



Dr. Vishwanath Karad

**MIT WORLD PEACE
UNIVERSITY** | PUNE

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY

Module - 4

VAPT Audit and Uses cases

Contents

VIPT Audit and Uses cases – 7 Hrs.

- Discovering patching vulnerabilities, Discovering web server vulnerabilities.
- Synthetic transactions, interface testing and fuzzing
- SDLC phases and security mandates. Perform Penetration Testing assessments
- Detect and respond to network breaches found in a Penetration Testing assessments
- Preparation of a Penetration Test report
- Auditing the Systems
- Analysis and Reporting
- Case Studies of recent vulnerabilities and attacks

Discovering patching vulnerabilities

- A software system deals with various security implications after its release in the market.
- Correspondingly, firm releases security patches to counter those flaws discovered in the software system.
- A vendor releases a patch only if a vulnerability has been discovered in a software.
- It is an important aspect that encompasses the prediction of potential number of patches to be released to maintain the stability of a software.

Discovering patching vulnerabilities...

- Vulnerability Discovery Models (VDMs) help a software vendor to acknowledge the security trends, forecast security investments and to plan patches, but very few attempts have been made to model the Vulnerability Patch Modeling (VPM) based on the impact of vulnerabilities discovered over the time period.
- The vulnerability trends in a software significantly affect the discovery process and later trigger a patch deployment to suppress the possible likelihood of a breach.

Is patching part of vulnerability management?

- Vulnerability management is a continuous process of identifying, prioritizing, remediating, and reporting on security vulnerabilities in systems and the software that runs on them. **Patch management is a critical component of vulnerability management**, but it's just one piece of the puzzle.

Discovering patching vulnerabilities...

- The integrative approach underlines the association of vulnerability patch modeling with the vulnerability discovery phenomenon.

What is vulnerability patching?

- Vulnerability patching is the process of checking your operating systems, software, applications, and network components for vulnerabilities that could allow a malicious user to access your system and cause damage.

Discovering patching vulnerabilities...



Software development is not a one-and-done process, but rather a continuous one. With code and capabilities evolving so often, it's impossible for any system, no matter how well built to be left untouched after deployment.

Discovering patching vulnerabilities...

Why do we need patch management?

Patch management is important for the following key reasons:

Security: Patch management fixes vulnerabilities on your software and applications that are susceptible to cyber-attacks, helping your organization reduce its security risk.

System uptime: Patch management ensures your software and applications are kept up-to-date and run smoothly, supporting system uptime.

Compliance: With the continued rise in cyber-attacks, organizations are often required by regulatory bodies to maintain a certain level of compliance. Patch management is a necessary piece of adhering to compliance standards.

Feature improvements: Patch management can go beyond software bug fixes to also include feature/functionality updates. Patches can be critical to ensuring that you have the latest and greatest that a product has to offer.

Discovering patching vulnerabilities...

How your organization benefits from an efficient patch management program?

Your company can benefit from patch management in a variety of ways:

A more secure environment: When you're regularly patching vulnerabilities, you're helping to manage and reduce the risk that exists in your environment. This helps protect your organization from potential security breaches.

Happy customers: If your organization sells a product or service that requires customers to use your technology, you know how important it is that the technology actually works. Patch management is the process of fixing software bugs, which helps keep your systems up and running.

No unnecessary fines: If your organization is not patching and, therefore, not meeting compliance standards, you could be hit with some monetary fines from regulatory bodies. Successful patch management ensures that you are in compliance.

Continued product innovation: You can implement patches to update your technology with improved features and functionality. This can provide your organization with a way to deploy your latest innovations to your software at scale.

Discovering patching vulnerabilities...

The patch management process:

It would be a poor strategy to just install new patches the second they become available for all assets in your organization's inventory without considering the impact. Instead, a more strategic approach should be taken. Patch management should be implemented with a detailed, organizational process that is both cost-effective and security-focused.

Key steps to the patch management process include:

Develop an up-to-date inventory of all your production systems: Whether this be on a quarterly or monthly basis, this is the only way to truly monitor what assets exist in your ecosystem. Through diligent asset management, you'll have an informed view of operating systems, version types, and IP addresses that exist, along with their geographic locations and organizational "owners." As a general rule, the more frequently you maintain your asset inventory, the more informed you're going to be.

Devise a plan for standardizing systems and operating systems to the same version type: Although difficult to execute on, standardizing your asset inventory makes patching faster and more efficient. You'll want to standardize your assets down to a manageable number so that you can accelerate your remediation process as new patches are released. This will help save both you and technical teams time spent remediating.

Discovering patching vulnerabilities...

Make a list of all security controls that are in place within your organization: Keep track of your firewalls, antivirus, and vulnerability management tool. You'll want to know where these are sitting, what they're protecting, and which assets are associated with them.

Compare reported vulnerabilities against your inventory: Using your vulnerability management tool to assess which vulnerabilities exist for which assets in your ecosystem is going to help you understand your security risk as an organization.

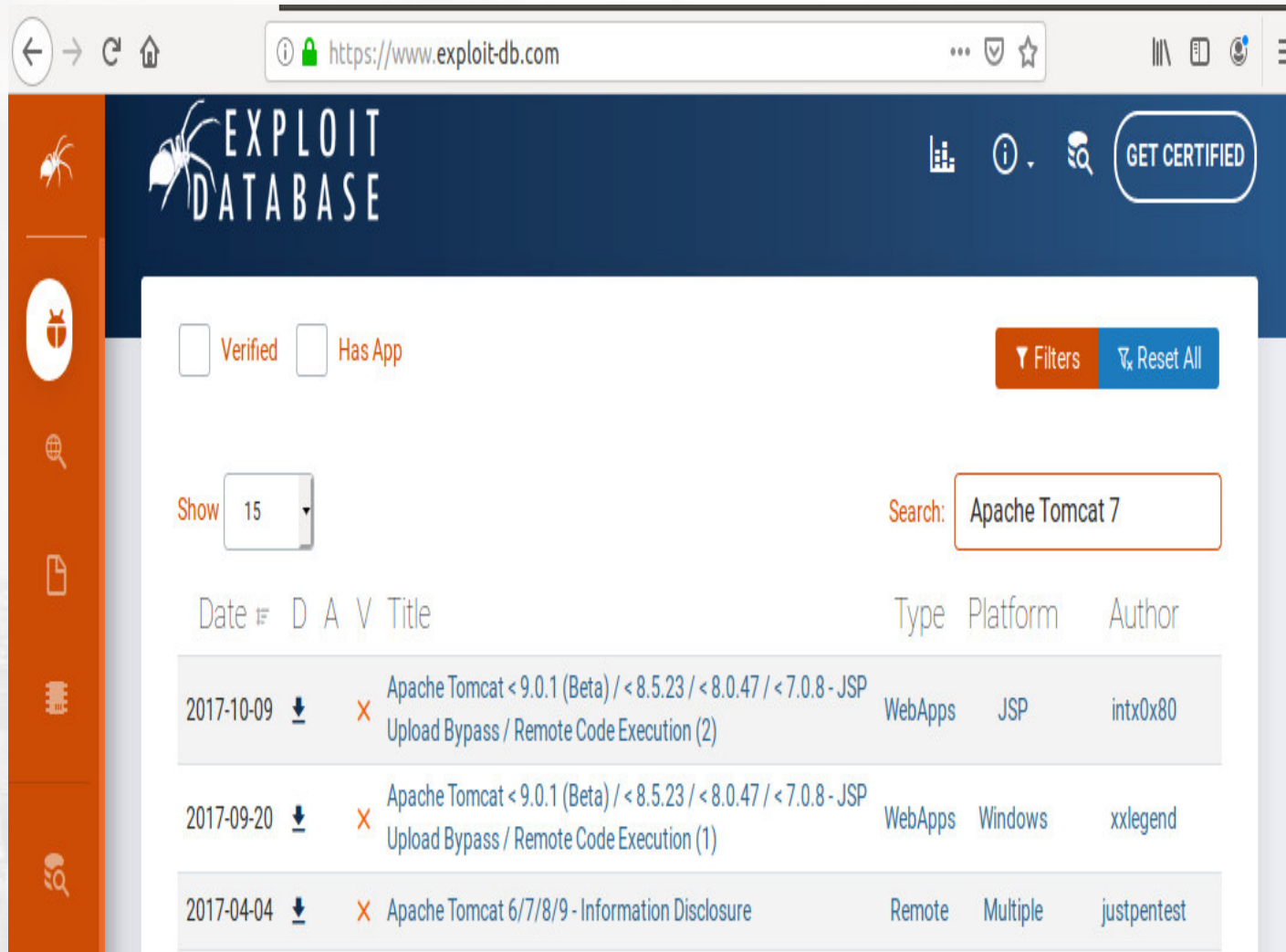
Classify the risk: Through vulnerability management tools you can easily manage which assets you consider to be critical to your organization and, therefore, prioritize what needs to be remediated accordingly.

TEST! Apply the patches to a representative sample of assets in your lab environment. Stress test the machines to ensure that the patches will not cause issues in your production environment.

Apply the patches: Once you've prioritized what needs to be remediated first, start patching to actually reduce the risk in your environment. More advanced vulnerability management tools also offer the ability to automate the time-consuming parts of the patching process. Consider rolling the patches out to batches of assets; although you already tested in your lab environment (you did do that right!?) there may still be unexpected results in production. Dip a few toes in before jumping in all the way to make there won't be any widespread issues.

Track your progress: Reassess your assets to ensure patching was successful.

Discovering patching vulnerabilities...



The screenshot shows the Exploit-DB website interface. The search bar contains 'Apache Tomcat 7'. The results table lists three entries:

Date	D	A	V	Title	Type	Platform	Author
2017-10-09	↓			Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution (2)	WebApps	JSP	intx0x80
2017-09-20	↓			Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution (1)	WebApps	Windows	xxlegend
2017-04-04	↓			Apache Tomcat 6/7/8/9 - Information Disclosure	Remote	Multiple	justpentest

Discovering patching vulnerabilities is as straightforward as identifying exactly which version of a particular software your target is running and then comparing that version to the latest stable release available from the software vendor. If your target is on an older release, you can then check public exploit databases to see if the newest release patched any remote code execution bugs that the older version may be vulnerable to.

Discovering patching vulnerabilities...

Patch management best practices

Some best practices to keep in mind when implementing patch management include:

Set clear expectations and hold teams accountable: Leveraging organizational agreements, such as service-level agreements, can keep teams in check, and ensure that the work of reducing risk is actually being done.

Work collaboratively with technical teams to ensure a common language: Security teams often refer to software errors as a “risk,” whereas IT/DevOps teams may use the term “patch.” Making sure that everyone is on the same page and recognizes the importance of patching is key to a successful patch management process.

Establish a disaster recovery process: In case your patch management process does fail and causes issues, it’s always a good idea to have a backup plan.

Discovering patching vulnerabilities...

Embedding patch management into your vulnerability management efforts:

Patch management is a vital part of every vulnerability management program. However, having a consistent approach to patch management doesn't always mean slapping a fix on everything in sight. When a vulnerability is identified, you essentially have three options:

Install a patch for the vulnerability, if available, to fix the issue.

Implement **compensating controls** so the vulnerability is mitigated without being fully patched. This route is common when a proper fix or patch is not yet available, and can be used to buy time before eventual remediation.

Accept the risk posed by that vulnerability and do nothing.

It's up to organizations to decide which option is best for them in specific situations, though patching is the ideal treatment to ultimately strive for.

Discovering patching vulnerabilities...

The terms “patch management” and “vulnerability management” are sometimes used interchangeably, but it is important to understand the difference. Though both strategies aim to mitigate risk, patch management (the process of managing software updates) is limited in scope. To gain a deeper understanding of your environment and make informed, impactful decisions, you need to move to a more holistic approach through vulnerability management. Vulnerability management is a continuous process of identifying, prioritizing, remediating, and reporting on security vulnerabilities in systems and the software that runs on them.

Patch management is a critical component of vulnerability management, but it’s just one piece of the puzzle. To successfully embed patch management into your vulnerability management program, the following steps should be implemented:

Discovering patching vulnerabilities...

Establish asset management. Your ability to reduce risk is only as good as the visibility you have into your environment. An asset management solution helps you gain a full understanding of the assets you have and the vulnerabilities associated with each asset. With that knowledge, you are equipped to prioritize vulnerabilities, remediate issues, and communicate effectively with stakeholders.

