Penetration Testing Approaches:

Penetration testing can be conducted in **two ways**: **Manual Testing** and **Automated Testing**.

Manual Penetration Testing:

Manual penetration testing is the testing that is done by human beings. In such type of testing, vulnerability and risk of a machine is tested by an expert engineer.

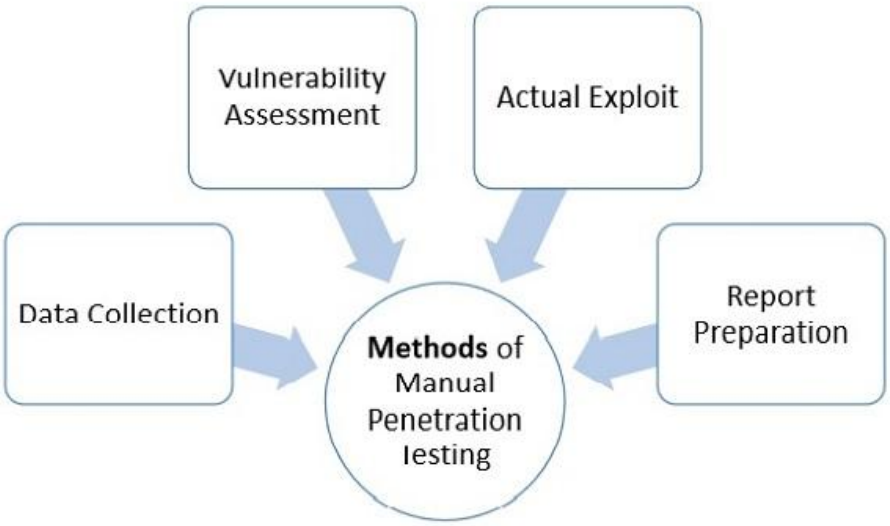
Generally, testing engineers perform the following methods −

Data Collection − Data collection plays a key role for testing. One can either collect data manually or can use tool services (such as webpage source code analysis technique, etc.) freely available online. These tools help to collect information like table names, DB versions, database, software, hardware, or even about different third party plugins, etc

Vulnerability Assessment − Once the data is collected, it helps the testers to identify the security weakness and take preventive steps accordingly.

Actual Exploit − This is a typical method that an expert tester uses to launch an attack on a target system and likewise, reduces the risk of attack.

Report Preparation − Once the penetration is done, the tester prepares a final report that describes everything about the system. Finally the report is analyzed to take corrective steps to protect the target system.



Types of Manual Penetration Testing

Manual penetration testing is normally categorized in two following ways −

Focused Manual Penetration Testing − It is a much focused method that tests specific vulnerabilities and risks. Automated penetration testing cannot perform this testing; it is done only by human experts who examine specific application vulnerabilities within the given domains.

Comprehensive Manual Penetration Testing − It is through testing of whole systems connected with each other to identify all sorts of risk and vulnerability. However, the function of this testing is more situational, such as investigating whether multiple lower-risk faults can bring more vulnerable attack scenario, etc

Automated Penetration Testing

Automated penetration testing is much faster, efficient, easy, and reliable that tests the vulnerability and risk of a machine automatically. This technology does not require any expert engineer, rather it can be run by any person having least knowledge of this field.

Tools for automated penetration testing are Nessus, Metasploit, OpenVAs, backtract (series 5), etc. These are very efficient tools that changed the efficiency and meaning of penetration testing.

Scenario	Manual	Automated
Large-scale testing	❌ No	✅ Yes
Finding deep security flaws	✅ Yes	❌ No
Quick security scanning	❌ No	✅ Yes
Testing business logic vulnerabilities	✅ Yes	❌ No
Reducing false positives	✅ Yes	❌ No

Testing Types: External to Internal & Internal to External

The terms **External to Internal** and **Internal to External** testing are commonly used in cybersecurity, network security, and penetration testing to describe different perspectives of testing an organization's security posture.

1. External to Internal Testing

This type of testing simulates an **attacker from outside** the organization's network, trying to gain unauthorized access to internal systems.

- **Purpose:** Identify vulnerabilities in externally facing systems (e.g., web applications, firewalls, VPNs).
- **Attackers:** Ethical hackers, penetration testers, or adversaries with no prior access.
- **Focus Areas:**
    - Open ports & services on public-facing servers.
    - Weak passwords, misconfigured firewalls, and exposed APIs.
    - Web application vulnerabilities (SQL injection, XSS, etc.).
    - Social engineering attempts like phishing.

2. Internal to External Testing

This type of testing evaluates security from the perspective of an **insider** (a compromised system or a malicious employee) trying to exfiltrate data or escalate privileges.

- **Purpose:** Assess security controls preventing data leakage and insider threats.
- **Attackers:** Malicious insiders, compromised accounts, or malware.
- **Focus Areas:**
    - Lateral movement within the internal network.
    - Access controls, privilege escalation.
    - Data exfiltration via email, USB, or external servers.
    - Detection and response mechanisms (SIEM logs, firewall rules).

Non-Destructive vs. Destructive Testing Techniques

1. Non-Destructive Testing (NDT)

This testing checks for vulnerabilities **without causing damage** to the system, ensuring normal operations continue.

- **Goal:** Identify security weaknesses safely.
- **Examples:**
  - Passive scanning (observing traffic without interfering).
  - Vulnerability scanning (checking for known flaws).
  - Social engineering (testing security awareness).
  - Network monitoring (analyzing logs and activities).

2. Destructive Testing

This testing actively **exploits vulnerabilities** to see how a system responds, which may cause downtime or damage.

- **Goal:** Test real-world attack impact.
- **Examples:**
  - Penetration testing (actively exploiting weaknesses).
  - DDoS simulations (overloading systems).
  - Malware injection (testing endpoint security).
  - Crash testing (forcing system failures).

Penetration Testing Reports: Structure & Importance

1. Importance of a Penetration Testing Report

A **Penetration Testing Report** is crucial because it:

✅ **Identifies security weaknesses** before attackers do.

✅ **Provides a risk assessment** with real-world attack scenarios.

✅ **Helps organizations fix vulnerabilities** with clear recommendations.

✅ **Ensures compliance** with security standards (ISO 27001, PCI-DSS, etc.).

2. Structure of a Penetration Testing Report

A well-structured report typically includes:

1️⃣ **Executive Summary**

- Overview of findings in simple terms.
- Risk rating (e.g., High, Medium, Low).
- Key vulnerabilities and their impact.

2 Scope & Methodology

- What was tested (networks, applications, systems).
- Testing approach (black-box, white-box, gray-box).
- Tools & techniques used.

3 Findings & Vulnerabilities

- List of security flaws with severity levels.
- Screenshots, logs, or proof of concept (PoC).
- Possible attack scenarios.

4 Risk Assessment

- Business impact of each vulnerability.
- Likelihood of exploitation.
- CVSS (Common Vulnerability Scoring System) scores.

5 Recommendations & Mitigation Steps

- How to fix each issue (patches, configurations, policies).
- Prioritization of fixes.

6 Conclusion & Next Steps

- Summary of overall security posture.
- Suggested improvements (e.g., security awareness training, continuous monitoring).

7 Appendices (Optional)

- Detailed logs, scripts, raw data.
- References to security guidelines.