device is accessible to everyone or limited to the authorized users and administrators alone.

The information collected from the above three steps can be used to find solutions and predict:

1. *The level and impact of risk.*
2. *Analysis of the impact of the said risk(s) on the business.*
3. *Proposing a risk strategy.*
4. *Simulating a cyber attack scenario.*
5. *Possible mitigation of risks for each device or service.*
6. *Setting up practices and policies for risk mitigation in each device.*

Step 2: System Baseline Definition

- Documentation of installed systems on your network, their capabilities, and the users who have access.
- Document all the services, processes, and open ports of those devices.
- Scan your device or use threat intelligence and a vulnerability database to detect vulnerabilities and remove false positives.

- Furthermore, it would help if you acquainted yourself with certified drivers and software.
- You can create a record of public data and vulnerabilities regarding the device program, vendor, version, and other significant details.

  This way, you can avoid installing any low-quality product with high risk.

Step 3: Vulnerability Assessment and Risk Assessment

- This stage helps in identifying the root cause of the vulnerabilities.
- From here, you can prioritize such vulnerabilities according to the threat level.
- For example, if the source of vulnerability is an obsolete version of an open-source library, you need to upgrade it.
- A certified ethical hacker or other specialized security analyst will conduct the risk assessment by allocating a severity score.

They will rank the vulnerability based on the following factors:

- *Severity of an attack.*
- *Systems affected during that attack.*
- *The potential business function(s) at risk.*
- *The possible harm the vulnerability may trigger.*

Step 4: Reporting and Remediation

- This stage is the most important because it aims to close all the security gaps. It is usually a team effort by the security staff, IT professionals, developers, and operations team.

- It also requires involvement from the incidence response team. Their presence ascertains the most effective response and remediation strategy that is mapped out in a vulnerability assessment report.

This report should include:

- Introduction of new techniques for risk mitigation.
- Identifying the potential gap between the results and the system baseline.
- Implementing measures to mitigate potential vulnerabilities.
- Solutions are reported based on the original assessment objectives.
- Conclusions are drawn according to the data collected during vulnerability assessment and are organized to assure the findings' assessment.

Vulnerability Assessment:

A vulnerability assessment is a systematic review of security weaknesses in an information system.

It evaluates if:

- *the system is susceptible to any known vulnerabilities,*
- *assigns severity levels to those vulnerabilities,*
- *and recommends remediation or mitigation,*
- *if and whenever needed.*

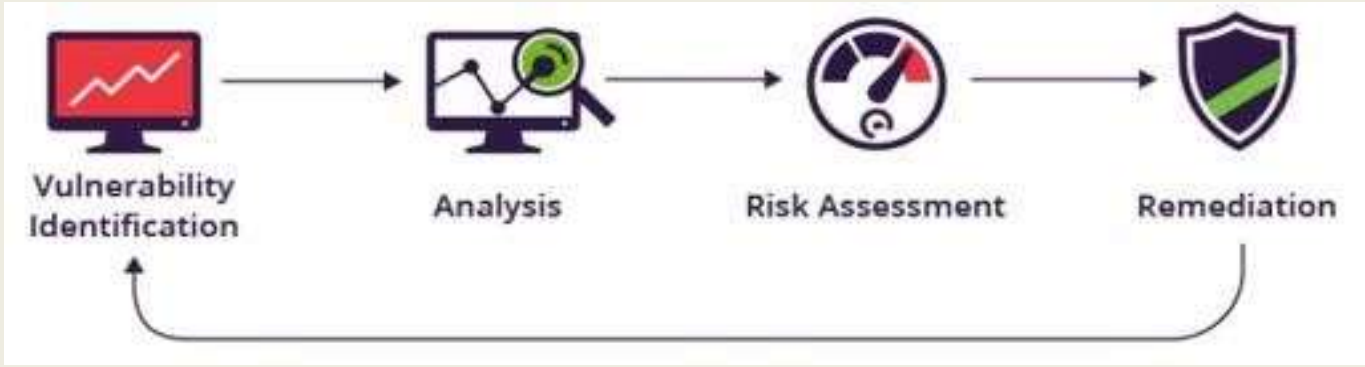
Examples of threats that can be prevented by vulnerability assessment include:

- *SQL injection, XSS and other code injection attacks.*
- *Escalation of privileges due to faulty authentication mechanisms.*
- *Insecure defaults – software that ships with insecure settings, such as a guessable admin passwords.*

Importance of vulnerability assessments:

•A vulnerability assessment provides an organization with information on the security weaknesses in its environment and provides direction on how to assess the risks associated with those weaknesses and evolving threats.

•This process offers the organization a better understanding of its assets, security flaws and overall risk, reducing the likelihood that a cybercriminal will breach its systems and catch the business off guard.



Risk assessment:

- The objective of this step is the prioritizing of vulnerabilities.
- It involves security analysts assigning a rank or severity score to each vulnerability, based on such factors as:

1. Which systems are affected.
2. What data is at risk.
3. Which business functions are at risk.
4. Ease of attack or compromise.
5. Severity of an attack.
6. Potential damage as a result of the vulnerability.

Vulnerability assessment tools:

Vulnerability assessment tools are designed to automatically scan for new and existing threats that can target your application.

Types of tools include:

1.Web application scanners that test for and simulate known attack patterns.

2.Protocol scanners that search for vulnerable protocols, ports and network services.

3.Network scanners that help visualize networks and discover warning signals like stray IP addresses, spoofed packets and suspicious packet generation from a single IP address.

Vulnerability assessments vs. penetration tests:

A vulnerability assessment often includes a penetration testing component to identify vulnerabilities in an organization's personnel, procedures or processes that might not be detectable with network or system scans.

The process is sometimes referred to as vulnerability assessment/penetration testing, or VAPT.

- However, penetration testing is not sufficient as a complete vulnerability assessment and is, in fact, a separate process.
- A vulnerability assessment aims to uncover vulnerabilities in a network and recommend the appropriate mitigation or remediation to reduce or remove the risks.

- In contrast, penetration testing involves identifying vulnerabilities in a network, and it attempts to exploit them to attack the system.
- Although sometimes carried out in concert with vulnerability assessments, the primary aim of penetration testing is to check whether a vulnerability really exists and to prove that exploiting it can damage the application or network.
- While a vulnerability assessment is usually automated to cover a wide variety of unpatched vulnerabilities.

- Penetration testing generally combines automated and manual techniques to help testers delve (to examine carefully) further into the vulnerabilities and exploit them to gain access to the network in a controlled environment.

Passive Analysis, Advanced Static Analysis with IDA Pro, Advanced Reverse Engineering

Passive Analysis:

- A Passive attack is also known as sniffing the password on a wired or wireless network.
- Passive Online Attacks is not perceivable to the end client. (Not visible to client or client is not aware of)
- The secret word is caught amid the confirmation procedure and would then be able to be analyzed against a lexicon (dictionary of) record or word list.
- Client account passwords are usually hashed or encoded when sent on the system to counteract unapproved get to and utilize.
- In the event that the secret key is secured by encryption or hashing, unique devices in the programmer's toolbox can be utilized to break the calculation.

Another Passive Online Attacks is known as man-in-the- center (MITM).

- In a MITM assault, the programmer blocks the validation demand and advances it to the server.
- A replay Passive Online Attacks is additionally a Passive Online Attacks; it happens when the programmer captures the secret key on the way to the confirmation server and after that catches and resends the validation parcels for later validation.

Examples of a Passive Attack:

- **Tapping**

Checking decoded correspondences, for example, messages or phone calls.

- **Encryption**

Blocking scrambled data streams and attempting to break the encryption.