Prioritize vulnerabilities. With limited time and resources and an ever-changing threat landscape, it's unrealistic to think that you can fix every vulnerability as soon as it appears. Consequently, prioritization is one of the most critical aspects of vulnerability management.

Remediate vulnerabilities to reduce risk. Identifying and prioritizing vulnerabilities is important, but you're not actually reducing risk unless you're remediating the issues.

Measure the success of your vulnerability management program. No matter how many fancy features a vulnerability management solution has, it's only worth the investment if it meets your organization's unique needs and adds value for you and your team. To determine if you're achieving a good ROI—and justify the purchase to senior leadership—you'll have to determine how to measure success.

Develop partnerships and support. When something goes wrong, you want to know you have a team of people you can rely on to help troubleshoot.

Discovering web server vulnerabilities...

What is a web server?

A Web Server is defined as an application that responds to web page requests submitted by various users over the Internet using the HTTP (Hypertext Transfer Protocol). The Web Server basically constitutes the interface between users and web based applications and databases. These web servers therefore form the back bone to the internet and the various networks and application connecting through it. Such a publicly accessed system application is prone to certain vulnerabilities.

Vulnerabilities in Web Servers

The applications/databases that users connect to through these Web servers are called websites. Any vulnerability occurring in the front end (the user interactive part of the application) applications, database or operating systems can translate to Web Server vulnerabilities.

Discovering web server vulnerabilities...

A network service has a configuration vulnerability when one of the service's configuration settings enables an attack vector.

Example is the Apache Tomcat web server. Often, it is configured to allow the deployment of arbitrary web application archive (WAR) files via the web GUI. This allows an attacker who gains access to the web console to deploy a malicious WAR file and gain remote access to the host operating system, usually with administrator-level privileges on the target.

Many times, when an IT/systems administrator installs something, it comes with a web interface listening on an arbitrary port, and the admin doesn't even know it's there. The web service ships with a default password, and the IT/systems administrator may forget to change it—or not even know they need to do so. This presents a golden opportunity for an attacker to gain remote entry into restricted systems.



Discovering web server vulnerabilities...

DoS Attacks

A Web Server is at its optimal performance when it responds to users requests in a timely manner, usually within seconds. A Denial of Service (DoS) attack is designed to achieve the opposite. The attackers overwhelm the web server with requests in a such way that deters legitimate users from accessing the service by affecting the ability of the server to respond in a timely manner. This is achieved by the attacker transmitting an excessive number of known invalid requests.

The web server upon receipt will attempt to serve these requests. Invalid requests are subject to delays as the server attempts to close the connections. No sooner is the connection closed that more invalid requests are sent. A significantly high volume of these invalid requests cause a bottleneck in closing connections, server delayed response time and low performance, blocking out access to legitimate users.

SQL Injection Vulnerabilities

Every website contains input fields and forms in their front-end applications to facilitate the interactive processes with the user. These input data fields/forms are used to pass data through SQL queries to access and query the database. In the event that these data input fields are not properly validated, attackers can use this vulnerability to pass malicious scripts to query the database in an SQL Injection attack. There is no scope as to how much damage can be caused to the website and its database through these vulnerabilities.

Discovering web server vulnerabilities...

XSRF (Cross Site Request Forgery) Attacks

XSRF (Cross Site Request Forgery) is an attack that successfully redirects legitimate users of a website to an alternate malicious website designed to look like the authentic site but whose objective is to steal users' login, personal and any other sensitive information. Having obtained the victim's credentials, the attacker uses them to perform undesired activities on behalf of the victim.

Attacks due to poor system configuration

Proper server configurations and adherence to industry standards are keys to deterring these types of attacks. When unnecessary applications are enabled or known default configuration parameters or settings are used, attackers can easily compromise such a system.

Discovering web server vulnerabilities...

Directory Attacks

A Directory attack occurs when a malicious attacker is successful in accessing beyond the exposed front-end and back-end interfaces of the application and is able to access the files on the underlying file system in the web root directory and beyond. Such a security breach exposes the Operating system and its related software and applications allowing a leak of highly sensitive information.

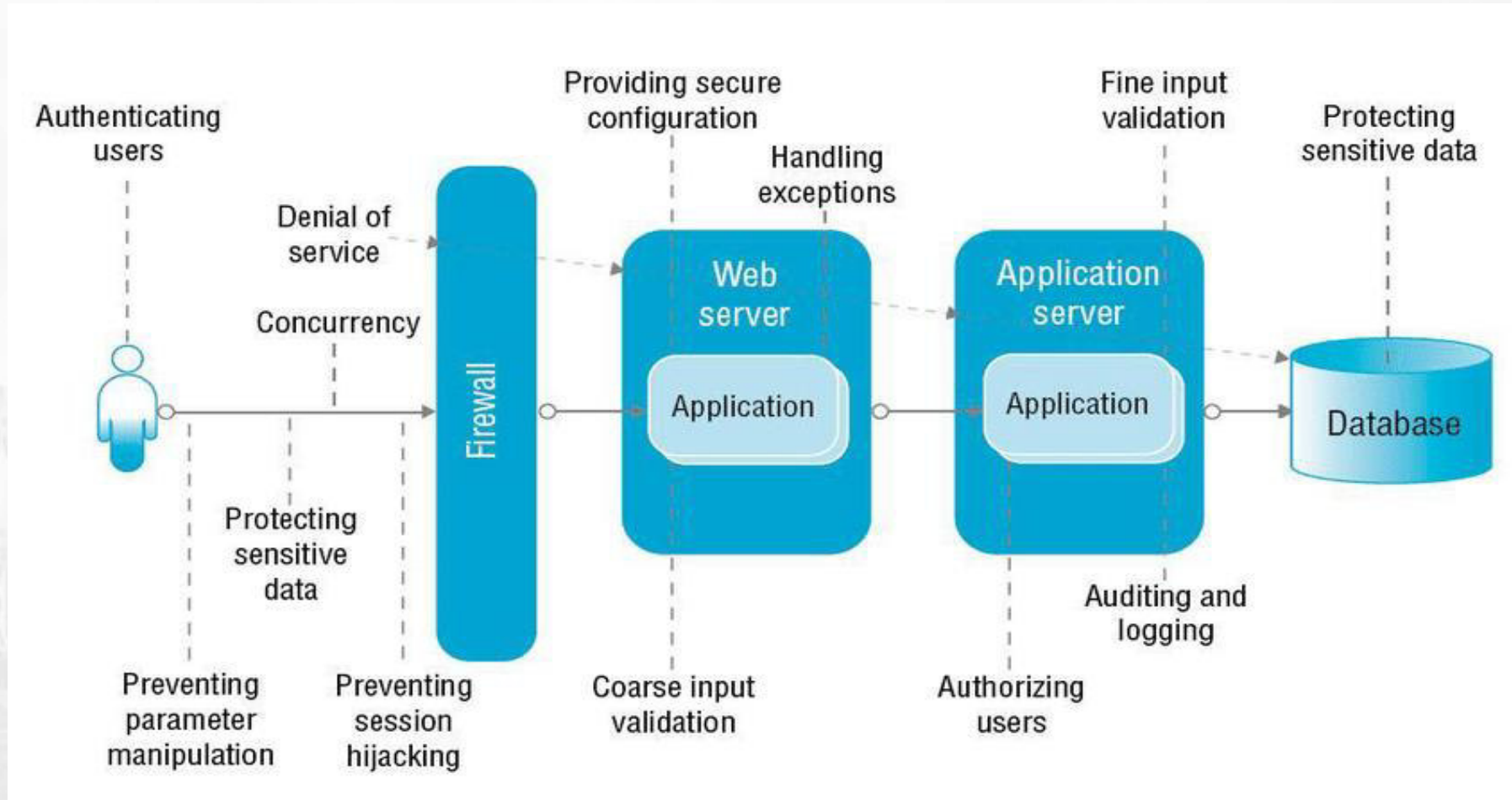
Web Server Monitoring

One of the keys to handling vulnerabilities and issues associated with web server is the conscious monitoring of applications hosted by the web server. Vulnerabilities associated with these web services pose a direct vulnerability to the application and servers. Since ports and services are the main windows through which applications on web servers are accessed, monitoring access to resources and attempts to access disallowed ports and services communicate is critical for dealing with this issue.

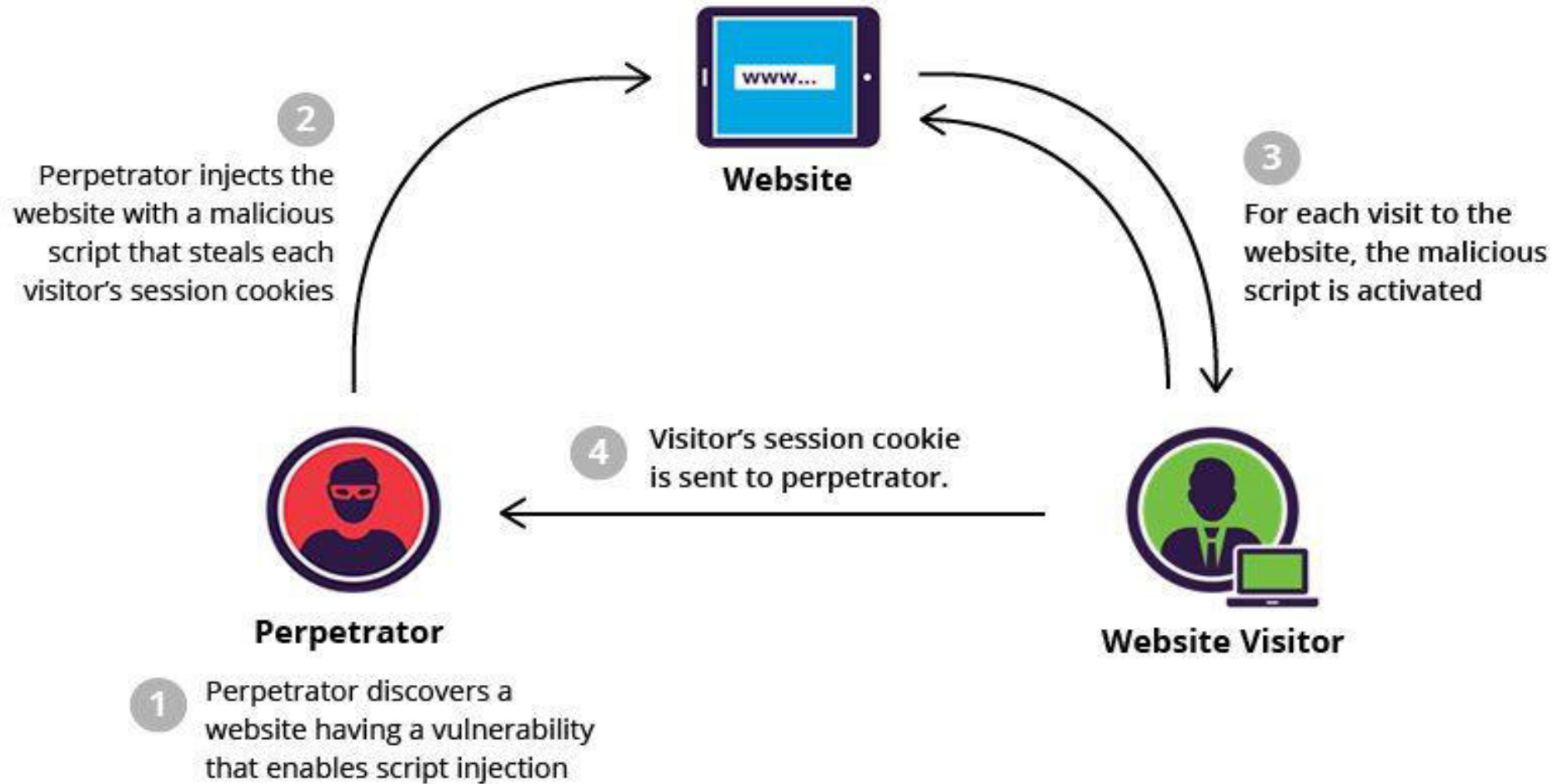
Black Box Testing

The process of testing web servers for issues and vulnerabilities involves some amount of penetration testing commonly known as black box testing.

Discovering web server vulnerabilities...

