Aligning between business IT and IT security - Penetration Testing

Author: Cristian Cornea

Have you ever asked yourself, what is a Penetration Test, and how one is being done?

In this article, we will dive in-depth into different types of Penetration Tests, various approaches, its lifecycle, and responsibility distribution within such an assessment.

1. What is a Penetration Test

In common language, you can define it as the way of finding out how a company can be hacked. In a professional language, you can explain it as a simulated and authorized cyber-attack on an organization's assets, performed to identify any security weaknesses.

1.1. What is not a Penetration Test

A Pentest is neither a Vulnerability Assessment nor Compliance Audit. It is not something basic to do, that’s why it is priced very high in comparison with other professional services within the Cyber Security industry. If it was that simple, anybody would be doing it! The reasons for that affirmation is that the methodology for Penetration Testing is learnt through years of practice, and hard challenges. Also, the biggest requirement in such an assessment is creativity and the “try harder!” mentality, which is found within a small percentage of testers out there.

1.2. Differences between different security assessments

Vulnerability Assessment

- Exposes low-hanging fruits and probably some risky issues
- The quality of results depends on the scanners used
- Automated around %90, and manual work around %10
- Push some buttons, watch YouTube, and export them in a nice report

Penetration Testing

- Exposes issues ranging from low severity to critical ones
- The quality of results depends on the expertise of the pentesting team
- Automated around %10, and manual work around %90
- Banging your head against a wall because that reverse shell is not talking back

Compliance Audit

- Checking if practices imposed by a compliance body are followed
- Tests not only the technical controls, but also the administrative ones, for example the policies enforced within the organization.

2. Different Approaches of Conducting a Penetration Test

Let's take a look over how a Pentest can be performed within your organization.

Firstly, it is the **Internal-Internal** model, where the assessment is being run by one of your internal teams, such as the IT Security department, and they are targeting only assets within the internal network. On the other side, the **Internal-External** approach is basically the same, but the pentest is being done from an external attacker perspective, targeting the Internet exposed assets in a black-box way. Allocating internal resources for this kind of assessment can save you a lot of money, but you are not sure how much offensive skills your team have, and it can lead to the possibility of missing serious vulnerabilities. If someone is good in IT, it doesn't mean it is good in Cyber Security too, those are two very distinct areas of technology.

Then, we have the **External-External** and the **External-Internal** models, where the organization is contracting a 3rd party to conduct the Penetration Test. The auditor can be a freelancer or security agency. Before starting working with someone external, double-check them, by verifying their certifications with the vendors, and confirming their past work reference. Remember that they possibly will get in some places that you wish nobody will get there, so trust must be established between you and them. Besides the trust, ensure that you will make a mandatory NDA to be signed by any 3rd party that is going to pentest your organization. This approach is not cheap, but you can be sure that it will cover most of the vulnerabilities. Some compliances and standards require you to have an external assessment frequently.

From a legal standpoint it should be better to opt for internal resources for penetration testing activities, because for those you have a better control, in comparison with a 3rd party, for which you cannot control or fully monitor them regarding what information they will store on their side, or exactly what sensitive actions did they do.

However, from an operational and usefulness perspective, the external resources are preferred, given their independence when performing the assessment. In this case, certain measures can be set in place to limit risks such as the loss of confidentiality or integrity: legal aspects (e.g. clear scope definition in the penetration contract, clearly detailed steps for the penetration testing, including the moment the penetration tester stops his/her attack) and cyber security aspects (e.g. use of the organization's laptops with proper security tools in place, logging of the penetration testing activity).

2.1. Black-box, Grey-box, and White-box Pentesting

The first question an auditor will ask you regarding a future assessment will be about which type of Penetration Test you want. Let's look a bit at all types of Pentests:

Black-box

When you are giving the auditor only an IP address or URL, and tell them: Try hack it! It is one of the most expensive options. Why? Because you are not giving the attacker any insight about the target or your organization, so much time must be spent on the recon phase of the Penetration Test. But, it also reflects the possibilities of an unprivileged external threat actor to breach the system.

Grey-box

Besides the targets, you are also giving the credentials for a low-privileged user. This is one of the most common techniques to conduct a good Penetration Test.

