Lifecycle of a Penetration Test

4.2 Lifecycle of a Penetration Test

+ **How does this support my pentesting career?**

  - Become a real pentester, not just a skilled hacker
  - Understand the role of a penetration test in a corporate environment
  - Be able to perform effective pentests

4.2 Lifecycle of a Penetration Test

- A Penetration Test is both a **complex** and a very **delicate** process.

- You have to thoroughly test your client's systems to find **any and every vulnerability** while, at the same time, you must guarantee that your activity will have the least impact possible on the production systems and services; this is crucial and is the difference between a **real professional** and an amateur.

4.2 Lifecycle of a Penetration Test

- It is important to carefully select the right tools and techniques to use during your tests to avoid **overloading your client systems and networks.**

- Deep understanding of what you are doing also allows you to communicate to your client what steps to take should anything go wrong during the pentest.

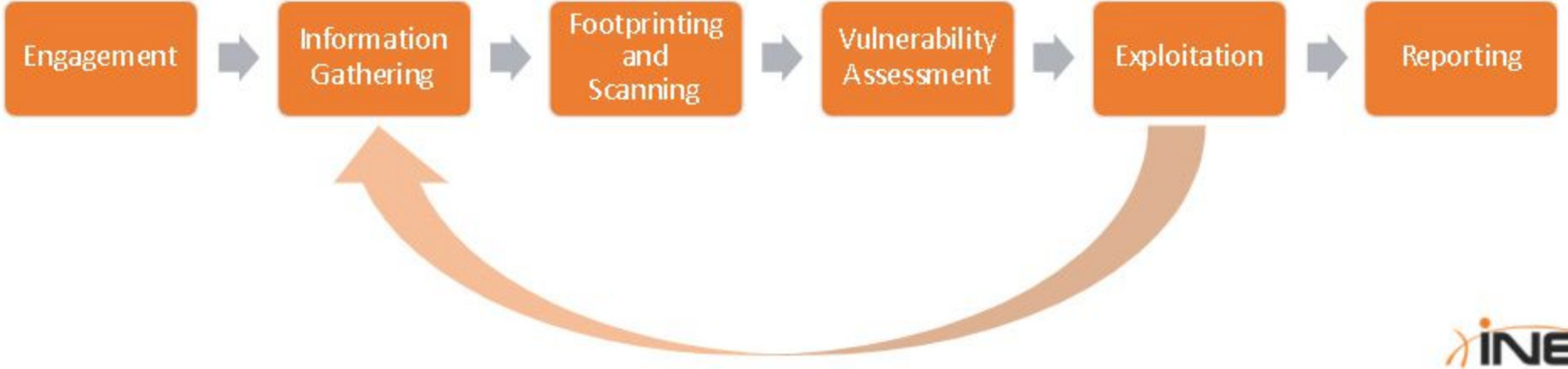
4.2 Lifecycle of a Penetration Test

+ Considering the penetration test as a **process**, rather than an unstructured block of tasks, ensures that every potential vulnerability or security weakness gets tested, with the lowest possible overhead.

+ As you will see in a moment, the success of a task depends on the success of the preceding tasks.

4.2 Lifecycle of a Penetration Test

+ Let's now look at every phase of the **penetration testing process**. Do not underestimate the value of every step!



4.2.1 Engagement

- All the details about the penetration test are established during the **Engagement** phase.

4.2.1.1 Quotation

- At the **Quotation** stage, a professional pentester defines the fee for the penetration test of a network, a web application or the whole organization.

- The fee will vary according to:
  - Type of engagement (Black Box, Gray Box, etc.)
  - How time-consuming the engagement is
  - The complexity of the applications and services in scope
  - The number of targets (IP addresses, domains, etc.)

4.2.1.1 Quotation

- Evaluating and quoting these aspects requires experience that you will gain in the field.
- If you are not able to quantify the amount of work required by an engagement, you can provide an hourly fee.

4.2.1.2 Proposal Submittal

- The best way to win a job is by providing a **sound and targeted proposal**.
- You should write the proposal keeping in mind the client's **needs** and **infrastructure**.

4.2.1.2 Proposal Submittal

+ The proposal should include:
  - The understanding of the client's needs. In other words, what you understood of their requirements.

  - The approach and methodology you want to use, like the use of automated scanning tools, manual testing, onsite testing and any other information that fits.

4.2.1.2 Proposal Submittal

- Furthermore, it should also include:
  - How you want to address their needs and what kind of value the pentest will bring to their business. Think in terms of **risks and benefits**, like business continuity, improved confidentiality, avoidance of money and reputation loss due to data breaches.