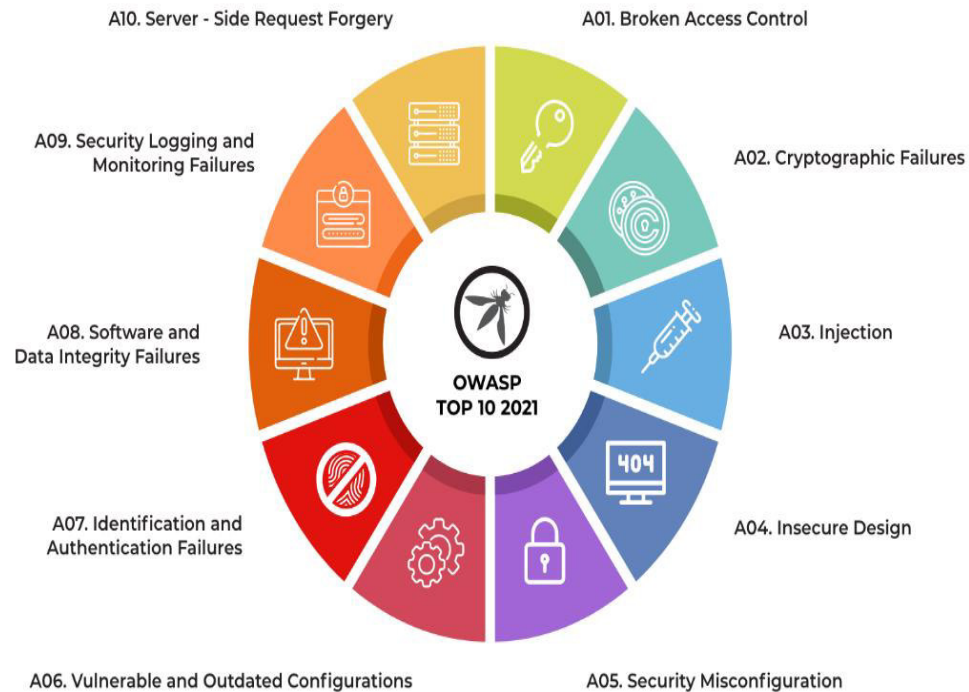


Discovering web server vulnerabilities...



Discovering web server vulnerabilities...

These are some real-life examples of each of the Top 10 Vulnerabilities and Cyber Threats for 2021 according to The Open Web Application Security Project (OWASP).



SSRF

- SSRF flaws occur whenever a web application is fetching a remote resource without validating the user-supplied URL. It allows an attacker to force the application to send a crafted request to an unexpected destination, even when protected by a firewall, VPN, or another type of network access control list (ACL).
- As modern web applications provide end-users with convenient features, fetching a URL becomes a common scenario. As a result, the incidence of SSRF is increasing. Also, the severity of SSRF is becoming higher due to cloud services and the complexity of architectures.

Scenario #1: Port scan internal servers – If the network architecture is unsegmented, attackers can map out internal networks and determine if ports are open or closed on internal servers from connection results or elapsed time to connect or reject SSRF payload connections.

Scenario #2: Sensitive data exposure – Attackers can access local files or internal services to gain sensitive information such as `file:///etc/passwd` and `http://localhost:28017/`.

Scenario #3: Access metadata storage of cloud services – Most cloud providers have metadata storage such as `http://169.254.169.254/`. An attacker can read the metadata to gain sensitive information.

Discovering web server vulnerabilities...

How do I check my network for vulnerabilities?

Bitdefender Home Scanner

app to scan your home network for vulnerabilities. Bitdefender Home Scanner is a free tool that scans your Wi-Fi network, maps devices and identifies and highlights network security flaws. Bitdefender Home Scanner looks for weak passwords, as well as vulnerable or poorly encrypted communications.

What are the vulnerabilities of web server?

Common Web Server Vulnerabilities

SQL Injection. ...

Cross-Site Scripting (XSS) ...

Distributed Denial of Service Attacks (DDoS) ...

Cross-Site Request Forgery (CSRF) ...

SQL Injection. ...

Cross-Site Scripting (XSS) ...

Distributed Denial of Service Attacks (DDoS) ...

How are the vulnerabilities discovered?

Some vulnerabilities are discovered by 'white hat' security researchers, who usually report the issue to the software vendors through established bug bounty programs (such as our Vulnerability Reward Program). Others are found by attackers, who put their discoveries to more harmful use.

Synthetic transactions

Synthetic transactions are the result of scripts simulating activity normally performed on an application or website by real users. These transactions are mainly used to test the performance of production environments, but they are also useful for testing the functionality of updated applications prior to deployment.

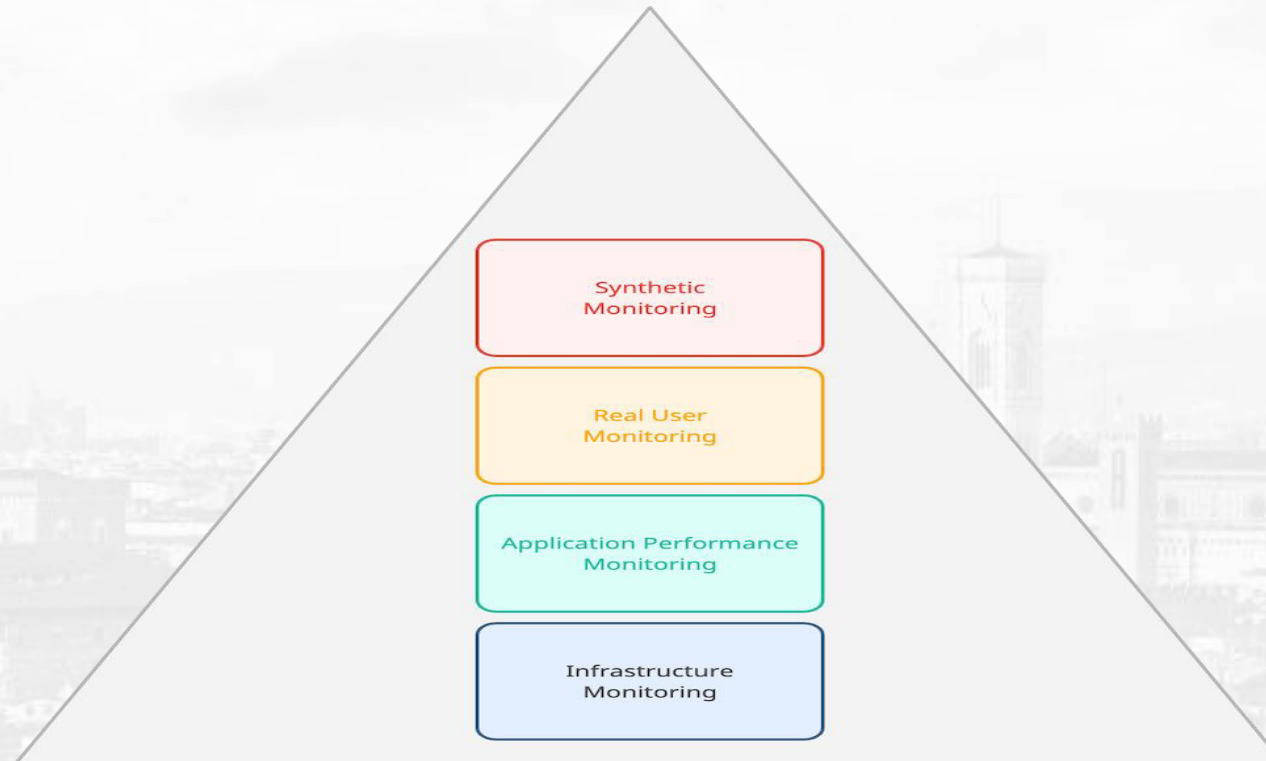


Fig. Synthetic Transactions and Monitoring in Testing

Synthetic transactions

For example, you can create a script that simulates a user's connection to your website through a defined point of entry. This script could also include any subsequent actions or navigations as a follow-up. In this case, you could use the script to validate that all users have definite access to certain resources regardless of their location or entry to your services. You could also use it to validate user roles and permissions.

Another example is simulating database connections. Using synthetic transactions that run at certain intervals allows you to see how your website or database would perform under various loads representing real user activity.

Synthetic Transactions vs. API Monitoring

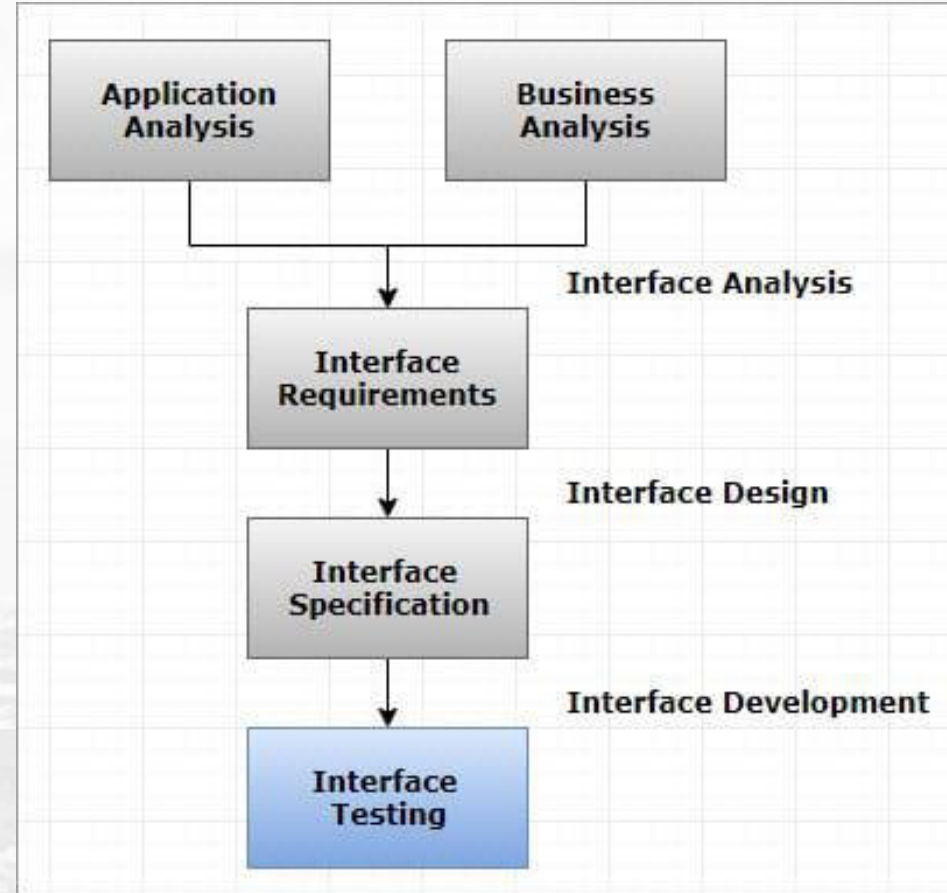
Synthetic monitoring: Mimics application behavior through simulations and emulations.

API monitoring: Validates the internal and third-party APIs used for communication.

Interface testing and fuzzing techniques

- **What is Interface Testing?**
- Interface Testing is defined as a software testing type which verifies whether the communication between two different software systems is done correctly
- A connection that integrates two components is called interface. This interface in a computer world could be anything like API's, web services, etc. Testing of these connecting services or interface is referred to as Interface Testing.

Interface testing and fuzzing techniques...



Interface testing and fuzzing techniques...

An interface is actually software that consists of sets of commands, messages, and other attributes that enable communication between a device and a user.

How to do Interface Testing?

Interface Testing includes testing of two main segments:

Web server and application server interface

Application server and Database server interface.

For above-mentioned scenarios, the interface testing is done to Check servers are executed properly or not

Errors are handled properly or return an error message for any query made by an application
Check the outcomes when connection to a web server is reset in between.

Interface testing and fuzzing techniques...

Example of Interface Testing

Suppose for any xyz application, the interface takes XML file as an input and delivers JSON file as an output. To test the interface of this application, all it requires is the specifications of XML file format and JSON file format. With the help of these specifications, we can create a sample input XML files and feed into the interface. And then validating the input (XML) and output (JSON) file with the requirement is Interface testing.

Interface testing and fuzzing techniques...

Why do Interface Testing?