White-box

The auditor has full transparency of the target, including accounts with high privileges, source code of the applications/software, network infrastructure, and so on. Many people say it is cost-less, but from a personal perspective of view, I think it is not like that, because the auditor will have to do source code analysis, infrastructure security review, threat modelling, and extra activities that cost money and that are not done through the Black-box testing.

2.2. Should I keep my prevention tools enabled during the Pentest?

It is a very common question asked by many IT administrators or business executives that want to engage in a Penetration Testing assessment. So, the answer is: It depends!

Do you expect a Pentest that will reflect as much as possible the reality and show you the real capabilities of the attackers? If yes, then you should keep your IDS/IPS/WAF running.

Or do you expect a Pentest that will reflect security issues and vulnerabilities in the standalone product/application? Those issues can be still exploited, but the attacker must bypass the detections in order to get to them.

3. Penetration Testing Lifecycle

i. **Planning Phase**: it is when the scope and the rules of the assessment are being defined. Documents such as Statement of Work (SoW), Rules of Engagement (RoE) are being created.

ii. **Information Gathering**: reconnaissance is being performed by the auditor through various techniques including Open Source Intelligence (OSINT) too. From a technical standpoint, it is scanning the assets for open ports, protocols, subdomains, web directories, hidden files, and so on. It is one of the most time-consuming phases within the lifecycle of a Penetration Test.

iii. **Vulnerability Discovery Phase**: it is the stage where the auditor is poking your systems, network, or applications in order to discover vulnerabilities. It is mostly manual work, because tools are not covering each attack vector possible, and they are not as intelligent as humans in order to find logic security issues or complex attacks.

iv. **Vulnerabilities Exploitation**: the findings from the previous step are actively exploited.

v. **Post-Exploitation**: simulating the capabilities of an attacker after he/she gained initial access. Techniques like Privilege Escalation and Lateral Movement are being used within this phase of the Penetration Test. The goal is to gain the highest permissions possible.

vi. **Reporting**: the tester will present a document that contains an executive summary of the assessment, and the vulnerabilities that were found, including references, steps to reproduce, screenshots, and remediation suggestions for those. Also, a good report will contain some threat models too, that will be used to describe how the issues can be linked together in an exploitation chain.

vii. **Re-Testing**: you have fixed all the vulnerabilities (or at least the most severe ones), and now you have to request the retest from the auditor, which means that he/she will come over and re-verify if the vulnerabilities are still there or not.

4. Responsibility Distribution in a Pentest Assessment

A Penetration Test is not a 5-minutes task to accomplish, on the contrary it is a complex process that requires engagement from a vast majority of people within your organization, from IT department up to the management or executive levels. In some moments we are going to discuss how the responsibility is distributed when a pentest is planned to be conducted.

4.1. Defining Rules of Engagement

RoE, or "Rules of Engagement" is a document that specifies the scope of the assessment, included/excluded tests to be made, favorable time interval for testing, auditors and their source IP addresses.

In order to establish this document, IT, Cyber Security, and Management must collaborate and define the terms.

IT and Cyber Security departments must review from a technical point of view, and the following questions must be addressed: What type of Pentest do we want? Which assets should we target and why? Do we enable the detection and prevention tools during the assessment? Are there any specific tests that we want to exclude, such as Denial-of-Service (DoS)?

Management department should look after this from a financial or business standpoint: Do we really need to test xyz? What's the best time interval for tests to be carried without disrupting the business activity or affecting its customers? What type of approach provides the best Return on Investment (ROI)? For example, a good strategy that has been seen within small-medium organizations would be to choose Penetration Testing for external-facing assets, and Vulnerability Assessment for internal infrastructure.

4.2. Preparing the Environment

If we are talking about a website, software, or application that is in-scope, then the testing environment must be a 1:1 copy of production. At the application-layer, the software development department is responsible for providing that, but at the network-layer, the IT must set up the infra stack.

If we are talking about an infrastructure, whenever that would be an internal network, or cloud deployment, then the IT is responsible for providing access to the environment, because that cannot be that easily replicable, so the auditors will do the tests on the real assets. Be very careful in that case, which tests you should exclude from the assessment, because it can lead to negative impact.

Cyber Security department should be involved only if the decision of removing security tools is being approved, or if the scope of the assessment is also to test the detection capabilities.

