UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS

VERITATE SOLA NOBIS IMPONETUR VIRILISTOGA. 1948

Facultad de Ingeniería Eléctrica

Departamento de Electrónica y Telecomunicaciones

TRABAJO DE DIPLOMA

"*Pruebas de penetración en entornos Wi-Fi*"

Autor: Pedro Enrique Iturria Rivera

Tutor: Ing. Carlos Miguel Bustillo Rodríguez

Co-tutor: Ing. Henry Moreno Díaz

Santa Clara

2015

"Año 57 de la Revolución"

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS

VERITATE SOLA NOBIS IMPONETUR VIRILISTOGA. 1948

Facultad de Ingeniería Eléctrica

Departamento de Electrónica y Telecomunicaciones

TRABAJO DE DIPLOMA

"*Pruebas de penetración en entornos Wi-Fi*"

Autor: Pedro Enrique Iturria Rivera

Email: piturria@uclv.edu.cu

Tutor: Ing. Carlos Miguel Bustillo Rodríguez

Especialista Principal en Seguridad Informática

DIC-UCLV

Email: cbustillo@uclv.edu.cu

Co-tutor: Ing. Henry Moreno Díaz

Profesor, Dpto. de Telecomunicaciones y Electrónica

Facultad de Ing. Eléctrica. UCLV

Email: henrym@uclv.edu.cu

Santa Clara

2015

"Año 57 de la Revolución"

Hago constar que el presente trabajo fue realizado en la Universidad Central "Marta Abreu" de las Villas como parte de la culminación de los estudios de la especialidad de Telecomunicaciones y Electrónica autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

_____

Firma del Autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

_____ _____

Firma del Tutor

Firma del Jefe de Departamento donde se defiende el trabajo

_____ _____

Firma del Co-tutor

Firma del Responsable de Información Científico- Técnica

PENSAMIENTO

"El sacrificio es parte de la vida. Es algo que debe asumirse. No es algo que se deba lamentar. Es algo a lo que debemos aspirar. Pequeños sacrificios. Grandes sacrificios."

Mitch Albom

DEDICATORIA

A mis padres, Mercy y Pedro.

AGRADECIMIENTOS

A mis padres por su apoyo y amor incondicional. Por ser ejemplos de persona y de profesional.

A mis queridos abuelos Mirta, Norma, Delio y "Papo" que en paz descanse.

A mi hermana Daylín, por estar siempre. A Victor por ser como un hermano para mí.

A mis sobrinos queridos José Alejandro y Jennifer por traernos a todos alegría.

A mis tíos y tías, Betty, Nancy, Jacqueline, Landy y Osmany por su apoyo en todo momento.

A todos mis primos.

A mis buenos amigos Arocha, Raidán, Adrián, Andrés, Felipe, Javier, Sierra, Omarito y Carlitos Abreu por estar en los buenos y malos momentos.

A mis amigos de lucha y estudio Daniel, Carlos "el Kaki" y Richard, por compartir tantas experiencias.

A Betty.

A mis tutores Carlos M. Bustillo y Henry Moreno, por su paciencia y ejemplo.

A todos mis profesores en la etapa de estudios universitarios, que compartieron sus saberes sin pedir nada a cambio. Por ser ejemplos de profesionalidad.

Al colectivo del Grupo de Redes que me acogieron como si fuese uno de ellos, en especial a Bustillo, Jorge Rafael, Adrián Collazo, Gretter, Yoel, Ernesto y Héctor Cruz.

A mis compañeros, por compartir casi 1000 días dentro un aula de clases.

A todas aquellas personas que me han brindado su apoyo, les agradezco profundamente.

Muchas Gracias

TAREA TÉCNICA

- Estudio de vulnerabilidades en escenarios reales de redes Wi-Fi.
- Estudio de lenguajes de programación como *Bash* y *Python* para la facilitación de las tareas de *pentesting* en redes Wi-Fi a través de *scripts*, en sistemas basados en GNU/*Linux*.
- Realización de pruebas de penetración en entornos reales de redes Wi-Fi, utilizando herramientas basadas en plataformas GNU/*Linux*.
- Evaluación del estado actual de la infraestructura de las redes Wi-Fi en el campus universitario, desde el punto de vista de seguridad.
- Plantear soluciones en la infraestructura inalámbrica para solventar los diversos ataques a las redes Wi-Fi.
- Confección de un manual para especialistas en pruebas de penetración y seguridad de redes.
- Elaboración del informe del trabajo de diploma.

