Cheatsheet Pentesting

Boite à idées pour l’évolution du document

Chers contributeurs,

Voici la section regroupant les idées d’évolutions du document. Idéalement, il sera la synthèse des accords qui auront été débattus et décidés par un minimum de parties prenantes. Veuillez vous référer à cette section avant toutes modifications.

Suggestion n°1 :

Prévoir un tableau généraliste d’outils en 5 colonnes contenant le nom de l’outil, l’étape du pentesting correspondant, le type de test dont il s’agit, le framework lié et un lien.

Par exemple, le point 6 serait à restructurer dans le tableau.

Nom	Etape pentesting	Type de test	Framework lié	Lien
Google dorking	Pre-enga… ▾	…	…	…

Peut-être le mettre en paysage pour plus de clarté. Encore une fois, c’est une suggestion. Peut-être même inclure des filtres (migration du document vers un autre format ?) pour ne pas encombrer le document et obtenir l’information souhaitée.

1. Pentesting Steps

1. Pre-engagement : define the scope of the test and the methods that will be used for it.

2. Reconnaissance : Collecting as much publically accessible information about the target as possible.

3. Enumeration/Scanning : Discover the applications and services running on the systems, scan the ports (nmap).

4. Exploitation : Exploit vulnerabilities discovered on a system or application either via public exploits or via exploitation of the application/system logic.

5. Privilege Escalation : extend your access to the system.

6. Post-exploitation : This phase is broken down into several sub-steps:

7. Full privilege collection : Additional information with a high-privileged user.

8. Pivoting : Check other potential targets.

9. Clean-up : Erase the traces of your intrusion.

10. Reporting : Report harvesters, the flaws that have been discovered, the methods used, the paths, the tools, the commands, or any other relevant information that can be used in the last phase.

11. Remediation : Fixing and proposing solutions for all vulnerabilities that were identified during the reporting phase.

2. Different types of testing

1. Web Application Penetration Testing

Discover vulnerabilities or security gaps in web-based applications, identify security weaknesses or vulnerabilities in applications and components like databases, source code, and backend networks and provide practical solutions for remediation.

The penetration tester uses at least three steps to evaluate your web application:

- **Reconnaissance**—gathering information about the operating system, services, and resources being used.
- **Discovery—**attempting to find vulnerabilities.
- **Exploit**—using the vulnerabilities to gain unauthorized access to sensitive data or systems.

2. Network Penetration Testing

Testing every network system for found vulnerabilities (workstation, switchers, routers, printers, …)

Protect an organization from common network-based attacks such as bad configuration router/switching attacks, evasion of IPS or IDS systems, DNS attacks, SSH attacks, proxy attacks, database attacks, man-in-the-middle attacks and FTP/SMTP attacks.

application	use	source
Metagoofil (Linux/Windows)	extracting metadata of public documents (.pdf, .doc, .xls, .ppt, .odp, .ods) list potential usernames for brute force attacks. also extracts paths and MAC address information from the metadata.	
Exif Reader (Windows)	Image file analysis for Windows	http://www.takenet.or.jp/~ryuuji/minisoft/exifread/english
Gobuster	finding hidden pages on websides	

3. Wireless Penetration Testing

Identifies risks and vulnerabilities related to wireless networks.(configuration, sesson, wireless devices)

4. Physical Penetration Testing

Analyzes physical security violations (locks and physical access mechanisms, security cameras and security guards) and implements resolution measures.

