

# **Creando una Trayectoria Profesional en Seguridad Digital:**

## **Análisis de Amenazas (sesión técnica)**

**Agosto, 2021**

# Tabla de contenidos

Análisis de Amenazas - Sesión Técnica

Identificación de puertos abiertos (escaneo o rastreo)

Seguridad en aplicaciones

Seguridad en microservicios

Seguridad en dispositivos y aplicaciones móviles

# Análisis de Amenazas - Sesión Técnica

En esta sesión se abordarán los siguientes temas:

- ▶ Identificación de puertos abiertos y reconocimiento de servicios y *banners*
- ▶ Reconocimiento de sistemas, plataformas y aplicaciones
- ▶ Explotación
- ▶ Ataques a contraseñas
- ▶ Seguridad en aplicaciones web
- ▶ Breve introducción al top 10 de OWASP
- ▶ Ataques *client-side*
- ▶ Breve introducción a la seguridad en dispositivos móviles
- ▶ Breve introducción a la seguridad en dispositivos inalámbricos (802.11)

# Identificación de puertos abiertos (escaneo o rastreo)

## (1/12)

- ▶ El escaneo o rastreo de puertos se emplea para **analizar**, por medio de un programa, el estado de los puertos (TCP/UDP) de una máquina conectada a una red de comunicaciones
- ▶ Su objetivo es **detectar** si un puerto está **abierto** o **cerrado**, para posteriormente identificar qué servicios son alcanzables en la máquina y las posibles vulnerabilidades de seguridad según la información obtenida
- ▶ También puede llegar a detectar información acerca del sistema operativo que se está ejecutando en la máquina, **según la información obtenida de los puertos disponibles**

# Identificación de puertos abiertos (escaneo o rastreo)

## (2/12)

### NMAP (Network Mapper)

- ▶ **NMAP** es una utilidad gratuita y de código abierto, desarrollada por Fyodor; está diseñada para el descubrimiento de sistemas, redes y la auditoría de seguridad
- ▶ Es multi-plataforma (Windows/Linux/Mac) y contiene interfaces de comandos o gráfica.

# Identificación de puertos abiertos (escaneo o rastreo)

## (3/12)

```
(kali㉿kali)-[~]  
$ nmap -F oea-labs.ddns.net  
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 22:48 CDT  
Nmap scan report for oea-labs.ddns.net (52.23.247.62)  
Host is up (0.10s latency).  
rDNS record for 52.23.247.62: ec2-52-23-247-62.compute-1.amazonaws.com  
Not shown: 96 filtered ports  
PORT      STATE SERVICE  
22/tcp    open  ssh  
25/tcp    open  smtp  
80/tcp    open  http  
8000/tcp  open  http-alt  
  
Nmap done: 1 IP address (1 host up) scanned in 4.59 seconds
```

Figura 1: nmap ejecutado con la opción -F para un rastreo rápido.

# Identificación de puertos abiertos (escaneo o rastreo)

## (4/12)

### Capacidades de *nmap*

Entre sus capacidades, *nmap* permite escanear redes, bloques de direcciones IP, hacer barridos de *ping*, identificar equipos activos, rastrear puertos abiertos (protocolos TCP/UDP), banners, servicios, sistemas operativos, además de una serie de scripts para ejecutar tareas más avanzadas de penetración.

### ¿Qué es un banner o banner grab?

Es una técnica de reconocimiento que consiste enviar solicitudes a los servicios, para poder obtener respuestas **que permitan aprender acerca de las versiones del servicio o detección del sistema operativo**

# Identificación de puertos abiertos (escaneo o rastreo)

## (5/12)

```
(kali@kali)-[~]
└─$ nmap -sV -F oea-labs.ddns.net
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 22:54 CDT
Nmap scan report for oea-labs.ddns.net (52.23.247.62)
Host is up (0.063s latency).
rDNS record for 52.23.247.62: ec2-52-23-247-62.compute-1.amazonaws.com
Not shown: 96 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
25/tcp    open  smtp?
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
8000/tcp   open  http     SimpleHTTPServer 0.6 (Python 3.5.2)
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-s
service :
SF-Port25-TCP:V=7.91%I=7%D=4/18%Time=607CFF05%P=x86_64-pc-linux-gnu%r(NULL
SF:1C,"421\x20please\x20try\x20again\x20later\r\n");
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.99 seconds
```

Figura 2: nmap ejecutado con las opciones -F para un rastreo rápido y -sV para la detección de versiones.