- **Scanning**

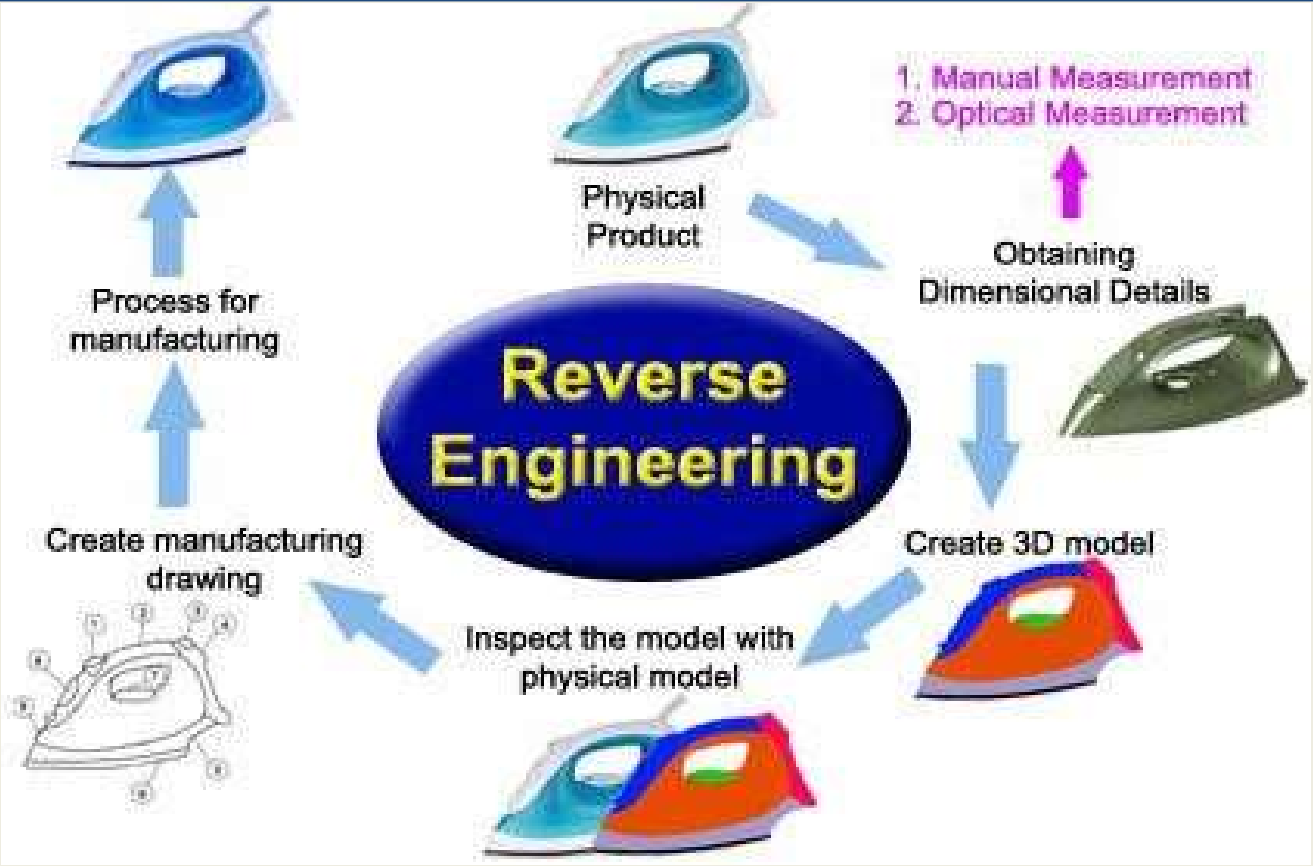
Examining a gadget associated with the web for vulnerabilities, for example, open ports or a powerless working framework rendition.

- **Traffic Analysis**

Observing web movement to construct information, for example, who is going to what site.

.