Interface Testing is done

- To ensure that end-users or customer should not encounter any problem when using a particular software product
- To identify which application areas are usually accessed by end-users and to check its user-friendliness as well.
- To verify security requirements while communication propagates between the systems
- To check if a solution is capable to handle network failures between an application server and website

Interface testing and fuzzing techniques...

Interface Testing Vs Integration Testing

Interface Testing	Integration Testing
<ul style="list-style-type: none">• Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems.	<ul style="list-style-type: none">• An integration test type that is concerned with testing the interfaces between components or systems

Interface testing and fuzzing techniques...

Summary:

In Software Engineering, Interface testing is testing of connection that integrates two components of a system are called interface.

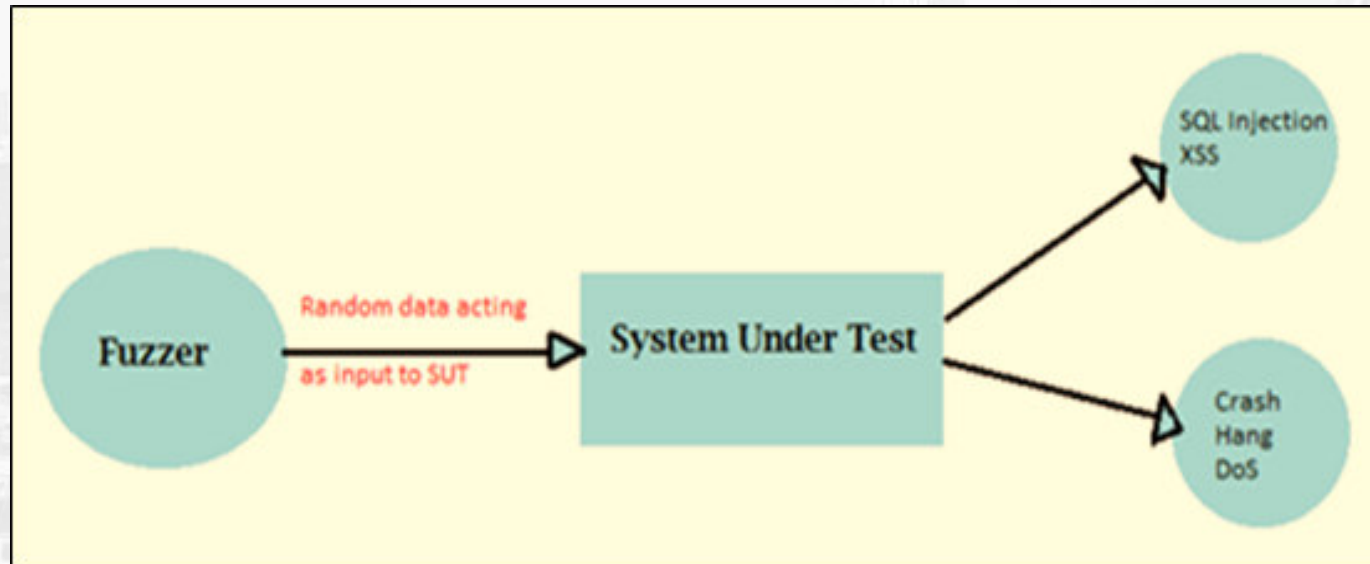
Interface Testing include testing of two main segments

- Web server and application server interface
- Application server and Database server interface.

This testing ensures that end-users or customer should not encounter any problem when using a particular software product.

fuzzing techniques...

In the world of cybersecurity, fuzz testing (or fuzzing) is an automated software testing technique that attempts to find hackable software bugs by randomly feeding invalid and unexpected inputs and data into a computer program in order to find coding errors and security loopholes.



fuzzing techniques...

- This is an old but increasingly common process both for hackers seeking vulnerabilities to exploit and defenders trying to find and first them fix.
- Fuzz testing typically involves inputting massive amounts of random data, called fuzz, to the software or system being tested in an attempt to make it crash or break through its defenses.
- If a vulnerability is found, a software tool called a fuzzer can be used to identify the potential causes

fuzzing techniques...

- Fuzzing can often reveal serious defects that are overlooked when software is written and debugged.
- Fuzzers work best for discovering vulnerabilities that can be exploited by SQL injection, buffer overflow, denial of service (DOS), and cross-site scripting.
- These are often used by malicious hackers to disable security with the intent of either taking down a system or stealing information.

SDLC phases and security mandates

- Security System Developmental Life Cycle is the overall process of developing software to minimize security risk and vulnerability. Learn the steps of investigating, analyzing, designing, and implementing security systems.
- Makes security a continuous concern, including all stakeholders in the security considerations. Helps detect flaws early in the development process, reducing business risks for the organization. Reduces costs by detecting and resolving issues early in the lifecycle.

SDLC phases and security mandates...

Security in SDLC

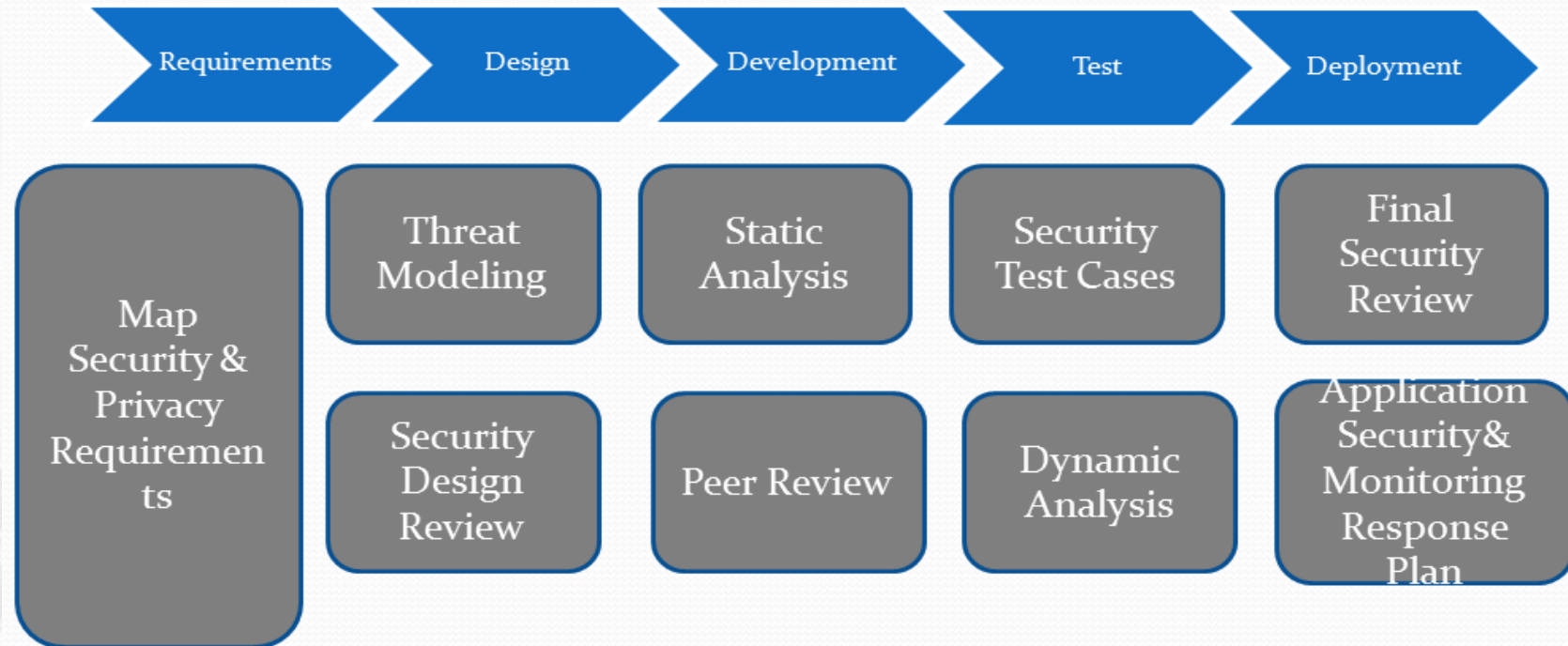
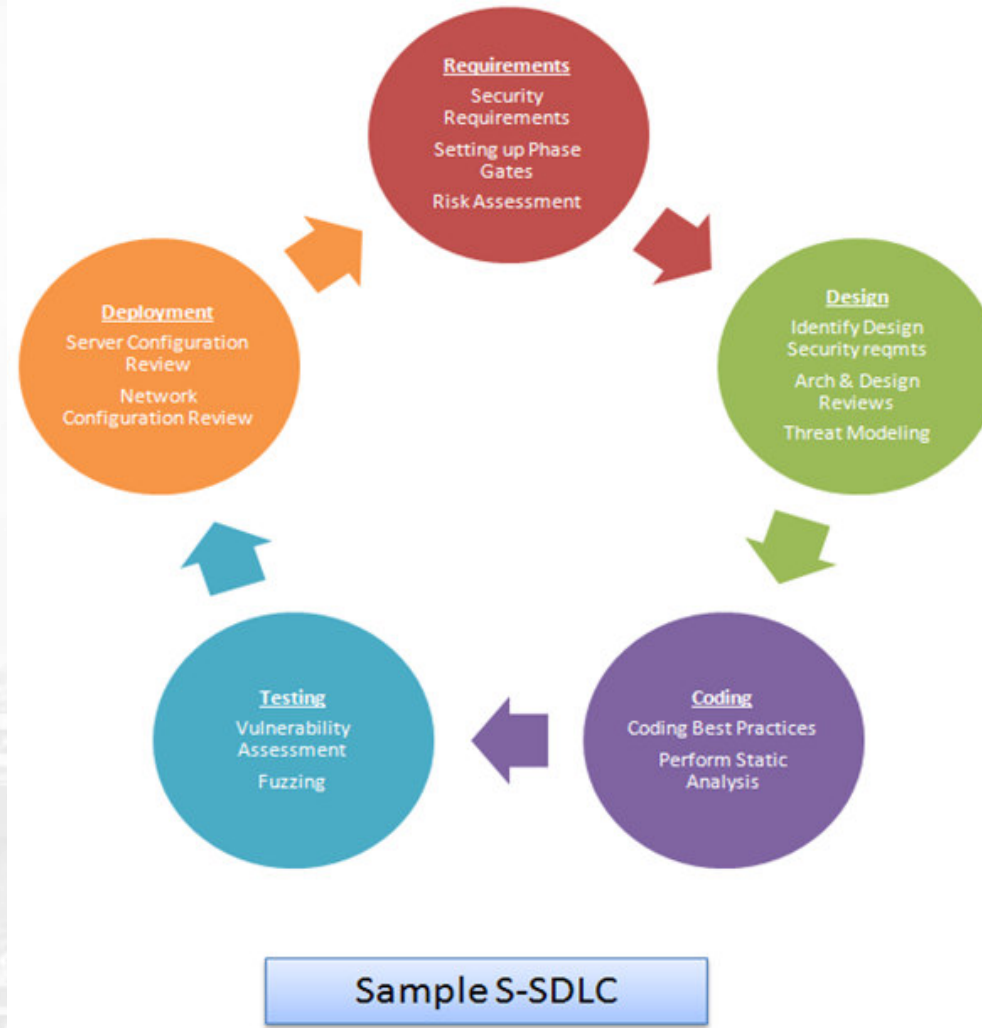


Fig.: Security Testing in SDLC

SecSDLC phases



SDLC phases and security mandates...

- Today, security of software applications and databases has become as important as the software and data itself.
- Security forms a major aspect of the business development process.
- Security System Development Life Cycle is defined as the series of processes and procedures in the software development cycle, designed to enable development teams to create software and applications in a manner that significantly reduces security risks, eliminating security vulnerabilities and reducing costs.

SDLC phases and security mandates...

- The process, like the traditional Systems Development Life Cycle, is divided into a number of phases.

Investigation & Analysis

- The overall objective, goal, and budget of the project are brought into perspective.
- An information security policy is defined to detail the various security programs and their implementation plans within the organization.
- In the system analysis phase, a detailed document analysis of the documents from the investigative phase is done.
- Existing security policies, software, and applications are analyzed and assessed.
- Current threats, new risks, and their associated internal controls are evaluated.
- During the systems analysis phase, the process of Risk Management commences.
- Risk management is defined as the series of processes that identify and evaluate current and future risks and vulnerabilities.

SDLC phases and security mandates...