- Las herramientas de escaneo de puertos normalmente determinan si un equipo está activo antes de iniciar un scan o rastreo



# Identificación de puertos abiertos (escaneo o rastreo)

## (6/12)

- Usualmente inician a través de un **barrido de ping**

### ¿Qué es un ping?

Ping es una herramienta basada en el protocolo *ICMP* (*Internet Control Message Protocol*), que envía una petición a un servidor o host, que por lo general\* es contestada si el sistema está conectado.

NMAP envía un ICMP Echo-Request (Ping) y un paquete al puerto 80 para **determinar si un sistema está activo**, antes de escanear.

\*Un firewall puede bloquear tráfico ICMP

# Identificación de puertos abiertos (escaneo o rastreo)

## (7/12)

### ¿Qué es un firewall?

Es un dispositivo/aplicación que analiza/monitorea **tráfico entrante y saliente**. Permite o bloquea los *paquetes* basado en reglas de seguridad.

# Identificación de puertos abiertos (escaneo o rastreo)

## (8/12)

### Escaneo de Puertos TCP y UDP

- ▶ TCP y UDP tienen puertos (un campo en el cabecera TCP/UDP)
- ▶ Existen  $2^{16} = 65,536$  puertos TCP/UDP
- ▶ Existe una asignación de puertos por la IANA<sup>†</sup>:
  - ▶ Puerto TCP **80**, servidores web ( protocolo HTTP)
  - ▶ Puerto TCP **23**, Telnet
  - ▶ Puerto UDP/TCP **53**, DNS

<sup>†</sup> *Internet Assigned Numbers Authority*

# Identificación de puertos abiertos (escaneo o rastreo)

(9/12)

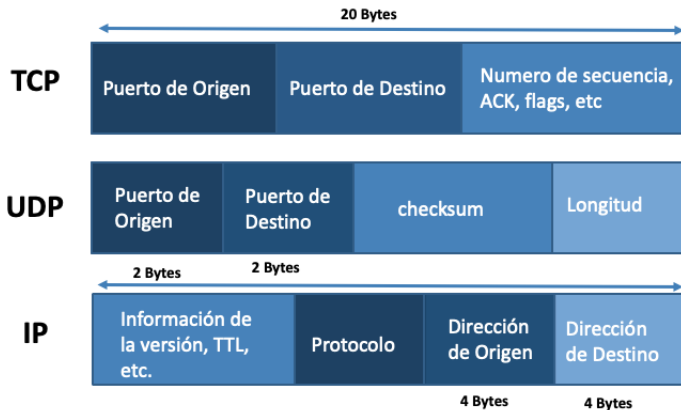


Figura 3: Información de los datagramas de protocolos *TCP*, *UDP* e *IP*.

# Identificación de puertos abiertos (escaneo o rastreo)

## (10/12)

```
(kali@kali)-[~]
$ nmap -sV -A -p 80,8000,5001 oea-labs.ddns.net
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 22:53 CDT
Nmap scan report for oea-labs.ddns.net (52.23.247.62)
Host is up (0.089s latency).
rDNS record for 52.23.247.62: ec2-52-23-247-62.compute-1.amazonaws.com

PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.18 ((Ubuntu))
|_ http-server-header: Apache/2.4.18 (Ubuntu)
|_ http-title: Apache2 Ubuntu Default Page: It works
5001/tcp   open  http      Werkzeug httpd 1.0.1 (Python 3.5.2)
|_ http-server-header: Werkzeug/1.0.1 Python/3.5.2
|_ http-title: Site doesn't have a title (text/html; charset=utf-8).
8000/tcp   open  http      SimpleHTTPServer 0.6 (Python 3.5.2)
|_ http-server-header: SimpleHTTP/0.6 Python/3.5.2
|_ http-title: Site doesn't have a title (text/html).

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.56 seconds
```