Reverse Engineering:



Reverse Engineering:

- Whether it is rebuilding a car engine or diagramming a sentence, people can learn about many things simply by taking them apart and putting them back together again.
- This process of breaking something down to understand it, build a copy to improve it, is known as reverse engineering.
- The process of reverse engineering was originally applied to hardware only, but it is now being applied to software, databases and even human DNA as well.

In the field of cyber security, the reverse engineering can be used to identify the details of a breach that how the attacker entered the system, and what steps were taken to breach the system.

What is reverse engineering?

At the highest level, it is simply taking a product apart to understand how it works.

- Understand the capabilities of the product's manufacturer.
- Understand the functions of the product in order to create compatible components.
- Determine whether vulnerabilities exist in a product.
- Determine whether an application contains any undocumented functionality.

Here are some essential tools that are used to perform reverse engineering in Kali Linux:

1. Apktool
2. dex2jar
3. diStorm3
4. edb-debugger
5. Jad Debugger
6. Javasnoop
7. OllyDbg

Manual Auditing of Binary Code:

- Disassembler The purpose of a disassembler is to generate assembly language from a compiled binary.

- Decompiler the purpose of a decompiler is to attempt to generate source code from a compiled binary.

Decompilation Tools

- JReversePro and Jad (Java Decompiler).
- IDA Pro
- Hex-rays

IDA Pro:

- **https://www.hex-rays.com/products/ida/support/download_freeware/**
- Developer Ilfak Guilfanov
- Use Decompiler
- Structure IDA Pro is actually a database application.
- Working When a binary is loaded for analysis, IDA Pro loads each byte of the binary into a database and associates various flags with each byte. These flags can indicate whether a byte represents code, data, or more specific information such as the first byte of a multibyte instruction.
- .IDB Disassemblies are saved as IDB files separate from the original binary.

Dynamic Linking to analyze dynamically linked binaries, IDA Pro makes use of embedded symbol table information to recognize references to external functions.

Static Linking For statically linked C/C++ binaries, IDA Pro uses a technique termed Fast Library Identification and Recognition Technology (FLIRT), which attempts to recognize whether a given machine language function is known to be a standard library function, by matching disassembled code against signatures of standard library functions used by common compilers.

Powerful Features:

- Code graphing capabilities to chart function relationships

  Flowcharting capabilities to chart function flow
- A strings window to display sequences of ASCII or Unicode characters contained in the binary file
- A large database of common data structure layouts and function prototypes
- A powerful plug-in architecture that allows extensions to IDA Pro's capabilities to be easily incorporated
- A scripting engine for automating many analysis tasks
- Several integrated debuggers.

IDA Pro Freeware:

- Installation and Basic Interface
- IDA pro has multiple version, freeware, paid and evaluation for Windows, Linux and Mac
- Download installer from Hex-rays official site as per your OS. For Linux: make binary executable by $chmod +x idapro XX.run Then run the installer
- Move to the folder example, /home/kali/idaprofreeware
- $ ./ida64

The process of reverse engineering involves using certain tools which consist of:

•Disassemblers. Disassemblers are used to disect binary (machine) codes into assembly codes. They are also employed in extracting strings, functions (both imported and exported), libraries, etc. they help to convert the machine language into a user-friendly format.

•Debuggers. Programmers use debuggers to set breakpoints as well as edit assembly codes at run time. Also, they let the reverser step through the code by running a line at a time so as to investigate the results.

- Hex Editors. Hex editors are used to view and edit binary files, they are sometimes referred to as a binary editor or a binary file editor.
- PE (Portable Executable) and Resource Viewer. This tool allows programmers to view and edit resources (menus, dialog, string table icons, cursors etc.) that are embedded in the EXE file. They let them change icons, edit menu, version information, dialog, etc.
- Reverse engineering software allows programmers to manipulate raw data into a useful form, thanks to the development of various digitizing devices.

Static / Passive Reverse Engineering:

- Static (also called passive) reverse engineering techniques in which you attempt to discover potential flaws and vulnerabilities simply by examining source or compiled code.