Logical & Physical Design

- The logical design phase involves the development of tools and blueprints of the various information security policies.
- Backup and recovery processes and details of the organization's incidence response actions are laid out.
- Details of business response action to disaster are carefully planned.
- The decision as to whether the project is developed in-house or outsourced is reached during this phase.
- The physical design phase is the point at which the technical teams move into action.
- The information security technology that will be needed for the implementation of the all blueprints and analysis, detailed during the logical design phase, are evaluated and acquired.
- During this phase, alternative solutions, investigated for any unforeseen issues which may arise, are analyzed and mapped out.
- All the different teams at this point issue their stamp of approval of all processes and the green light is given to proceed.

SDLC phases and security mandates...

- **Which phase of SDLC should security be integrated?**
- A better practice is to integrate security activities across the SDLC—from the planning phase to release. This helps discover (and fix!) defects close to the time they're introduced.
- **What is security SDLC explain its different phases?**
- The cycle consists of a number of phases including systems investigation, systems analysis , logical design, physical design, implementation and maintenance and testing.

SDLC phases and security mandates...

- Once implementation is done, the security of the system and data, depend on the maintenance and testing phase which spans the life of the project.
- **What is SDLC in information security?**
- SDLC is the acronym for the framework Software Development Life Cycle, also referred to as secure development lifecycle. This framework helps developers and system engineers build applications and information systems by defining work phases and tasks.

Penetration Testing assessments

- Vulnerability Assessment and Penetration Testing (VAPT) are two types of vulnerability testing. The tests have different strengths and are often combined to achieve a more complete vulnerability analysis.
- Vulnerability assessment tools discover which vulnerabilities are present, but they do not differentiate between flaws that can be exploited to cause damage and those that cannot.

Penetration Testing assessments

- Vulnerability scanners alert companies to the pre-existing flaws in their code and where they are located. Penetration tests attempt to exploit the vulnerabilities in a system to determine whether unauthorized access or other malicious activity is possible and identify which flaws pose a threat to the application.
- Penetration tests find exploitable flaws and measure the severity of each. A penetration test is meant to show how damaging a flaw could be in a real attack rather than find every flaw in a system
- Together, penetration testing and vulnerability assessment tools provide a detailed picture of the flaws that exist in an application and the risks associated with those flaws.

Penetration Testing assessments...

What is The **VAPT Methodology**?



Establishing Scope:

Maintain a checklist to assess a target application or network.

Vulnerability Assessment and Detection:

Carry out manual assessment or use testing tools.



Penetration Testing:

Verify potential vulnerabilities using penetration tests.

Reporting and Documentation:

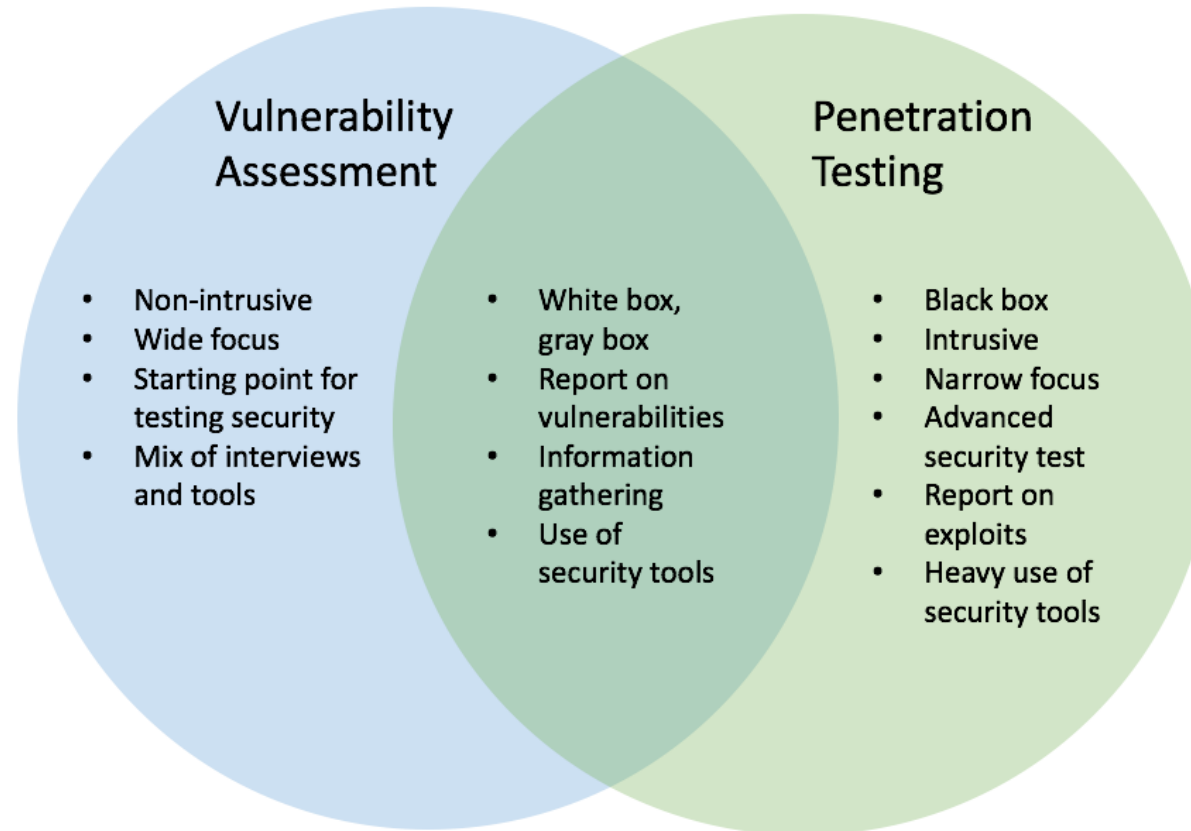
Analyze and document the outcome of a potential threat and give recommendations.



Penetration Testing assessments...

Vulnerability Assessment	Penetration Testing
This is the process of finding and measuring the vulnerability of a system	Penetration testing finds the vulnerabilities and exploits it to take advantage of the system
The end result is a list of vulnerabilities which is often prioritized by its potency	A penetration test is more goal oriented. It helps in charting the path which will be taken by the attacker to take over the system.
Vulnerability assessment is recommended when the system already has known security issues or the organization has no security measures and wants to get started in that area.	Penetration test, on the other hand, is recommended when the company has a good level of security and they want to search for some hidden vulnerabilities.
Emphasizes “breadth over depth”. Meaning it is more concerned about finding more vulnerabilities instead of understanding the true severity of each.	Emphasizes “depth over breadth”. They discover vulnerabilities with specific goals in mind. They want to know how a potential hacker can exploit the situation to take over the system.

Penetration Testing assessments...



Penetration Testing assessments...

Difference between Penetration Testing vs Vulnerability Assessment ?

To some extent, the fundamental difference between vulnerability assessment and penetration testing is that vulnerability assessment (VA) is list-oriented and penetration testing is goal-oriented approach. Vulnerability assessment intends to identify vulnerabilities in given web application, network system or environment. Basically, VA identify network and application vulnerabilities before they turn into real threats to your corporate security. Whereas the purpose of penetration testing is to determine whether a detected vulnerability is practical and capable to harm the system. Vulnerability assessment: Uncovers a wide range of possible vulnerabilities. Penetration testing: A “call to action” document. It lists the vulnerabilities that were successfully exploited.

Penetration Testing assessments...

What is vulnerability assessment report ?

A vulnerability assessment process is intended to discover the potential security threats and the risks which involves the use of automated testing tools, such as network security scanners. The VAR vulnerability assessment report contains the list of vulnerabilities.

How do you test for penetration testing?

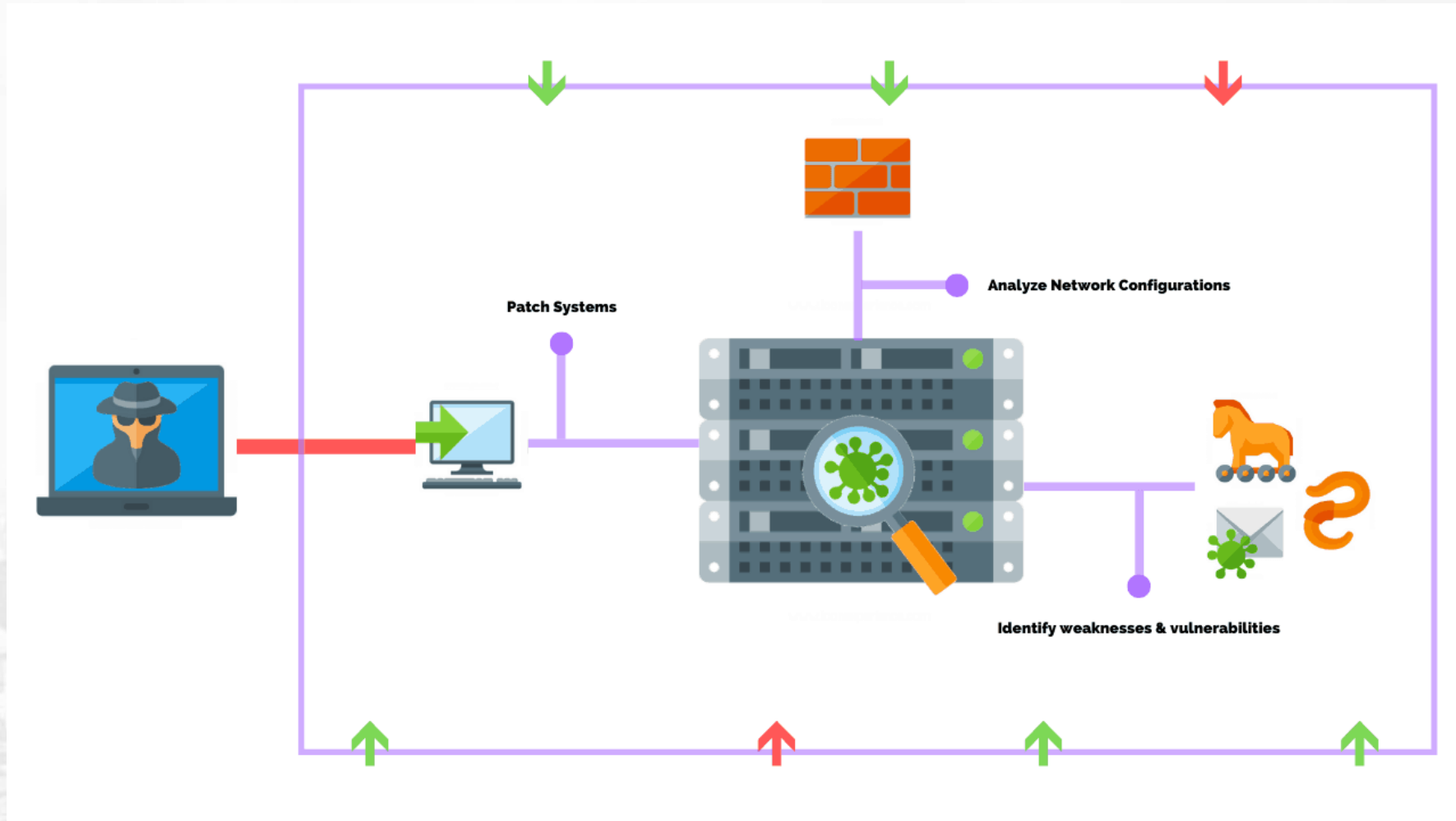
A pen test can be automated using security tools or it can be performed manually. In order to provide insights critical to the organization's ability to fine-tune its security policies and patch detected vulnerabilities, penetration tests need to expose vulnerabilities that would allow attackers system access.

Detect and respond to network breaches found in a Penetration Testing assessments

A network penetration test is often the next step to validate the risk assessment with the goal of enhancing a business's security posture.

There are four main steps to performing a network penetration test which include 1) information gathering and clarifying client expectations, 2) reconnaissance and discovery, 3) performing the penetration test, and 4) reporting on recommendations and remediation.

Detect and respond to network breaches found in a Penetration Testing assessments...



Detect and respond to network breaches found in a Penetration Testing assessments...

What Is A Network Penetration Test?

A network penetration test is the process of identifying security vulnerabilities in applications and systems by intentionally using various malicious techniques to evaluate the network's security, or lack of, responses. Similar to vulnerability assessments, a network penetration test, also known as a pen test, aims to identify vulnerabilities in a network.