  - A quotation in terms of price and an estimate of the time required to perform your job.

4.2.1.2 Proposal Submittal

- Finally, any proposal must address:
  - The type of engagement. Is your activity a penetration test or vulnerability assessment? Is it remote or onsite?
    And so on.

  - **The scope of engagement** in terms of IP addresses, network blocks, domain names or any other information useful in defining the scope.

4.2.1.3 Staying in Scope

- As a professional penetration tester, you should be aware that your client might not have enough knowledge of some IT areas, especially when communicating the target to you.

4.2.1.3 Staying in Scope

- You should always make sure that the target of your engagement is the property of your client. Be careful especially when asked to perform an engagement (e.g., on a single website).

- If it is a part of shared hosting, you **must not** conduct an assessment on such a target unless you are given written permission from the hosting provider.

4.2.1.3 Staying in Scope

- Always analyze the target scope and verify **if it's your client's property** and if you have **written permission to conduct the assessment**.

- You should take any possible out of scope incidents very seriously; in many countries, such unauthorized activity might be considered **breaking the law**.

4.2.1.4 Incident Handling

- When conducting a penetration test, you should take into consideration that **incidents happen.**

- An **incident** is an unplanned and unwanted situation that affects the client's environment and disrupt its services.

4.2.1.4 Incident Handling

- Even when sticking to all of the best practices and performing every test very carefully, **there is always a likelihood of damaging the tested assets**, especially when you have little knowledge about the tested environment and cannot predict the result of every single operation.

4.2.1.4 Incident Handling

- **You should always aim not to damage the target.**

- In case of planning some intensive or risky tests, **you might want to communicate with the customer**. For instance, if there are some preferred hours when possible service stoppage will be less painful to them.

4.2.1.4 Incident Handling

- It is a best practise to have an **incident handling procedure**.
- Many large organisations already have such processes set up, while the smaller ones might not have implemented such procedures within them.

4.2.1.4 Incident Handling

- An **incident handling procedure** is a set of instructions that need to be executed by both you and your customer on how to proceed when an **Incident** (e.g., service damage or unavailability) occurs.

4.2.1.4 Incident Handling

- If there is no fixed procedure established by the client, the simplest way to handle an incident is to **have an emergency contact,** a technical person on the client's site **that is available** (via phone or another form of contact) that might coordinate further **incident handling** for the customer's company.

4.2.1.4 Incident Handling

- Once the emergency contact is set, it is worth adding a statement to the **rules of engagement**:

```
In case of technical inquiries regarding the target assets,
Pentester will contact bob@itservice.corp. In the event of
suspecting that a major incident took place (e.g., service
unavailability), Pentester will immediately contact Bob of
IT Service at phone number +12 345 678 90
```

4.2.1.5 Legal Work

- Once the previous steps are completed, you have to deal with the legal responsibilities of each party involved; this is done by producing some legal paperwork.

- Sometimes you will need to involve a lawyer as information security laws vary a lot from country to country. Other times, professional insurance is required, and it is strongly advised to have it as it only costs a few hundred dollars per year and can turn out to be very useful just in case.

4.2.1.5 Legal Work

- Companies usually want you to sign one or more Non-Disclosure Agreements (NDAs). These documents enforce your full confidentiality regarding any information or confidential data you may come across during your engagement.

- It does not matter if you have been exposed to private data, information on secret processes or products, it is your duty to **keep them private and encrypted on your PC**.

4.2.1.5 Legal Work

- With an NDA, a company ensures that you will not divulge any confidential information to any third party. Confidentiality is just one of the legal aspects of pentesting. Another key point is outlining what you **can and cannot do**.

- **All of the steps seen thus far apply if you are a Freelance Penetration tester. If you work for an IT Security services company, the legal department will deal with it, and your penetration testing process will start from the next step.**

4.2.1.5 Legal Work

- **Rules of Engagement** is another document that will define the scope of engagement and will put on paper what you are entitled to do and when; this includes the time window for your tests and your contacts in the client's organization.

4.2.1.5 Legal Work

- You will want these contacts (client's employees or managers) to coordinate activities, or to promptly communicate with if you accidentally break something during your tests.

- Once everything is clearly documented, you can move on to the practical part of the engagement, starting from information gathering.

4.2.2 Information Gathering

- **Information gathering** is the first and one of the most fundamental stages of a successful penetration test.

- Most beginners tend to overlook or rush this phase. If you want to perform an effective pentest; do not do that!