The risks when setting up an unprotected testing environment is that you will give threat actors a larger attack surface, unless you will not expose it to the Internet, and only keep it internally behind a VPN.

4.3. Mitigating the Vulnerabilities

That's very simple to decide, based on the issue type. If it is a platform or software vulnerability, then the development is being responsible. If it is a network or endpoint based security issue, then the IT department should act on it.

Before acting against the findings, be sure to create a mitigation plan, and present it to the management for approval. Keep in mind that fixing vulnerabilities means effort, which is represented in costs.

How can we prioritize which vulnerabilities must be mitigated first? We can do that based on multiple criteria:

Severity - all the findings will be tagged from "Low" up to "Critical", and optionally numbered using CVSS ratings, which is a scoring mechanism for the impact of the vulnerabilities. It is recommended to prioritize the Critical and High ones first.

Estimated Effort to Fix - the second factor that must be taken in consideration when prioritizing the fixes for the security issues, is represented by the time needed to fix.

Impacted Asset Value - vulnerabilities affecting important servers/devices for the business functionality should be placed above those impacting low importance assets.

5. Takeaways

Penetration Testing is more like a journey, and not a one-way project, from scoping to mitigation and to re-testing. It is the journey that takes your organization's security to the next level. You should find a trusted external tester in order to get the best results.

As a bonus, we have attached also a guideline for doing Penetration Testing on Web Applications.

6. Guideline for Web Penetration Testing

Checks for Security Misconfigurations

- Unencrypted Communication (HTTP)
- SSL/TLS Misconfigurations
- Missing/Misconfigured Security Headers
- Missing Security Flags on Cookies
- Missing Rate-Limiting
- OPTIONS/TRACE Methods Allowed
- No custom pages defined for error pages
- Directory Listing
- Clickjacking

Checks for Information Disclosure

- through Error Pages
- through Response Headers
- through comments
- through StackTrace/Debug messages
- through direct request

- through other HTTP Methods
- through files

Checks for Vulnerable Components

- Vulnerable Libraries/Server/Proxy/Frameworks
- Vulnerable/Misconfigured WAF

Checks for Username/Email Enumeration

- through Login Error Message Discrepancy
- through Forgot/Reset Password Functionality
- through Registration Form
- through Response Time Discrepancy
- through Response Size Discrepancy
- through Account Lockout Message

Checks for Session Management Issues

- Missing Sessions Invalidation after Password Reset
- Missing Sessions Invalidation after Account Disable
- Missing Sessions Invalidation after Account Changes
- Session Fixation
- Logout doesn't Expire Token
- Concurrent Sessions
- Predictable Session Cookie Value/Token
- Missing Idle Timeout
- Missing Session Expiration after x time

Checks for Authentication & Authorization Issues

- Bypass Authentication
- Missing/Broken Multi-Factor Authentication
- Missing Authentication on Pages/Files/Resources
- Brute-Force/Dictionary Attacks
- Weak/Predictable Passwords
- Broken Password Reset Functionality
- Broken Access Control through Direct Request
- Broken Access Control through UI Manipulation
- Insecure Direct Object Reference (IDOR)
- Privilege Escalation
- Account Takeover

Checks for File Upload Issues

- Malicious File Upload
- Bypass Extension Check
- Bypass Content-Type Check
- EXIF Metadata not Removed from Images
- Missing File Size Check
- Overwrite Web Server File
- Path Traversal

Checks for Common Vulnerabilities

- Reflected Cross-Site Scripting (XSS)
- Stored Cross-Site Scripting (XSS)
- DOM-Based Cross-Site Scripting (XSS)
- Self Cross-Site Scripting (XSS)

- Blind SQL Injection
- Error-based SQL Injection
- Stacked SQL Injection
- Time-based SQL Injection
- XML External Entities (XXE)
- CSV/Formula Injection
- CSS Injection
- HTML Injection
- Template Injection
- Cross-Site Request Forgery (CSRF)
- Blind Remote Code Execution (RCE)
- Remote Code Execution (RCE)
- Deserialization
- Application-Layer DoS
- Open Redirect
- CRLF Injection
- GraphQL Abuse
- RFI/LFI
- SSRF
- Host Header Injection
- Arbitrary File Read/Write/Download
- XSLT Injection
- CORS