Figura 4: nmap ejecutado con las opciones -F para un rastreo rápido, -sV para la detección de versiones, -A modo agresivo (detección del sistema operativo, scanning y traceroute) y -p para la detección de puertos personalizados.

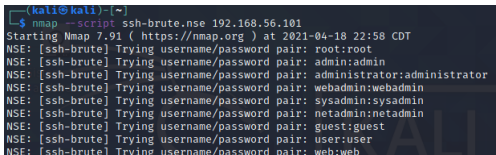
# Identificación de puertos abiertos (escaneo o rastreo)

## (11/12)

NMAP cuenta con scripts para enumerar información de protocolos específicos, con la opción `--script scriptname.nse`

Los scripts se pueden localizar con los comandos `find` o `locate` y el patrón `.nse`:

```
$ locate .nse
```



```
(kali@kali)~$  
$ nmap --script ssh-brute.nse 192.168.56.101  
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 22:58 CDT  
NSE: [ssh-brute] Trying username/password pair: root:root  
NSE: [ssh-brute] Trying username/password pair: admin:admin  
NSE: [ssh-brute] Trying username/password pair: administrator:administrator  
NSE: [ssh-brute] Trying username/password pair: webadmin:webadmin  
NSE: [ssh-brute] Trying username/password pair: sysadmin:sysadmin  
NSE: [ssh-brute] Trying username/password pair: netadmin:netadmin  
NSE: [ssh-brute] Trying username/password pair: guest:guest  
NSE: [ssh-brute] Trying username/password pair: user:user  
NSE: [ssh-brute] Trying username/password pair: web:web
```

Figura 5: `nmap` ejecutado con la opción `--script ss-brute.nse` para ejecutar un ataque de fuerza bruta a las credenciales del protocolo ssh.

# Identificación de puertos abiertos (escaneo o rastreo)

## (12/12)

```
(kali@kali)-[~]
$ nmap -sV --script vulners oea-labs.ddns.net 255 x
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 23:15 CDT
Nmap scan report for oea-labs.ddns.net (52.23.247.62)
Host is up (0.066s latency).
rDNS record for 52.23.247.62: ec2-52-23-247-62.compute-1.amazonaws.com
Not shown: 995 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
vulners:
cpe:/a:openbsd:openssh:7.2p2:
PACKETSTORM:140070 7.8 https://vulners.com/packetstorm/PACKETSTORM:140070*
EXPLOIT*
EXPLOITPACK:5BCA798C6BA71FAE29334297EC0B6A09 7.8 https://vulners.com/exploit
pack/EXPLOITPACK:5BCA798C6BA71FAE29334297EC0B6A09 *EXPLOIT*
EDB-ID:40888 7.8 https://vulners.com/exploitdb/EDB-ID:40888 *EXPLOIT*
CVE-2016-8858 7.8 https://vulners.com/cve/CVE-2016-8858
CVE-2016-6515 7.8 https://vulners.com/cve/CVE-2016-6515
1337DAY-ID-26494 7.8 https://vulners.com/zdt/1337DAY-ID-26494 *EX
PLOIT*
```

Figura 6: Con algunos scripts adicionales como [nmap\\_vulners](#) , se pueden ampliar las capacidades de **nmap**, para buscar posibles vulnerabilidades en el objetivo.