Vulnerability Analysis v/s Reverse Engineering:

- One Should be interested in reverse engineering if want to extend their vulnerability assessment skills beyond the use of the pentester's standard bag of tricks.
- Nessus Tool
- Pentesting tools can only report on **what they know.** They can't report on **undiscovered vulnerabilities,**
- Reverse engineering helps to find new and Unknown vulnerability
- Vulnerability researchers use a variety of reverse engineering techniques to find new vulnerabilities in existing software.

Reverse Engineering Considerations:

- Failure to check for error conditions
- Poor understanding of function behaviors
- Poorly designed protocols
- Improper testing for boundary conditions

Client-side browser exploits

Client Side Browser Exploits:

- Client-side vulnerabilities are vulnerabilities in client software such as web browsers, e-mail applications, and media players.
- An attacker have to get access to your client workstation in order to target vulnerabilities in your client software.
- The firewall should protect you from those attacks and uses a proxy server to protect against web attacks, so that is double protection.

If an attacker wants to attack your firewall-protected computer, he will normally be blocked by your firewall.

He needs to find a vulnerability either in the browser or in a component that the browser uses to display web content.

Client-Side Applications Are Often Running with Administrative Privileges

- Many users log onto their workstation as a user in the local Administrators group. If the users are logged in as an administrator, their Internet Explorer or Outlook session is also running as an administrator.
- This gives all the same rights as an attack against a system-level service—administrators can install rootkits and key loggers, install and start services, and access LSA (Local Security Authority) secrets.
- **(LSA Secrets** is a registry location which contains important data that are used by the Local Security Authority like authentication, logging users on to the host, local security policy etc.)

Client-Side Vulnerabilities Can Easily Target Specific People or Organizations:

- Some attackers are interested in targeting specific victims or victims belonging to a specific group, company, or organization.
- Client-side attacks with the intent of industrial espionage and stealing secrets. This is sometimes referred to as spear phishing.
- Attackers build sophisticated, convincing e-mails that appear to be from a trusted associate.
- Victims click on a link in the e-mail and end up at evil.com with the attacker serving up malicious web content from an attack web server to the victim's workstation.

Internet Explorer Security Concepts:

ActiveX Controls

- Micros These “controls” are just small programs written to be run from within a container, usually Internet Explorer. once added ActiveX support to Internet Explorer to give developers the opportunity to extend the browsing experience.
- ActiveX controls can do just about anything that the user running them can do, including access the registry or modify the file system.
- However, it presents a security warning to the user along with a digital signature from the control’s developer.
- The danger comes when a control is marked as safe to be scripted by anyone, is signed by a trustworthy corporation, and has a security vulnerability.

Protecting Yourself from Client-Side Exploits:

- Keep Up-to-Date on Security Patches
- Stay Informed
- Run Internet-Facing Applications with Reduced Privileges.

Exploiting Windows Access Control Model for Local Elevation Privilege

Exploiting the Windows Access Control Model:

- Vulnerabilities You Find Are Easy to Exploit.
- How Windows Access Control Works
- Security Identifier
- Access Token
- Security Descriptor
- The Access check

Exploiting the Windows Access Control Model:

- To find instances of misconfigured access control that are exploitable for local privilege escalation.

- Access control is about the science of protecting things.
- Finding vulnerabilities in poorly implemented access control is fun because it feels like what security is all about.
- It isn't blindly sending huge, long strings into small buffers, neither is it tricking Internet Explorer into loading an object not built to be loaded in a browser.
- Exploiting access control vulnerabilities is more about elegantly probing, investigating, and then exploiting the single bit in the entire system that was coded incorrectly
- Compromising the whole system because of that one tiny mistake.