Firma del Autor

Firma del Tutor

Firma del Co-tutor

RESUMEN

Las pruebas de penetración o *pentesting* en las redes de telecomunicaciones son de gran importancia si se desea garantizar la seguridad informática en las instituciones o empresas. La seguridad en redes inalámbricas constituye el talón de Aquiles de toda infraestructura de red por sus características específicas en el control de acceso y la transparencia del medio en que se propaga. En la presente investigación se realiza un estudio de los mecanismos de seguridad en las capas física y de enlace de las redes Wi-Fi, así como las principales vulnerabilidades y ataques existentes en estas. Se define además la metodología a seguir para realizar pruebas de penetración. Se realiza una caracterización de las distribuciones de seguridad informática activas, basadas en GNU/*Linux*, y se selecciona la distribución *Kali Linux* como la óptima para las pruebas de penetración, por su versatilidad y preferencia entre los especialistas de seguridad a nivel mundial. Se definen las potencialidades de las herramientas presentes en la distribución *Kali Linux*, y se demuestra el uso de herramientas personalizadas o creadas por el *pentester* para la facilitación de las pruebas. Por último se realizan una serie de ataques que responden a las vulnerabilidades generales y específicas en la UCLV, para concluir con consideraciones de implementación de una infraestructura inalámbrica segura, quedando de esta manera un manual destinado a los especialistas de seguridad informática de la Universidad Central “Marta Abreu” de las Villas u otros interesados.

TABLA DE CONTENIDOS

INTRODUCCIÓN

La seguridad informática abarca un amplio dominio, pero se puede definir como: "la protección contra todos los daños sufridos o causados por una herramienta informática y originados por el acto voluntario y de mala fe de un individuo" (Royer, 2004). En la actualidad este término posee gran relevancia en las redes de empresas e instituciones, debido a la inmensa cantidad de información disponible en internet. En las redes inalámbricas la seguridad informática toma un enfoque diferente debido a sus características específicas. En el caso de aquellas empresas o instituciones donde se desee implementar una red inalámbrica, con dispositivos que controlen los accesos de la red cableada, el impacto que se ocasiona en la infraestructura no es significativo. En temas de seguridad se debe tener en cuenta la gran cantidad de vulnerabilidades de estas redes, ya sea debido a una mala configuración de los elementos de red como por la elección de mecanismos de seguridad inadecuados para el entorno que se desea proteger. Las redes inalámbricas resultan altamente atractivas para los atacantes y usuarios malintencionados, quienes tratan por todos los medios de violentar los accesos y acceder a la información de los usuarios legítimos del sistema.

En Cuba se le concede un tratamiento especial a estos temas, a tono con las nuevas directivas relacionadas con la informatización y ciberseguridad. Con este fin gran parte de los profesionales se dedican al estudio de la seguridad en redes inalámbricas, específicamente en entornos Wi-Fi, y se garantiza la presencia de un grupo de especialistas en cada empresa, institución u organización.

En el ámbito de seguridad informática no se conciben sistemas impenetrables, pues siempre se detectará una vulnerabilidad de seguridad en la infraestructura tecnológica a la cual se desea penetrar. Lo fundamental es documentar a los usuarios acerca de cómo protegerse de las amenazas y cómo utilizar los recursos para prevenir ataques.

En múltiples investigaciones realizadas se considera el tema de la seguridad informática como una disciplina del conocimiento donde se busca cerrar la brecha de los eventos inesperados que puedan comprometer los activos de una organización y así contar con estrategias para avanzar ante cualquier eventualidad (Cano, 2004). De este modo surge el término de inseguridad informática, el cual se considera como una disciplina dual donde los académicos y practicantes de la industria buscan las maneras detalladas para que ocurran

eventos inesperados, establecen las condiciones extremas de funcionamiento de los dispositivos o estrategias, todo con el objetivo de hacer caminar en condiciones límite, la operación de la organización y sus negocios. Esta estrategia dual sugiere contextualizar en un escenario real la incertidumbre inherente de la seguridad informática. Considerar la inseguridad informática como parte del ejercicio de seguridad informática de las organizaciones, sugiere la capacidad de las organizaciones para cuestionarse sobre la situación real del balance entre seguridad, facilidad de uso y funcionalidad, no para lograr mayores niveles de confiabilidad y aseguramiento de sus arquitecturas, sino para evaluar el nivel de dificultad requerido por los atacantes para ingresar y vulnerar los medios de protección. Con un pensamiento de este nivel, las organizaciones no buscan solamente incrementar la confianza de sus clientes, sino comprender que la seguridad no es un problema de tecnología, sino un problema de riesgos y las diferentes maneras de comprenderlos y manejarlos (Cano, 2004).

Para realizar pruebas de penetración o más conocido como "*pentesting*" (del inglés *penetration testing*) (en el proyecto se utilizan indistintamente ambos términos) se emplea este concepto. Las empresas e instituciones hacen uso de especialistas en pruebas de penetración para desequilibrar la seguridad existente y así lograr la "invulnerabilidad". Estos especialistas se denominan "sombrero blanco" o "*White Hat*" en el mundo de la informática y se clasifican como "*hackers*" éticos pues se dedican a las pruebas de penetración y en otras metodologías de prueba para garantizar la seguridad de las organizaciones.

Las pruebas de penetración, implican la simulación de ataques reales para medir el riesgo asociado con brechas de seguridad potenciales. En una prueba de penetración (en oposición a un análisis de vulnerabilidad), los especialistas en *pentesting* no solo descubren las vulnerabilidades que pueden ser explotadas por los atacantes sino que también ejecutan ataques para vulnerar las fisuras encontradas, y donde fuese posible, evaluar a que recursos los atacantes podrían acceder después de una explotación exitosa (Weidman, 2014).