However, unlike a vulnerability assessment, a penetration test is an exact simulation of a potential attack to identify vulnerabilities that are harder to find in a network.

Detect and respond to network breaches found in a Penetration Testing assessments...

What are The Benefits of Performing A Network Penetration Test?

There are numerous benefits to performing network penetration tests on your systems including:

- Understanding the network baseline
- Testing your security posture and controls
- Preventing network and data breaches
- Ensuring network and system security

Detect and respond to network breaches found in a Penetration Testing assessments...

- Understand The Network Baseline

Most of the time, the network's baseline is identified through the use of scanning tools like port scanners, network scanners, and vulnerability scanners. Understanding a network's baseline allows the business owner to understand what security controls are working, identify existing vulnerabilities, and provide them additional information about their network.

- Test Your Security Posture And Controls

Unlike a vulnerability assessment, a network penetration test will put your security controls to the ultimate test. A network penetration test's goal is to breach your network and exploit those vulnerabilities to understand the areas that need improvement.

- Prevent Network And Data Breaches

When a successful penetration test is performed, the results assist a business owner in designing or adjusting their risk analysis and mitigation strategies. This helps the business prevent future breaches because the network penetration test simulates a real-world attacker attempting to break into your systems.

Detect and respond to network breaches found in a Penetration Testing assessments...

Ensure Network and System Security

A network penetration test helps to ensure system security in a variety of ways.

For example, a business may have a mature security strategy with strong external defenses but their internal defenses, such as a host-based Intrusion Prevention System (IDS) that prevent attacks from trusted hosts on the network, have been neglected.

Detect and respond to network breaches found in a Penetration Testing assessments...

To perform a successful penetration test, there are 4 steps that must be completed:

Step 1: Information Gathering and Client Expectations

When you are discussing the goals of the network penetration test, there are a few important things to consider.

Penetration tests fall into three main categories:

Black box testing

Gray box testing

White box testing

Detect and respond to network breaches found in a Penetration Testing assessments...

Black Box Testing

A network penetration test that is performed from the position of an average hacker, with minimal internal knowledge of the system or the network, is known as black box testing.

This type of test is typically the quickest as it employs tools to identify and exploit vulnerabilities in the outward-facing network. It is important to note that if the perimeter cannot be breached in this type of penetration test, any internal vulnerabilities will remain undiscovered.

Gray Box Testing

A network penetration test that is performed from the position of a user, that has access to the system, potentially including elevated privileges, is known as gray box testing. This type of test aims to provide a more focused assessment of the network's security, with insights into the external and internal vulnerabilities.

Detect and respond to network breaches found in a Penetration Testing assessments...

White Box Testing

A network penetration test that is performed from the position of an IT or IS user, that has access to the source code and architecture documentation, is known as white box testing. This type of penetration test typically takes the longest, with the most challenging aspects being the large amounts of data that must be scrutinized to identify vulnerabilities.

It is important to know the types of network penetration tests that can be performed, whether you are a penetration tester or a business owner because they all provide specific benefits to the businesses.

Detect and respond to network breaches found in a Penetration Testing assessments...

Step 2: Reconnaissance and Discovery

Now, it is time for you to put your penetration tester hat on.

After you have discussed the goal of the network penetration test, including the information that will be used during the test and the time and date in which it will occur, the reconnaissance and discovery step begins.

Detect and respond to network breaches found in a Penetration Testing assessments...

Reconnaissance

During your reconnaissance, you will begin by employing port and network scanners on the network and systems to get a view of the network, the devices on the network, and existing vulnerabilities. Your goal will be to see where the vulnerabilities are located in order to begin your exploitation of those vulnerabilities. Social engineering, the use of deception to manipulate individuals into divulging confidential or personal information that may be used for fraudulent purposes, could be used as a tactic to identify vulnerabilities in the network that will allow you to gain access more easily.

Discovery

Discovery is when you find the information that you were seeking during reconnaissance. By aggregating the information found a path can be identified to breach the network.

Now, let's put this step into action. During a gray box penetration test on a client's network, tools like a port scanner, a tool that identifies open ports on a system, and a vulnerability scanner, a tool that identifies vulnerabilities on a system, are used to begin to identify ways to gain access to the network.

Detect and respond to network breaches found in a Penetration Testing assessments...

Step 3: Performing the Network Penetration Test

During step 3, the pen tester will perform the network penetration test based on the vulnerabilities that you identified in step 2.

This step often uses tools that include exploit scripts or custom scripts you may code yourself.

Preparation of a Penetration Test report

It is not necessary that an experienced penetration tester can write a good report, as writing report of penetration testing is an art that needs to be learnt separately.

What is Report Writing?

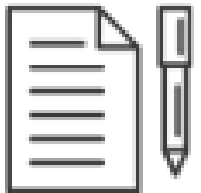
In penetration testing, report writing is a comprehensive task that includes methodology, procedures, proper explanation of report content and design, detailed example of testing report, and tester's personal experience.

Once the report is prepared, it is shared among the senior management staff and technical team of target organizations. If any such kind of need arises in future, this report is used as the reference.

Preparation of a Penetration Test report

REPORT

Relevant Findings



+

Prioritized Severity



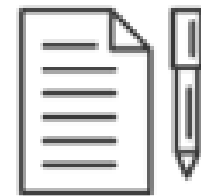
+

Accurate Descriptions



+

Actionable Mitigation



=

Value



Preparation of a Penetration Test report

Report Writing Stages

Due to the comprehensive writing work involved, penetration report writing is classified into the following stages –

- Report Planning
- Information Collection
- Writing the First Draft
- Review and Finalization

Preparation of a Penetration Test report



Preparation of a Penetration Test report

Report Planning

Report planning starts with the objectives, which help readers to understand the main points of the penetration testing. This part describes why the testing is conducted, what are the benefits of pen testing, etc. Secondly, report planning also includes the time taken for the testing.

Major elements of report writing are –

Objectives – It describes the overall purpose and benefits of pen testing.

Time – Inclusion of time is very important, as it gives the accurate status of the system. Suppose, if anything wrong happens later, this report will save the tester, as the report will illustrate the risks and vulnerabilities in the penetration testing scope during the specific period of time.

Target Audience – Pen testing report also needs to include target audience, such as information security manager, information technology manager, chief information security officer, and technical team.

Report Classification – Since, it is highly confidential which carry server IP addresses, application information, vulnerability, threats, it needs to be classified properly. However, this classification needs to be done on the basis of target organization which has an information classification policy.

Report Distribution – Number of copies and report distribution should be mentioned in the scope of work. It also needs to mention that the hardcopies can be controlled by printing a limited number of copies attached with its number and the receiver's name.

Preparation of a Penetration Test report

Information Collection

Because of the complicated and lengthy processes, pen tester is required to mention every step to make sure that he collected all the information in all the stages of testing. Along with the methods, he also needs to mention about the systems and tools, scanning results, vulnerability assessments, details of his findings, etc.

Writing the First Draft

Once, the tester is ready with all tools and information, now he needs to start the first draft. Primarily, he needs to write the first draft in the details – mentioning everything i.e. all activities, processes, and experiences.

Review and Finalization

Once the report is drafted, it has to be reviewed first by the drafter himself and then by his seniors or colleagues who may have assisted him. While reviewing, reviewer is expected to check every detail of the report and find any flaw that needs to be corrected.

Preparation of a Penetration Test report

Following is the typical content of a penetration testing report –

Executive Summary

- Scope of work
- Project objectives
- Assumption
- Timeline
- Summary of findings
- Summary of recommendation

Methodology

- Planning
- Exploitation
- Reporting

Detail Findings

- Detailed systems information
- Windows server information

References

- Appendix

Preparation of a Penetration Test report

What should a penetration test report include?

Publisher Summary

An effective penetration testing report should include an executive summary, a detailed report, and raw output. The executive summary should be a very brief overview of the major findings.

How should you prepare for the Pen Test?

- Identify and communicate your scope and objectives with the security professionals conducting your pen test.
- Decide on the best time to conduct the test.
- Backup your data.
- Ensure that your internal IT team is available.
- Explain what you want to see in the report.

Preparation of a Penetration Test report

What is the final stage of a penetration test?

A pentester will often use a vulnerability scanner to complete a discovery and inventory on the security risks posed by identified vulnerabilities. Then the pentester will validate if the vulnerability is exploitable. The list of vulnerabilities is shared at the end of the pentest exercise during the reporting phase.

Auditing the Systems

Audit

The term audit is derived from the Latin term ‘audire,’ which means to hear. In early days an auditor used to listen to the accounts read over by an accountant in order to check them. Auditing of a system or system processes is carried out against agreed upon requirements. Such an audit is carried out in order to verify that the individual processes within the system are effective and suitable in achieving the stated objectives. The audit is used to verify whether the system / the processes are operating within the specified limits and achieving the specified targets (objectives). The system / process audit examines the process activities / steps for verifying whether the inputs, actions, and the outputs are in accordance with the defined requirements. A system / process audit is an evaluation of the sequential steps and techniques of the process within the system. Auditing of a system or system processes provides value for the management by the evaluation of the processes, their control, risks, and the achievement of the objectives.

Auditing the Systems

System audit is defined as ‘a systematic and independent examination to determine whether activities and related results comply with the planned arrangements and whether these arrangements are implemented effectively and are suitable to achieve objectives’. It is also defined as ‘a systematic, independent, and documented process for obtaining audit evidence and evaluating it objectively to determine the extent to which audit criteria are fulfilled’.

One of the main differences between the process audit and a system audit is the scope definition and expansion. A process audit can be a singular process or a part of a process. Process audit can start at any level of the process where work takes place.

Auditing the Systems

Referring to the 'control levels triangle', it can be seen that the process audit can start from level 4 and go up to the top, while the system audit starts from the top (level 1) and goes down. A system or a sub-system audit is against the agreed upon requirements. Top level requirements drive the formation of subsystems and processes for meeting the requirements.



Auditing the Systems

Terms used in a system audit

Various terms used during the auditing of system and processes are described below.

- **Audit** – A planned and documented audit performed in accordance with manual, procedures, records, and other documents like checklists etc. for the intended purpose of verifying applicable elements of a system and processes and its implementation.
- **Audit plan**– It is typically an audit action plan based on the applicable audit requirements in the standards / norms for the system / processes being audited and the audit report summary, with additional questions / issues which are to be verified included in or attached to these documents as needed to ensure objectivity and impartiality. It can also be a marked up copy of the procedure / process documentation, identifying evidence to be collected to verify conformance.

Auditing the Systems

- **Auditor**— A qualified and trained person who is authorized to perform specific audit functions under the direction of a lead auditor.
- **Audit coordinator** – A person with responsibility / authority for scheduling audits, selecting auditors (ensuring objectivity and impartiality), and ensuring issues raised are effectively addressed.
- **Effectiveness**— It is the evidence, including the relationship with inputs and outputs for the process. It shows the process is working, driving performance, and supporting the organization's policy, objectives, and compliance with requirements (laws, regulations, etc.).

Finding— It is an issue needing resolution. It can be an actual problem (something requiring corrective action), a potential problem (something requiring preventive action), or any other opportunity for improvement (including those making it better and / or helping to be more fiscally responsible).

Auditing the Systems

- **Internal auditor**— A qualified and trained person of the organization who performs audit of system or processes, reports non-conformances and observations, evaluates the adequacy of corrective and preventive actions, and reports audit findings to the organization management.
- **Lead auditor** — A qualified and trained and certified person, who is authorized to plan, organize, and direct audit of system and processes of an organization, to report non conformances and observations, and to evaluate the adequacy of corrective and preventive actions.
- **Non-compliance**— It is the evidence which indicates the organization is not complying with a regulation, rule, or requirement where compliance is mandatory (i.e., law, corporate policy, etc.).
- **Non-conformance**— It is the evidence which indicates the actions by those fulfilling a process and the information in supporting documentation do not conform to one another and / or requirements outlined in the standards.

Auditing the Systems

Types of audits

There are several types of audits as described below.

- **Adequacy audit** – It is the audit exercise which determines the extent to which the documented system, represented by the manual, the associated procedures, work instructions and record forms adequately meets the requirements of the system and processes and if it provides objective evidence that the system and the processes are correctly designed in this respect.
- **Compliance audit** – It is the audit which determines the extent to which the documented system and processes are implemented and observed within the organization.
- **External audit** – It is an audit carried out for the system / processes of the organization with whom there is a contract to purchase goods or services or intend to do so. It can be adequacy and / or compliance audit or both. It is also known as second party audit.