4.2.2.1 General Information

- Information gathering can start once the legal paperwork is complete but **not before** the beginning of the testing period. You don't want the client to find anything in their logs before that start date.

- During this stage, you are an investigator who wants to harvest information about the client's company.

4.2.2.1 General Information

- Such information includes:
  - Board of directors
  - Investors
  - Managers and employees
  - Branch location and addresses

  } Names and email addresses

- The above information is extremely useful if **Social Engineering** is allowed by the rules of engagement, as you will be able to mount effective targeted attacks.

4.2.2.2 Understanding the Business

- As the goal of a penetration test is to mimic the effects of a black hat hacker attack, you need to understand what are the risks involved and what are the client's critical infrastructures.

- Having an understanding of the business is a key aspect in understanding what is important for your client; this allows you to know what is critical and vital for the client, thus allowing you to rate the risks associated with a successful attack.

4.2.2.3 Infrastructure Information Gathering

- After collecting the *General Information* and you have an *Understanding of the Business*, the **Infrastructure Information Gathering can begin**.

- In this phase, you transform the IP addresses or the domains in scope into actionable information about servers, operating systems and much more.

4.2.2.3 Infrastructure Information Gathering

- If the scope is defined as a list of IP addresses, you can move on to the next step.
- If the scope is the whole company or some of their domains, you will have to harvest the relevant IP blocks by using **WHOIS** and other **DNS information.**

4.2.2.3 Infrastructure Information Gathering

- The goal of this phase is to give **meaning to every IP address in scope** by determining:
  - If there is a live host or server using it.
  - If there are one or more websites using that IP address.
  - What OS is running on the host or the server.

4.2.2.3 Infrastructure Information Gathering

+ This will help you:
    - Focus your efforts to actual live clients and servers.
    - Target your attacks.
    - Sharpen your tools for the exploitation phase, when you have to find out the vulnerabilities and the exploitability of the client systems.

4.2.2.4 Web Applications

- If there is any web application in scope, in this phase you will harvest:
  - Domains
  - Subdomains
  - Pages (website crawling)
  - Technologies in use, like PHP, Java, .NET and so on.
  - Frameworks and content management systems in use, like Drupal, Joomla, Wordpress, and others.

4.2.2.4 Web Applications

- You should treat web applications as completely separate entities, that require a separate study.
- You can gather information about web applications by browsing and inspecting through application proxies such as Burp.

4.2.3 Footprinting and Scanning

- During the **Footprinting and Scanning** phase, you deepen your knowledge of the in-scope servers and services.

4.2.3.1 Fingerprinting the OS

EXAMPLE

+ Fingerprinting the Operating System of a host not only gives you information about the OS running on the system, but also helps you narrow down the number of potential vulnerabilities to check in the next phases.

```
You would never check for a typical MS Windows
vulnerability on a Linux host!
```

4.2.3.1 Fingerprinting the OS

- There are tools that can make educated guesses about the OS, the version and even the patch level of a remote system.
- Those tools exploit some singularities you can find in the network stack implementation of every operating system.

4.2.3.2 Port Scanning

- After having detected and fingerprinted the live hosts, it's time for **port scanning**!

- With a scan of live hosts, you can determine which **ports** are open on a remote system; this is a crucial phase of the engagement because any mistake made here will impact the next steps.

4.2.3.2 Port Scanning

- Currently, the de facto port scanner is **nmap**.
- With `nmap`, a penetration tester can exploit different scanning techniques to reveal open, closed and filtered ports.
- You will see how `nmap` works in the penetration testing part of the course.

http://nmap.org/

4.2.3.3 Detecting Services

- Knowing that a port is open is just half of the job.
- Next, you will need to know what is the service listening on that port!

4.2.3.3 Detecting Services

- In fact, knowing just the port is not enough because, as you know from the *Networking* module, a system administrator can configure a service to listen to any TCP or UDP port.

- To detect which service is listening on a port, you can use `nmap` or other fingerprinting tools.

4.2.3.3 Detecting Services

- By knowing the services running on a machine, a penetration tester can infer:
  - The **operating system**.
  - The **purpose** of a particular IP address; for example, if it is a server or a client.
  - The **importance** of the host in the client's business. For example, an e-commerce enterprise will heavily rely upon its website and its database servers.

4.2.3.3 Detecting Services

+ After a map of the network infrastructure and the services running on it is built, you can start the vulnerability assessment using a vulnerability scan and/or manual inspection.

4.2.4 Vulnerability Assessment

- The **vulnerability assessment** phase is aimed at building a **list of the vulnerabilities present** on the target systems.