# Explotación de servicios (1/8)

- ▶ La explotación es una fase en la que un **atacante logra ejecutar una actividad no autorizada** en el entorno objetivo
- ▶ Existen millones de configuraciones inseguras, cientos de miles de exploits, miles de payloads (cargas útiles) e innumerables métodos para comprometer la seguridad de un sistema (CIA), a través de una debilidad técnica o vulnerabilidad
- ▶ Vectores como la inyección de caracteres, fallas de autenticación, exposición de información, contraseñas triviales y servicios expuestos por defecto, son comúnmente utilizados para explotar vulnerabilidades
- ▶ **No existe un estándar para crear exploits.**
  - ▶ El framework de explotación más popular de la industria es [Metasploit](#), creado por H.D. Moore, originalmente lanzado en 2003



## Explotación de servicios (2/8)

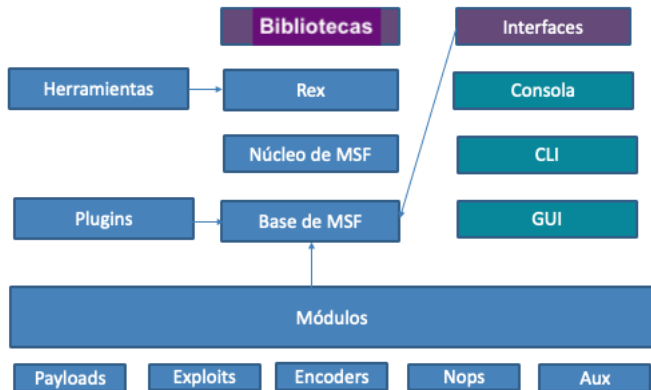


Figura 7: Arquitectura de Metasploit.

# Explotación de servicios (3/8)

## Acerca de Metasploit

Metasploit puede ser utilizado por línea de comandos o interfaz web.

Pasos para la **explotación**:

1. Identifique un servicio vulnerable
2. Seleccione el exploit
3. Algunos exploits validan si el servicio es vulnerable
4. Seleccione el objetivo (TARGET)
5. Seleccione la carga (PAYLOAD)
6. Si el exploit no tiene PAYLOAD es posible que se pueda indicar el comando a ejecutar
7. Ajuste las opciones y **¡Explotar el objetivo!**

# Explotación de servicios (4/8)

```
(kali㉿kali)-[~]  
└─$ msfconsole  
  
#####  
;."  
;id; .---,..  
" 00000'.'00 00000'.'0000 "  
'.0000000000000 000000000000 0;  
'.0000000000000 0000000000000 '  
"--'.000 -'.0 0 '-.'--"  
".0' ; 0 0 \.' ;'  
|0000 000 0 .  
' 000 00 00 ,  
'0000 00 ,  
' 00 0 ;  
( 3 C ) <|__ <Metasploit!>  
;0'. __*__.' <|__ <Metasploit!>  
'(,...."/  
  
=[ metasploit v6.0.30-dev ]  
+ -- ==[ 2099 exploits - 1129 auxiliary - 357 post ]  
+ -- ==[ 592 payloads - 45 encoders - 10 nops ]  
+ -- ==[ 7 evasion ]  
  
Metasploit tip: Display the Framework log using the  
log command, learn more with help log  
  
msf6 > █
```

Figura 8: Ejemplo de la consola de comandos de Metasploit.

# Explotación de servicios (5/8)

## Ataques a contraseñas

Las contraseñas de acceso a los sistemas deben ser protegidas contra:

- ▶ Divulgación/modificación/remoción **no autorizada**
- ▶ Deben almacenarse en repositorios seguros y cifrados

Existen servicios que están expuestos a ataques en línea:

- ▶ Adivinación de contraseña (*password guessing*)
- ▶ Ataques de diccionario (*root, toor, password, 1234, soporte, etc.*)
- ▶ Ataques de fuerza bruta (*AA, AB, AC, AD, ...*)
- ▶ Ataques Híbridos (*password123, abc123, ...*)

# Explotación de servicios (6/8)

```
(kali㉿kali)-[~]  
└─$ hydra -l ubuntu -P /usr/share/wordlists/rockyou.txt.gz 192.168.56.101 -t 4 ssh 255 x  
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret  
service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and et  
hics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-04-18 23:21:00  
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100  
tries per task  
[DATA] attacking ssh://192.168.56.101:22/  
[22][ssh] host: 192.168.56.101 login: ubuntu password: 123456  
1 of 1 target successfully completed, 1 valid password found  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-04-18 23:21:04
```

Figura 9: Ejemplo de un ataque de fuerza bruta mediante la herramienta **hydra**.