Vulnerabilities You Find Are Easy to Exploit:

- The UPnP host (Universal Plug and Play) actually a vulnerability that is fixed by Microsoft in 2006.
- The access control governing the Universal Plug and Play (UPnP) service on Windows XP allowed any user to control which binary was launched when this service was started.
- It also allowed any user to stop and start the service.
- Windows includes a built-in utility (sc.exe) to change what binary is launched when a service starts.

How Windows Access Control Works:

- To attack Windows Access Control the security identifier (SID), the access token, the security descriptor (SD), and the access check.

Security Identifier:

- Every user and every entity for which the system needs to make a trust decision is assigned a security identifier (SID).
- The SID is created when the entity is created and remains the same for the life of that entity.
- No two entities on the same computer will ever have the same SID. The SID is a unique identifier that shows up every place a user or other entity needs to be identified.

- SIDs come in several different flavors. Every system has internal, well-known SIDs that identify built- in accounts and are always the same on every system.

- They come in the form S-[revision level]- [authority value]-[identifier].
- For example:
  - *SID: S-1-5-18 is the Local System account. It's the same on every Windows machine.*
  - *SID: S-1-5-19 is the Local Service account on every Windows XP and later system.*
  - *SID: S-1-5-20 is the Network Service account on ever Windows XP and later.*

Access Token:

- If you work in an environment with controlled entry, you are probably familiar with presenting your badge to a security guard or a card reader to gain access.
- For example, a blue badge might grant a person access at times when a yellow badge or purple badge is denied entry.
- Windows access tokens work in a similar manner as an employee badge.
- The access Token is a container of all a user's security information and is checked when that user requests access to a secured resource.

Security Descriptor

- The operation performed by the operating system anytime access to a securable object is requested.
- The other half of the Access Check operation is the security descriptor (SD) of the object for which access is being requested.
- The SD describes the security protections of the object by listing all the entities that are allowed access to the object.
- SD holds the owner of the object, the Dynamic Access Control List (DACL), and a System Access Control List (SACL). The DACL describes who can and cannot access a securable object by listing each access granted or denied in a series of access control entries (ACEs).

The Access Check:

- The core function of the Windows Access Control model is to handle a request for a certain access right by comparing the access token of the requesting process against the protections provided by the SD (security descriptor) of the object requested.
- Windows implements this logic in a function called Access Check.

Intelligent Fuzzing with Sulley

Intelligent Fuzzing with Sulley:

- Fuzzing (Fuzz testing) can effectively identify security vulnerabilities in software by providing a large amount of unexpected input to the target program.

- **Fuzzing** is a Black Box software testing technique, which basically consists in finding implementation bugs using malformed/semi-malformed data injection in an automated fashion.
- The techniques require the specification of the input data format or analyze the input data format by manual reverse engineering.

Intelligent fuzzing:

- Instead of blindly throwing everything, techniques have been developed to analyze how a server works and to customize a fuzzer to get past the filters and reach deeper inside the server to discover even more vulnerabilities.

Intelligent fuzzing Steps:

- Need to conduct a protocol analysis of the target.
- Need a way to fuzz that protocol and get feedback from the target as to how you are doing.

Sulley/Boofuzz:

- The Sulley fuzzing framework automates this process and allows you to intelligently sling packets across the network.
- This tool is truly revolutionary in that it provides not only a great fuzzer and debugger, but also the infrastructure to manage a fuzzing session and conduct postmortem analysis.
- https://github.com/OpenRCE/sulley
- https://github.com/jtpereyda/boofuzz

- The Sulley fuzzing framework automates this process and allows you to intelligently sling packets across the network.
    - Protocol analysis
    - Sulley fuzzing framework

Protocol Analysis

- Servers perform a routine task and need to interoperate with random clients and other servers, most servers are based on some sort of standard protocol.
- The Internet Engineering Task Force (IETF) maintains the set of protocols that forms the Internet.

Sulley Fuzzing Framework

- Sulley gets its name from the fuzzy character in the movie Monsters, Inc.
- This tool is truly revolutionary in that it provides not only a great fuzzer and debugger, but also the infrastructure to manage a fuzzing session.