- The penetration tester has to carry out a vulnerability assessment on **each target** found in the previous steps.

4.2.4 Vulnerability Assessment

- The next phase, exploitation, will go through this list to exploit the systems.
- The bigger the list, the more the chances to exploit the systems in scope.

4.2.4 Vulnerability Assessment

+ You can carry out a vulnerability assessment:
  - **Manually** by using data collected in the previous phases
  - By utilizing **automated tools**

+ Vulnerability assessment tools are scanners that send probes to the target systems to detect whether a host has some well-known vulnerabilities.

4.2.4 Vulnerability Assessment

- Once the vulnerability scan is complete, the scanner will deliver a report that the pentester can use in the exploitation phase.

- As automated scanners can perform a huge number of probes, it is **extremely** important to properly configure them leveraging the information collected in the previous steps.

4.2.4 Vulnerability Assessment

- Otherwise, the scanner will blindly perform all its probes, even the ones that do not apply to your targets; this would increase the chances of crashing services and would also take more time than necessary to complete.

4.2.4 Vulnerability Assessment

- Most of the time this phase is done by using both automated scanners and manual inspection.

- **Automated tools can help carry out a penetration test, but they will not perform a penetration test on their own.**

4.2.5 Exploitation

- At this point, it's time to verify if the vulnerabilities really exist. The **exploitation** phase takes care of exploiting all the vulnerabilities found during the previous step.

- During the exploitation phase a penetration tester **checks and validates a vulnerability** and also **widens** and **increases** the pentester's privileges on the target systems and networks.

4.2.5 Exploitation

- A successful exploit of a machine helps to investigate the target network further, to discover new targets and to **repeat the process** from the information gathering phase!
- A penetration test is indeed a **cyclic process**.

4.2.5 Exploitation

- The process ends when there are no more systems and services **in-scope** to exploit.
- Remember, a penetration test is used to find **any and all vulnerabilities.**

4.2.6 Reporting

- Lastly, the final **penetration test report** is as important as the whole testing phase, as it is your way to officially deliver and communicate the results of your tests with:
  - Executives
  - IT Staff
  - Development team

- The report shows and explains the result of your tests and is the actual deliverable of your professional engagement.

4.2.6.1 The Report

- The report must address:
  - Techniques used
  - Vulnerabilities found
  - Exploits used
  - Impact and risk analysis for each vulnerability
  - Remediation tips

- Targeted tips on how to effectively remediate each vulnerability are the **real value** for the client.

4.2.6.1 The Report

- Remember that the work of a penetration tester is much more appreciated if, other than his elite exploitations skills, it provides **useful suggestions and techniques** the client can use to resolve their security issues.

4.2.6.2 Consultancy

- Penetration testers are often asked to provide some hours of consultancy after delivering the report; this is an additional service to the client should they need further clarification or help regarding your findings.

4.2.6.2 Consultancy

- After the consultancy step, the engagement is closed and the penetration tester must keep the report **encrypted in a safe place,** or better yet, **destroy** it.

4.2.7 The Secret of an Effective Pentest

Why wouldn't an experienced penetration tester just skip to the exploitation phase? In the end, it's what they are paid for, isn't it?

+ Imagine the systems in scope as a **target**. The bigger the target, the more chances you have to hit it with your darts.

+ Stages like information gathering and fingerprinting do just that; they **make your target wider**!

4.2.7 The Secret of an Effective Pentest

- In technical jargon, this activity is called "**widening the attack surface**".

- Using your time at widening the attack surface is much more valuable than shooting darts at an unknown target. You do not know where to shoot, and you do not know which technique is the best to use.

4.2.7 The Secret of an Effective Pentest

- On the other hand, a targeted attack has many more chances to succeed! Your main goal as a pentester is to first increase your chances of success and then shoot your darts.

- **Sticking to the process you've just seen is the real secret for an effective pentest!**

4.2.7 The Secret of an Effective Pentest

- In fact, highly motivated and experienced hackers spend most of their time investigating their victims and gathering information about them using as many sources as possible; this helps them launch highly targeted attacks that do not trigger alarms in the victim's defense system.

4.2.7 The Secret of an Effective Pentest

- A successful and stealthy attack is made possible by a deep understanding of the target, which comes from a thorough information gathering phase.

4.2.7 The Secret of an Effective Pentest

- During the penetration testing part of this course, you will see the tools and techniques to carry out each and every step of the penetration testing process we have studied in this module: information gathering, scanning, vulnerability assessment, and exploitation!

References

- Nmap: http://nmap.org/