AT Commands	Commands over an Android device's USB port to rewrite device firmware, bypass security mechanisms, exfiltrate sensitive information, perform screen unlocks, and inject touch events.	https://atcommands.org/
USB Rubber Ducky	Customizable keystroke injection attack platform masquerading as a USB thumbdrive.	http://usbrubberducky.com/
LAN Turtle	Covert "USB Ethernet Adapter" that provides remote access, network intelligence gathering, and MITM capabilities when installed in a local network.	https://hak5.org/products/lan-turtle
Proxmark3	RFID/NFC cloning, replay, and spoofing toolkit often used for analyzing and attacking proximity cards/readers, wireless keys/keyfobs, and more.	https://www.proxmark3.org/

5. Social Engineering Penetration Testing

Obtain sensitive information such as login credentials by deception, measure the level of vigilance of employees, test filtering mechanisms such as anti-spam.

application	use	source

Cree.py	Geolocation of a target and automation of data collection	https://www.geocreepy.com/
List of social networks		https://sites.google.com/site/listaredessociales/listaredessociales
Maltego	Automate the task of collecting data open source intelligence and forensics application	http://www.paterva.com/ good tutorial here : https://www.youtube.com/watch?v=qiv4-wy3mxo
TheHarvester	To gather email accounts and subdomain names from different public sources (search engines, pgp key servers)	
NetGlub	Data mining and information-gathering tool that presents the information gathered in a format that is easily understood.	
recon-ng	A full-featured Web Reconnaissance framework written in Python	https://github.com/lanmaster53/recon-ng
Datasploit	Tries to find out credentials,api-keys, tokens, subdomains, domain history, legacy portals, etc. related to the target. Generates HTML, JSON reports along with text files.	https://github.com/DataSploit/datasploit
check user name		https://checkusernames.com/
ixquick	Metadata leakage	http://ixquick.com/
MetaCrawler -	Metadata leakage	http://metacrawler.com
dogpile	Metadata leakage	http://www.dogpile.com
Search.com	Metadata leakage	Search.com
Visionneuse Exif de Jeffery		http://regex.info/exif.cgi
Shodan	Check after not securised	

	devices	
FOCA Windows	Reads metadata from a wide range of document and media formats, also relevant usernames, paths, software versions, printer details, and email addresses. This can all be performed without the need to individually download files.	
Image Search	Found picture, Tineye can be used to find other profiles on the Internet for more information about a person	http://www.tineye.com

6. Client Side Penetration Testing

Looks for security vulnerabilities in any type of software that can be exploited on client computers, such as employee workstations. Examples include web browsers like Google Chrome, Firefox or Safari; content creation software packages like MadCapFlare or FrameMaker, media players, etc.

You can perform client-side testing to prevent attacks like:

- Cross-origin resource sharing (CORS)
- Client-side malware infection
- Cross - Site Scripting
- Form hijacking
- HTML injection
- Malicious redirection

7. IoT Penetration Testing

reveal security vulnerabilities across the entire IoT ecosystem—hardware, embedded software, communication protocols, servers, web and mobile applications.

Hardware, firmware, and communication protocol testing should be appropriate to the device being tested. For example, testers can attempt to breach device authentication, or perform data dumps through firmware vulnerabilities or signal capture.

8. Mobile Application Penetration Testing

Tests run on mobile applications, excluding mobile APIs and servers. This typically involves two types of tests:

- **Static analysis**—extracting elements (both metadata and source code) and using them to perform reverse engineering on the application.
- **Dynamic analysis**—involves finding vulnerabilities while the application is running on the device. For example, testers may attempt to bypass controls or extract data from RAM.

The OWASP Mobile AppSec Verification Standard

defines a mobile app security model and lists generic security requirements for mobile apps. It can be used by architects, developers, testers, security professionals, and consumers to define and understand the qualities of a secure mobile app. The MSTG maps to the same basic set of security requirements offered by the MASVS and depending on the context they can be used individually or combined to achieve different objectives.

Navigating the Mobile Security Testing Guide (MSTG)

The General Testing Guide : mobile app security testing methodology and general vulnerability analysis techniques as they apply to mobile app security.

The Android Testing Guide : mobile security testing for Android

The iOS Testing Guide : mobile security testing for the iOS platform.

https://mobile-security.gitbook.io/mobile-security-testing-guide/overview/0x03-overview

3. Framework and utility

a. OSSTMM

The “*Open Source Security Testing Methodology Manual*” provides a detailed framework of testing strategies for **systems**, **software**, **applications**, **communications** and the **human aspect** of cybersecurity.