# Explotación de servicios (7/8)



The screenshot shows the incibe-cert website header with navigation links: Alerta, Incidentes, and Servicio. Below the header, a breadcrumb trail reads: Inicio / Alerta Temprana / Bitácora Ciberseguridad / Facebook guarda accidentalmente contraseñas en texto plano. The main headline is "Facebook guarda accidentalmente contraseñas en texto plano" dated 21/03/2019. The article text states that Facebook has informed that, by error, it has been storing user passwords in plain text, and although it has not provided details of the flaw, it has confirmed that both Facebook and Instagram users have been affected. It also mentions that Facebook has indicated that passwords have never been exposed to the Internet, only employees have had access, and it will contact affected users. A "Referencias:" section lists five sources with dates from 21/03/2019 to 23/03/2019, including newsroom.fb.com, motherboard.vice.com, genbeta.com, welivesecurity.com, and unaaldia.hispasec.com. To the right of the references, there are three red links: "Keeping Passwords Secure", "Facebook Mistakenly Stored 'Hundreds of Millions' of User Pass...", and "Facebook almacenó las contraseñas de cientos de millones de s...", followed by two lines of red text: "Facebook expuso millones de contraseñas de usuarios a sus em..." and "Facebook ha estado guardando contraseñas de usuarios en text..."

incibe-cert

Alerta ▾ Incidentes ▾ Servicio ▾

Inicio / Alerta Temprana / Bitácora Ciberseguridad / Facebook guarda accidentalmente contraseñas en texto plano

## Facebook guarda accidentalmente contraseñas en texto plano

21/03/2019

Facebook ha informado que, por error, ha estado almacenando las contraseñas de sus usuarios en texto plano y, aunque no ha proporcionado detalles de este fallo, ha confirmado que se han visto afectados tanto usuarios de Facebook, como de Instagram.

En su comunicado también ha indicado que las contraseñas nunca han estado expuestas a Internet, que únicamente los empleados de Facebook han tenido acceso a ellas y que no han detectado usos ilícitos. Además, van a contactar con los usuarios afectados para informarles del problema.

**Referencias:**

- 21/03/2019 newsroom.fb.com
- 21/03/2019 motherboard.vice.com
- 21/03/2019 genbeta.com
- 22/03/2019 welivesecurity.com
- 23/03/2019 unaaldia.hispasec.com

Keeping Passwords Secure

Facebook Mistakenly Stored 'Hundreds of Millions' of User Pass...

Facebook almacenó las contraseñas de cientos de millones de s...

Facebook expuso millones de contraseñas de usuarios a sus em...

Facebook ha estado guardando contraseñas de usuarios en text...

Figura 10: Ejemplo de una configuración no segura para el almacenamiento de contraseñas.

# Explotación de servicios (8/8)

Existen configuraciones que están expuestas a **ataques fuera de línea**, como por ejemplo el *password cracking*, diseñado para obtener la versión en texto claro de una contraseña a partir de un *hash* o un *dato cifrado*. Las técnicas de ataque y herramientas disponibles dependen del servicio objetivo.

- ▶ En Linux: `etc/shadow` + `etc/passwd`
- ▶ En Windows: `C:\Windows\System32\config\SAM` + `SYSTEM`

## ¿Qué es un hash?

Es una función unidireccional utilizada para relacionar datos de *tamaño arbitrario* a *tamaño específico*. Los valores resultantes se denominan *códigos hash*, *digests* o simplemente *hashes*.

Uno de sus usos más comunes es **comprobar la integridad de los datos**.