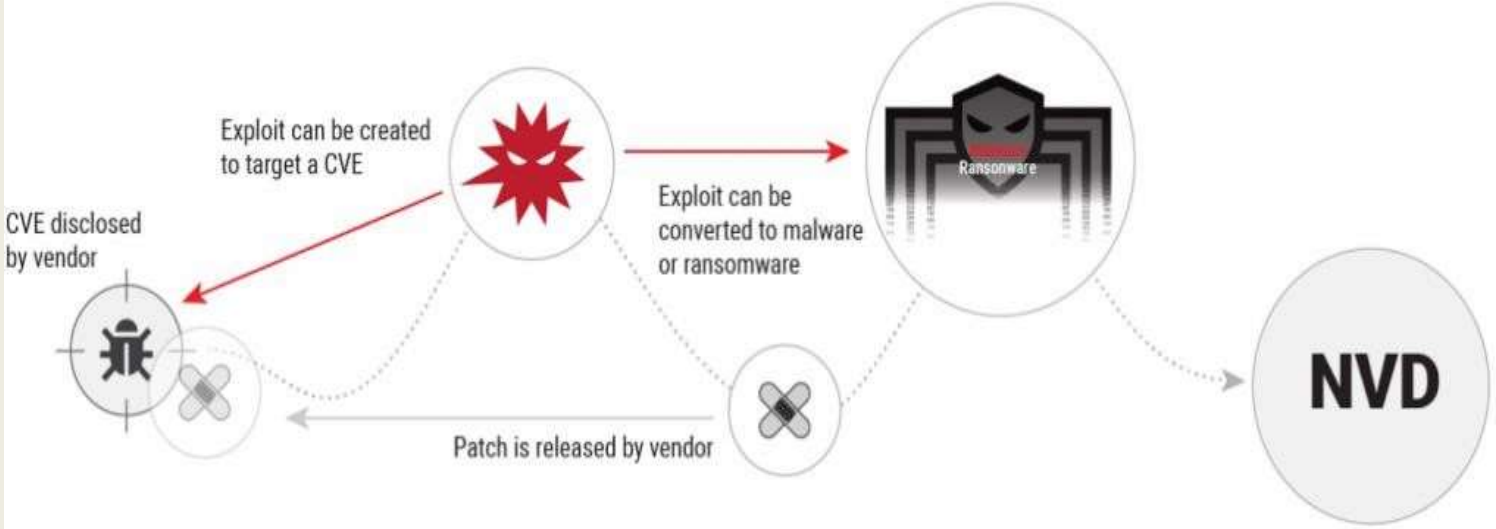
- **Installing Sulley**
  - *Download the latest version of Sulley from* *http://code.google.com/p/sulley/.*

From Vulnerability to Exploit

From Vulnerability to Exploit:

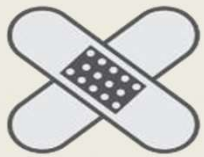
- It is used to discover a problem with a piece of software, locating a potential problem or causing a program to melt down.
- Two analysis are used normally
  - ***static analysis,***
  - ***dynamic analysis***
- With static analysis in particular, you face the task of determining exactly how to reach the vulnerable code while the program is executing.
- Additional (Dynamic) analysis followed by testing against a running program is the only way to confirm that your static analysis is correct.

Possible Random Events During Latency



NVD

National Vulnerability Database



Patch

Exploit

Exploitability

- Crash ability and exploitability are vastly different things.
- The ability to crash an application is, at a minimum, a form of denial of service.
- For true exploitability, you are really interested in injecting and executing your own code within the vulnerable process..

As a team,

- Identify any digital application/software (real time application)
- 2019 to 2022 : List the Vulnerabilities & Attacks
- Take 2 attacks and Infer how it is successful.
- How the weakness has been changed in the application?

Prepare Presentation of 8 slides (max) by addressing

Thank You