The **methodology** focuses primarily on **how these systems, applications communicate**, so it includes a methodology for:

- Telecommunications (phones, VoIP, etc.)
- Wired Networks
- Wireless communications

b. OWASP

The "*Open Web Application Security Project*" framework is a community-driven and frequently updated framework used solely to test the security of **web applications and services**.

The foundation regularly writes reports stating the **top ten security vulnerabilities a web application may have**, the testing approach, and remediation.

c. NIST

The “*National Institute of Standards and Technology*” Cybersecurity Framework is a popular framework used to **improve an organisations cybersecurity standards and manage the risk of cyber threats**. This framework is a bit of an honourable mention because of its popularity and detail.

The framework provides guidelines on security controls & benchmarks for success for organisations from critical infrastructure (power plants, etc.) all through to commercial. There is a limited section on a standard guideline for the methodology a penetration tester should take.

d. PTES

The “*Penetration Testing Execution Standard*” is a standard that was developed and continues to be enhanced by a group of information security experts from various industries. PTES provides a **minimum baseline for what is required of a penetration test**, expanding from initial communication between client and tester to what a report includes.

The goal of PTES is to provide quality guidance that **helps raise the bar of quality for penetration testing**. The standardization of penetration testing procedures **helps organizations better understand the services they are paying** for and gives penetration testers accurate direction on what to do during a penetration test.

e. Penetration Testing Framework 0.59

To complete

f. ISSAF

It's a methodology supported by *Open Information Systems Security Group* (OISSG) Although it is no longer maintained and, therefore, a bit out of date, one of its strengths is that it links individual pentest steps with pentesting tools.

For specific use in certain companies, this protocol makes it possible to meticulously plan and document each step of the test procedure from development to resolution of the incident while combining it with other tools.

The assessment section details a large part of the process including, at each step, information on attack vectors, the impact of an exploit and in some cases, data on the tools commonly used to target vulnerable areas. This information helps to anticipate complex attack patterns and to anticipate the consequences.

4. Rules of engagement and law frames

a. Project Description Overview

Contractual statement with the client that defines the objectives.

I. Purpose

The objective of the test is to identify risks and vulnerabilities, evaluate the effectiveness of the security configuration parameters and propose solutions with a penetration test report in order to help the customer on the improvement and investment measures to take.

II. Scope of testing

Provide a list of upcoming actions in terms of type of intrusion (external network, internal...)

b. Project Schedule of Activities

Define test deadlines (start, end, duration)

c. Project Logistics

First, make a list of all the people who will be involved in the test. The person in charge of the test and their managers, as well as the clients' contacts.

5. Useful resources

UseFull Github, with lot of ressources for pentesting (Android, Python, Kali, ….)
https://github.com/tanc7/hacking-books

Another usefull github : https://github.com/enaqx/awesome-pentest

Z-Library : find lot of things about pentesting : https://fr.b-ok.xyz/

Cyber-chef : convert crypted datas on stings : https://cyberchef.org/

6. find publics informations

Google Dorks, Shodan, Maltego, Social networks, Spiderfoot, setools kit (kali),

6.1 Google commands for find informations

“things”, interpret between these quotation marks as exact and only return the results of the exact phrase provided.

filetype: indicate which file extension used (ex filetype:pdf)

cache: indicate masked google version of url
intille,: the specified word appear the firstpage.

site : indicate the website when google need to search.

official Google dork helpful page : https://www.google.com/advanced_search

Exemple: intitle:.mp4 site:youtube.com
Find vidéos when the name contains   .mp4 on youtube

6.2 Osint (Open-source intelligence)

lot of tools for obtains data all over the internet (images, vidéos, ….)

OsintFramework or use this link : https://github.com/lockfale/osint-framework

Amazing osint tool list : https://start.me/p/rx6Qj8/nixintel-s-osint-resource-list
Explains the most current osint tools vidéo here
Maltego
Recon-NG

7.0 Enumeration

Use tools like Nmap, TheHarvester, Gobuster, Dnsrecon, …. for finding lot of informations on servers.