# Seguridad en aplicaciones (1/8)

- ▶ Muchas aplicaciones web tienen un *front end* y un *backend* donde se **almacena/procesa información sensible**
- ▶ Un atacante puede interactuar con el *front end* para **manipular el comportamiento de la aplicación** y el manejo de la información
- ▶ Cada aplicación utiliza un lenguaje, secuencias de código, lógicas de negocio y funcionalidades particulares, pero existen **características y riesgos generales**



# Seguridad en aplicaciones (2/8)

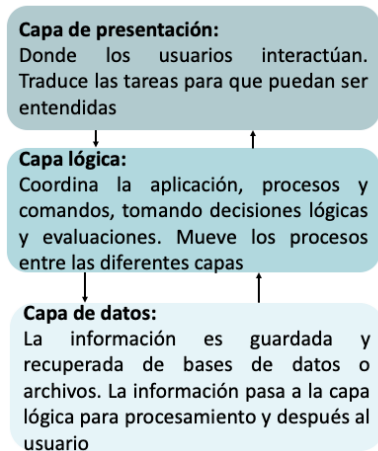


Figura 11: Capas empleadas en el diseño de aplicaciones.

# Seguridad en aplicaciones (3/8)

## OWASP (Open Web Application Security Project)

Es un proyecto dedicado a la **seguridad en aplicaciones web**. Se basa en un **top diez** donde se reportan de manera regular acciones relacionadas con la seguridad en aplicaciones, las cuales ocurren con más **prevalencia** o **criticidad**.

# Seguridad en aplicaciones (4/8)

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Figura 12: Vulnerabilidades y actualizaciones del framework de OWASP TOP 10.

# Seguridad en aplicaciones (5/8)

## ¿Qué es un proxy de HTTP?

- ▶ Un proxy HTTP es un programa que **intercepta las conexiones del cliente** y genera una nueva conexión al servidor
- ▶ Puede ser utilizado para filtrado de contenido y monitoreo
- ▶ Puede ser usado como herramienta ofensiva: (e.j. **BURP** o **OWASP ZAP** )

# Seguridad en aplicaciones (6/8)

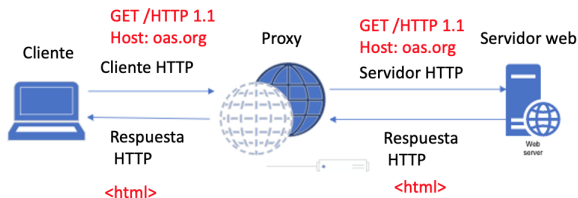


Figura 13: Diagrama de un proxy en el protocolo HTTP.

# Seguridad en aplicaciones (7/8)

## ¿Qué es un ataque del lado del cliente?

- ▶ El atacante puede intentar comprometer a los clientes, en lugar de acceder al servidor directamente
- ▶ El cliente afectado puede ser usado como **pivote**\* para acceder a la información del servidor
- ▶ Ejemplo de estos ataques puede ser el *spear-phishing* o el *spoofing*

\* *Técnica que utiliza un sistema comprometido para atacar otros activos en la misma red.*

# Seguridad en aplicaciones (8/8)



Figura 14: Diagrama de un ataque hacia el lado del cliente.

# Seguridad en microservicios

## ¿Qué es la seguridad en micro-servicios?

Es aquella utilizada para adoptar mecanismos a nivel granular, para poder aplicar técnicas de defensa en profundidad, para microservicios más seguros.

La OWASP provee una lista de verificación denominada [OWASP Serverless Top 10](#) para analizar diferentes vulnerabilidades de servicios serverless en la nube provistos por diferentes vendedores como *Amazon Web Services*, *Microsoft Azure* y *Google Cloud Platform*.



# Seguridad en dispositivos y aplicaciones móviles (1/9)

## ¿Qué es la seguridad en dispositivos y aplicaciones móviles?

Es una rama de la seguridad que se encarga de la protección de dispositivos móviles y aplicaciones de posibles ataques y del ambiente en dónde el dispositivo se encuentra conectado.