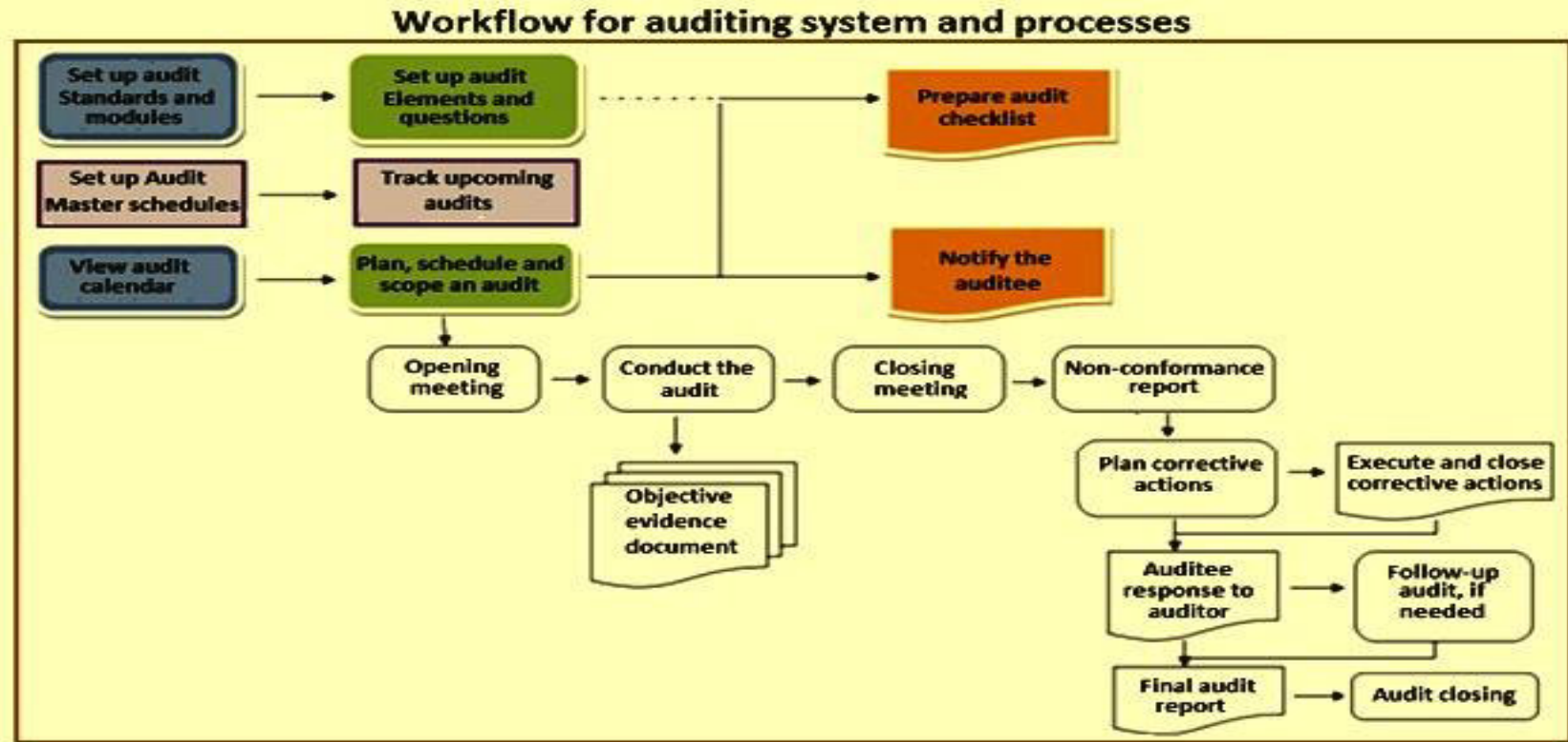
Auditing the Systems

- **Extrinsic audit** – It is an external audit carried out by an independent accredited third party using a standard to provide assurance on the effectiveness of the system and processes. This audit can also be adequacy and / or compliance audit or both. It is also known as third party audit.
- **Internal audit** – It is an audit which is carried out by the organization from its own internal sources for its system and processes for the purpose of providing assurance to the management that the system and processes are functioning properly and are effectively achieving the planned objectives. These audits are carried out by those employees of the organization who are not directly involved in the system and processes. Sometimes organizations take the help of external agencies for carrying out the internal audit. It is also known as first party audit.

Auditing the Systems

- **Process or product audit** – It is a vertical audit which looks into complete system that goes into the production of a specific end product or service.

Process of Auditing



Workflow for auditing system and processes

Auditing the Systems

The process of auditing can be divided into the following steps.

Audit initiation – It defines the scope and the frequency of the audit. The scope of the audit is determined on the needs of the organization and a decision is made with respect to system's elements such as activities, departments and locations etc. which are to be audited within a time frame. This is normally done along with the lead auditor. The frequency of the audit is determined after considering specified or regulatory requirements and any other pertinent factors. Both internal and external audits are to be part of the audit schedule. The frequency of the internal audits is normally much more than the external audit since it provides input to the management not only about the normal functioning of the system but also inputs for the decision making.

Auditing the Systems

Audit preparation – As a basis for planning the audit, the auditor is to review the manual and the auditing procedure of the system and if there is any inadequacy it is to be resolved first. After this an audit plan / programme is to be made along with the auditee. This programme is to be approved and after approval it is to be communicated to the auditors and the auditees. This plan is to include (i) the objective and scope of the audit along with the activities to be audited, (ii) the persons who are directly responsible for the audited activities and the audit scope is to be identified with them, (iii) reference documents such as the system standard and system manual etc. are to be identified on which the audit is to be conducted, (iv) the team members for the audit are to be finalized, (v) the date, time, and the place of the audit is to be finalized

Auditing the Systems

(vi) the units of the organization are to be finalized, (vii) the expected time and duration of each of the audit activity is to be decided, (viii) the schedule of meetings with the management need to be finalized, (ix) audit is to fulfill the requirement of the confidentiality if any is there in the system and the processes, (x) the language of the audit is to be decided, and (xi) the distribution of the audit report to be finalized. All the documents needed for the audit are to be made available to the auditors to facilitate auditing. The auditors are to prepare also a check list to assist them during conducting of the audit. A further audit is sometimes necessary to check the corrective actions taken on a non conformity report (NCR).

Auditing the Systems

Audit execution – A structured audit is having the following four execution steps.

- An opening meeting – It is chaired by the lead auditor where he introduces the team members to the auditees, confirms the arrangements made for the audit, briefs the auditees about the audit details, explain to the auditees difference between major and minor NCRs, ensures that the guides are available during the auditing, explain the timings for daily liaisoning meetings and the final closing meeting. The opening meeting is to include the senior management and all the persons involved in the audit.
- The examination and evaluation of the system – The audit is to cover entire scope and is to run to the plan. During the audit clear and precise NCRs are to be raised based on the sound objective evidence. Regular liaisoning meeting are to be held.

Auditing the Systems

- A closing meeting – Like the opening meeting this meeting is also to be chaired by the lead auditor. It is held at the end of the audit. In preparation of this meeting auditors explain their findings during the audit to team members and these findings are reviewed and the actions to be taken on these findings are taken. During closing meeting the lead auditors briefs about the audit scope, and tells the findings of the audits. The NCRs noticed during the audit are explained by the team members and are handed over to the auditees. Team leader give an overall summary of the findings and the conclusions including the actions to be taken are recommended.
- The audit report – This report is handed over during the closing meeting.

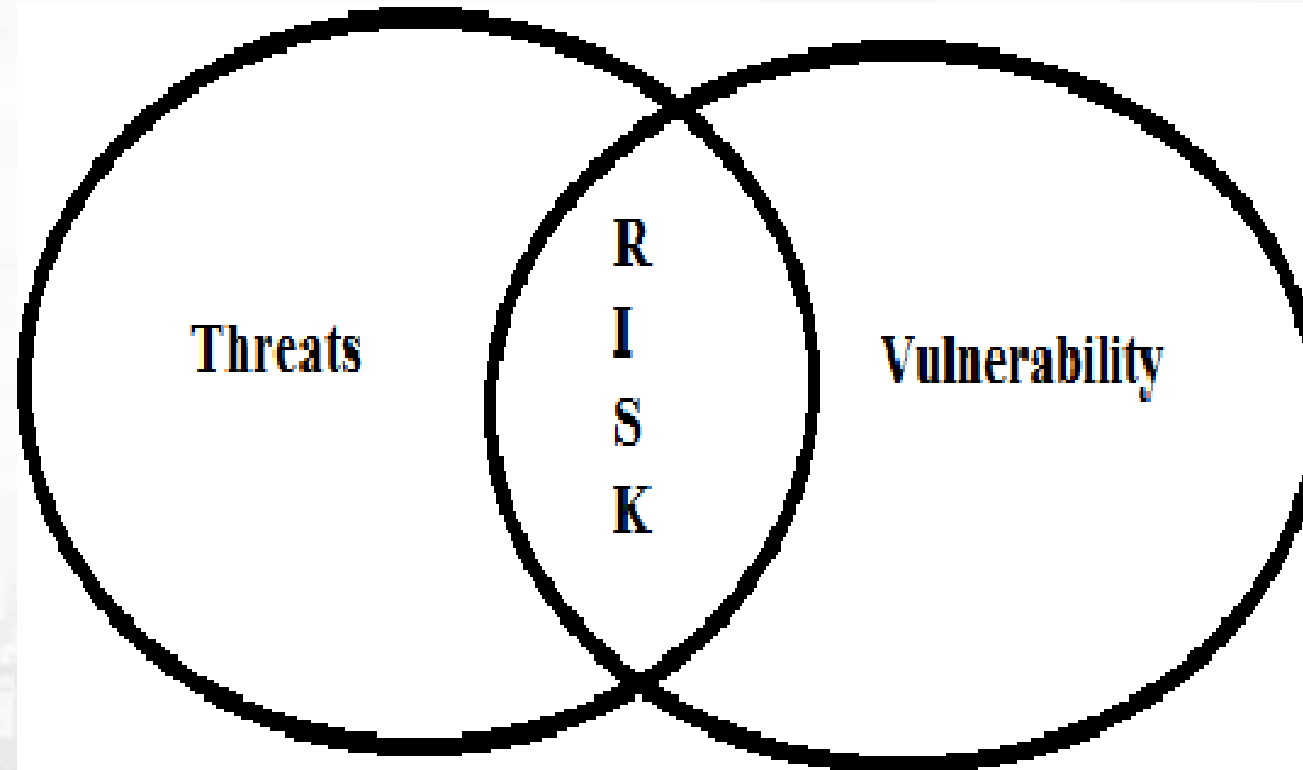
Auditing the Systems

Audit report – The lead auditor has the responsibility of the preparation of the audit report. The audit report is to faithfully reflect the tone and the conduct of the audit. It is also to be signed with date by the lead auditor. The audit report is to contain only factual statement of discrepancies supported by the objective evidences. The audit report is to include, if applicable, such items as (i) the scope and the objective of the audit, (ii) details of the audit plan, (iii) the standard and any other document against which the audit was conducted, (iv) observations of the non conformity reports, (v) audit team's judgment to the extent of the compliance with the applicable standards and other documents, (vi) the ability of the system to achieve the objectives, and (vii) the distribution list for the audit report. Any communication made between the closing meeting and the issue of the report should be made by the lead auditor in the report.

Analysis and Reporting

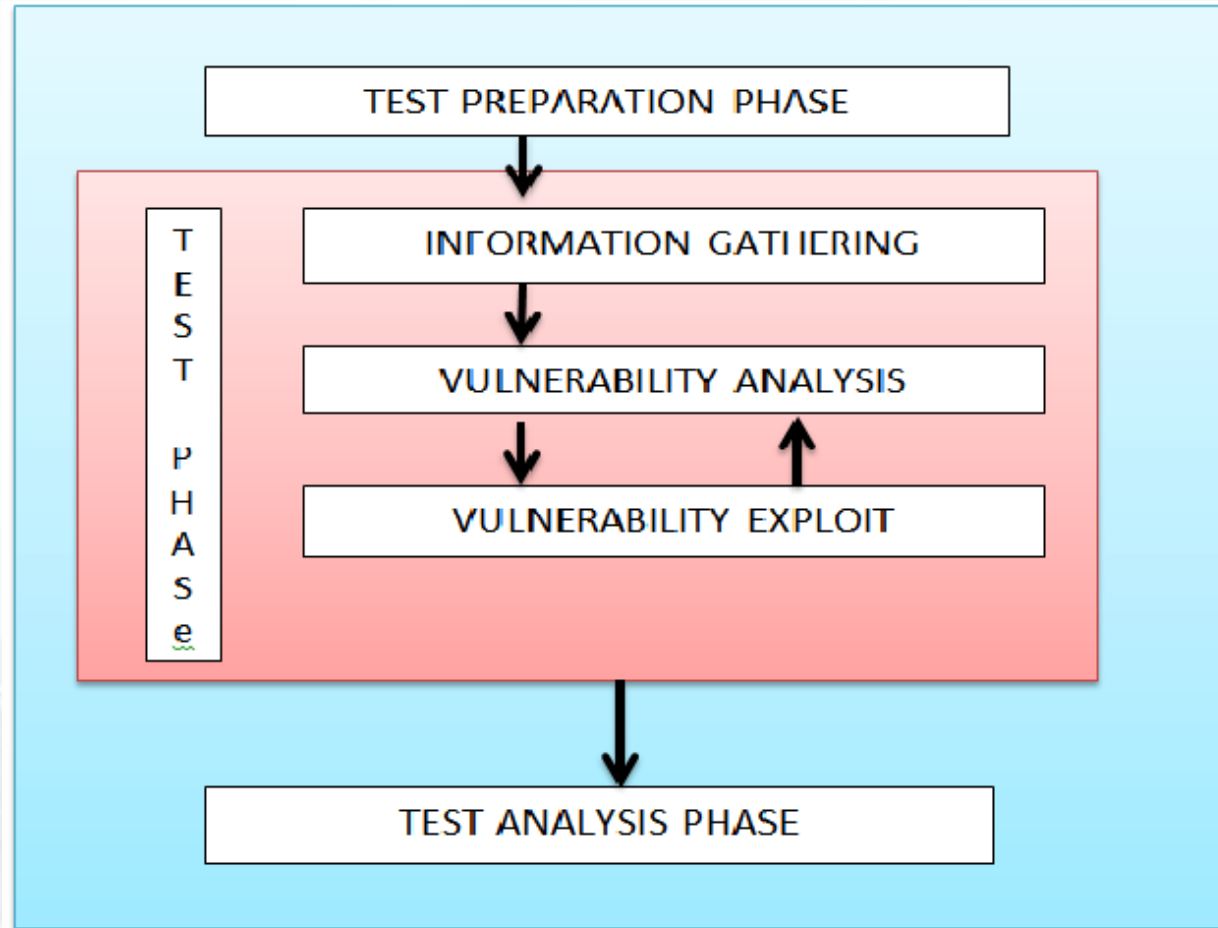
Web application is becoming so popular and significant part of our daily lives. Due to the use of web applications increasing day by day, the web application security is becoming vital for user's secret data. In parallel to this, the number of reported web application vulnerabilities is increasing dramatically. Most of the vulnerabilities are the result of improper input validation. This paper discuss the Tainted Mode Model (TMM) which allows inter module vulnerabilities detection. Besides, the seminar presents a new approach to vulnerability analysis which incorporates advantages of penetration testing and dynamic analysis. This approach effectively utilizes the extended Tainted Mode Model.

Analysis and Reporting



Visualization of vulnerability in the application

Analysis and Reporting



Penetration Testing Methodology

Analysis and Reporting

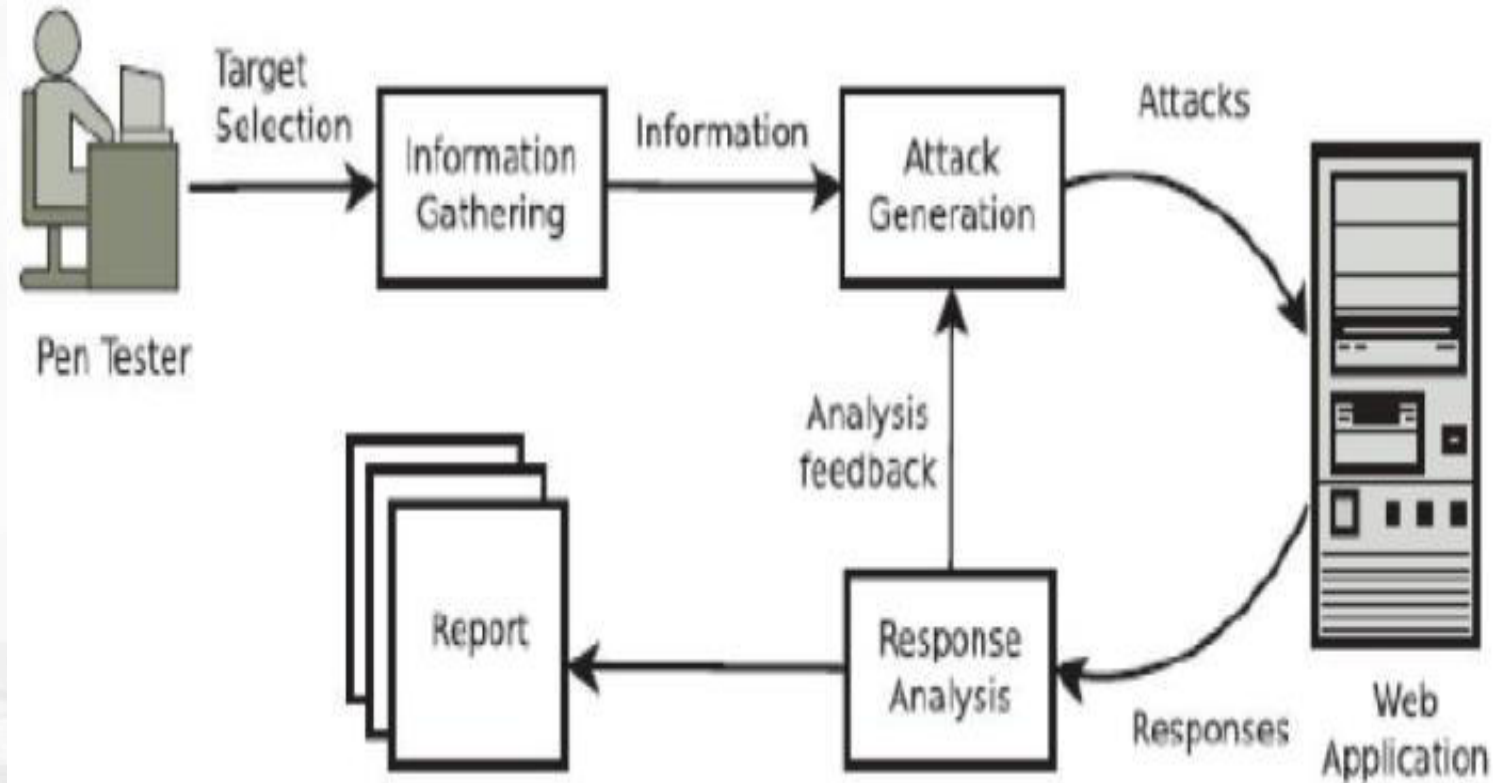


Figure 2 Penetration Testing Process [2]

Penetration Testing Process

Analysis and Reporting

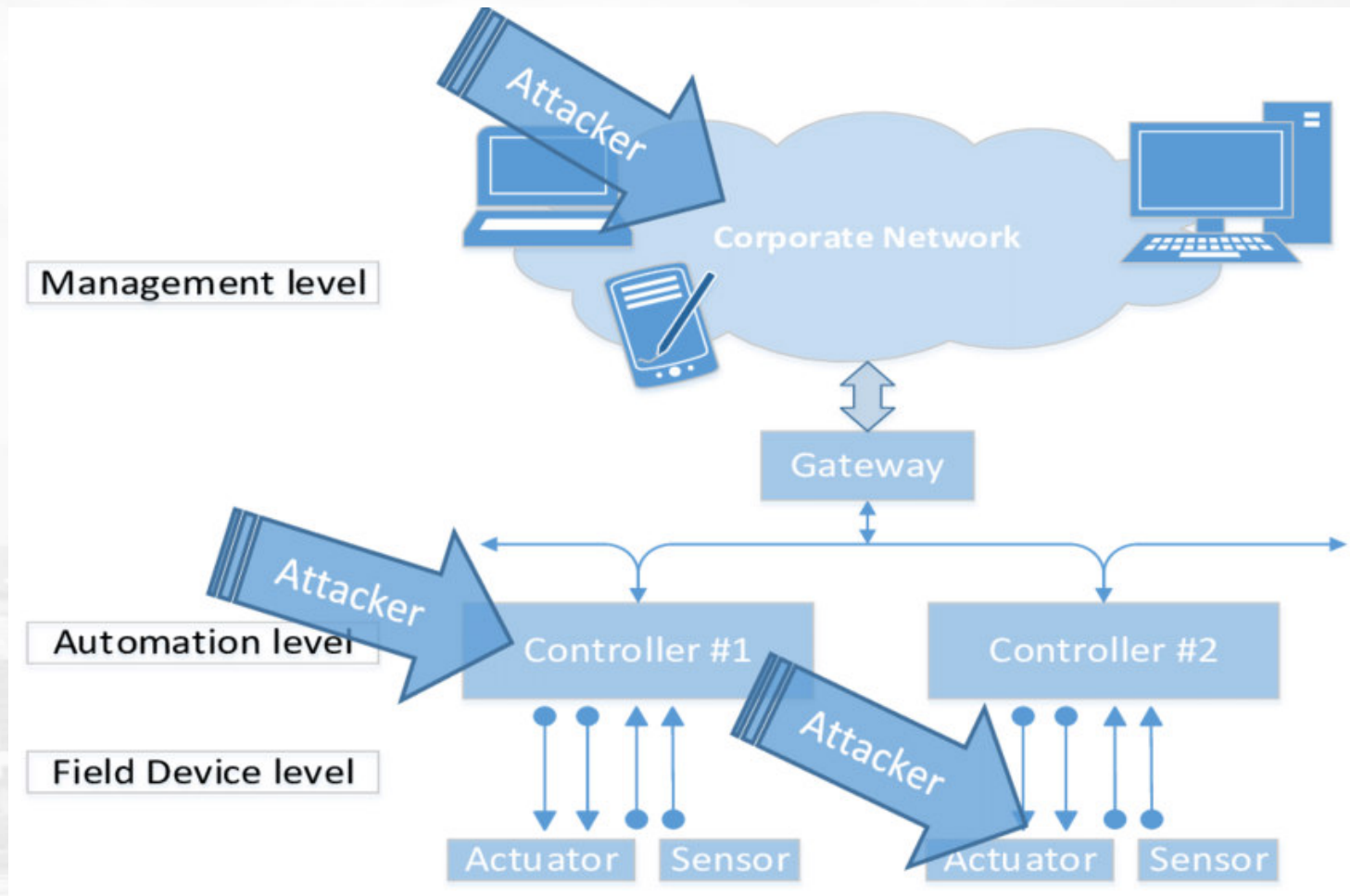
What is reporting in penetration testing?

A Penetration Testing report is a document that contains a detailed analysis of the vulnerabilities uncovered during the security test. It records the weaknesses, the threat they pose, and possible remedial steps.

Why report writing is major component of penetration testing?

A well-skilled penetration tester not just finds the weaknesses but also explains their impact on the customer. It is important to write a report with real added value. The report should provide the customer with realistic solutions to the risks identified.

Case Studies of recent vulnerabilities and attacks



Case Studies of recent vulnerabilities and attacks

The OWASP Top 10 list is only a literal tip-of-the-iceberg representation of the increasing number of cyber threats facing us today. This list is expected to change as we see more transformations in the ways we work, play, and live our lives in these interesting times.

For instance, increased adoption of the cloud and the advent of new technologies like 5G will likely present more attack surfaces and therefore more cybersecurity challenges to organizations and individuals.

At the end of the day, cybersecurity affects all of us. We should therefore have an awareness of the potential harm to our digital lives.