El [OWASP Mobile Top 10](#) ofrece una lista para definir los riesgos más comunes para dispositivos móviles.



Figura 15: Riesgos definidos en el OWASP Mobile Top 10.

## Seguridad en dispositivos y aplicaciones móviles (3/9)



Figura 16: WiFi Pineapple: dispositivo que actúa como un *hotspot honeypot*<sup>†</sup> para realizar ataques de *hombre en el medio*.

<sup>†</sup>Es un punto de acceso WiFi no legítimo que emula ser autorizado y seguro.

# Seguridad en dispositivos y aplicaciones móviles (4/9)

¿Qué es un ataque de hombre en el medio?

Es un tipo de ataque de *estilo espionaje*, que ocurre cuando un atacante se inserta como relay/proxy en un sesión de comunicaciones entre dos entidades.

# Seguridad en dispositivos y aplicaciones móviles (5/9)

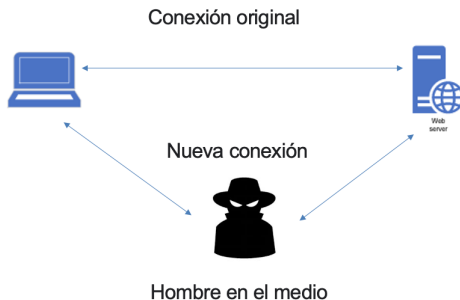


Figura 17: Diagrama de un ataque de hombre en el medio.

# Seguridad en dispositivos y aplicaciones móviles (6/9)

## ¿Qué es Wardriving?

- ▶ Es el proceso de descubrir redes inalámbricas
- ▶ Los datos de una red pueden ser utilizados para geo-localizar a un usuario
- ▶ A diferencia de las redes cableadas, los **datos viajan libremente** y las tramas pueden ser escuchadas por todos los usuarios en cobertura

# Seguridad en dispositivos y aplicaciones móviles (7/9)

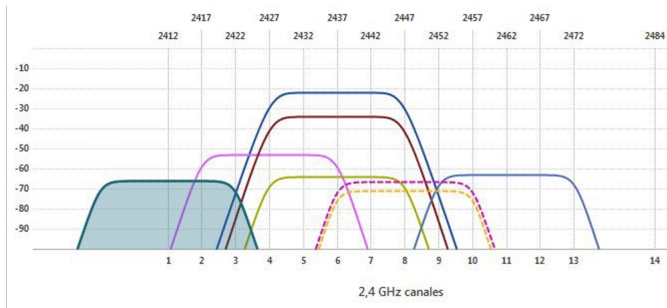


Figura 18: [NetSpot](#): herramienta para descubrir redes inalámbricas.

# Seguridad en dispositivos y aplicaciones móviles (8/9)

```
(kali㉿kali)-[~]  
$ aircrack-ng  
  
Aircrack-ng 1.6 - (C) 2006-2020 Thomas d'Otreppe  
https://www.aircrack-ng.org  
  
usage: aircrack-ng [options] <input file(s)>
```

Figura 19: **Aircrack** es una de las herramientas más utilizadas para romper contraseñas inalámbricas.

- ▶ El tipo de cifrado y los **mecanismos de autenticación** son los mayores retos para las redes móviles
  - ▶ WPA y WPA2 son ampliamente utilizadas
  - ▶ WEP es **vulnerable** y **obsoleto**
  - ▶ Se pueden agregar capas de **autenticación EAP (Extensible Authentication Protocol)** (*radius, NAC, doble factor de autenticación, autenticación de dos vías, etc.*)



# Seguridad en dispositivos y aplicaciones móviles (9/9)

*Las redes públicas son el escenario ideal para realizar múltiples ataques. Al convivir en el mismo segmento de red, se pueden hacer múltiples ataques a los clientes conectados.*

**El anonimato de estas redes es considerablemente amplio.**