Para satisfacer la necesidad de seguridad existen varias distribuciones *Linux*, como *Kali Linux*, una de las más actuales que es una completa reconstrucción de *BackTrack Linux*, y se adhiere completamente a los estándares de desarrollo de *Debian*. Este posee más de 300 herramientas de pruebas de penetración, y es libre. Presenta un amplio soporte para

dispositivos inalámbricos, tiene un entorno de desarrollo seguro, paquetes y repositorios firmados con GPG, y se encuentra en varios lenguajes (Allen et al., 2014).

En la red de la Universidad Central "Marta Abreu" de las Villas (UCLV) no se utiliza la herramienta *Kali Linux* para llevar a cabo evaluaciones de seguridad de redes inalámbricas, lo cual contribuirá a fortalecer la seguridad en esta institución docente. Además, el presente trabajo proveerá a los profesionales del ámbito de la seguridad de redes inalámbricas de un material para la utilización de esta distribución que permita identificar y explotar las vulnerabilidades en entornos Wi-Fi. Teniendo en cuenta las razones expuestas anteriormente, se plantea el siguiente problema de investigación: ¿Cuáles deben ser los aspectos a tener en cuenta para un óptimo análisis de seguridad en entornos Wi-Fi mediante herramientas basadas en plataformas GNU/*Linux*, dirigidos a los especialistas de seguridad informática?

Para dar cumplimiento al problema de investigación, se propone el siguiente objetivo general:

Realizar pruebas de penetración en entornos reales Wi-Fi, que sirvan de base para la realización de un manual en plataformas GNU/*Linux* enfocada a los especialistas de seguridad informática cubanos.

Para resolver el problema de investigación y dar cumplimiento al objetivo general, se plantean los siguientes objetivos específicos:

- Identificar las vulnerabilidades en entornos Wi-Fi.
- Evaluar las herramientas utilizadas en la distribución de seguridad *Kali Linux* para realizar pruebas de penetración en las redes inalámbricas.
- Proponer medidas de seguridad para la mitigación de las vulnerabilidades.
- Evaluar el estado de seguridad inalámbrica existente en la UCLV.

A partir de cada objetivo específico se crean interrogantes científicas, a las cuales se les dan respuestas en el desarrollo de la investigación:

- ¿Cuáles son las vulnerabilidades en los entornos inalámbricos, específicamente en redes Wi-Fi?
- ¿Cuáles son las herramientas de análisis de seguridad inalámbricas basadas en plataformas GNU/Linux?

- ¿Qué medidas tomar ante la presencia de algún ataque a la infraestructura inalámbrica?

El proyecto se basa en herramientas de código abierto (del inglés *open source*) de seguridad informática, además se cuenta con las condiciones necesarias para su desarrollo, desde el punto de vista del *hardware* y *software*, así como el interés por parte de la Dirección de Informatización y Comunicaciones (DIC) de la UCLV.

Los resultados de la investigación poseen una aplicación práctica y teórica de gran trascendencia para los especialistas de seguridad informática y administradores de redes en el centro o cualquier institución que posea una red inalámbrica Wi-Fi de computadoras porque proveerá un material para la utilización de esta distribución que permita identificar y explotar las vulnerabilidades en las redes inalámbricas.

Organización del informe

Para satisfacer los objetivos establecidos, el trabajo se dividió en: introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas, glosario y anexos.

En el primer capítulo se realiza una introducción a la seguridad de las redes Wi-Fi, mediante la definición de conceptos y la caracterización de los mecanismos de seguridad de estas redes. Además se evalúan las principales vulnerabilidades de las redes Wi-Fi reportadas en la literatura.

En el segundo capítulo se describen las herramientas y comandos dentro del compendio de la distribución *Kali Linux* para la auditoría, explotación y prevención de vulnerabilidades en redes Wi-Fi.

Por último, en el tercer capítulo se propone un manual donde se describen los diversos ataques a las redes Wi-Fi, mediante la detección, monitoreo y explotación de vulnerabilidades; así como soluciones para mitigar estos ataques.

CAPÍTULO I. Introducción a la seguridad de las redes Wi-Fi

Las empresas, instituciones u organizaciones utilizan las redes LAN inalámbricas desde principios de la década del noventa del siglo pasado, aunque en sus comienzos, el mercado era bastante pequeño y las tecnologías de propiedad. A finales de 1990 y principios de 2000, se sentaron las bases para la adopción masiva de las redes LAN inalámbricas. El punto de partida lo constituye la publicación de la norma ANSI/IEEE 802.11 donde se establecen las directrices para desarrollar productos interoperables a menor costo. Las redes Wi-Fi presentan notables ventajas como la flexibilidad, movilidad y costo reducido lo que las hacen prácticamente indispensables en cualquier solución de infraestructura de redes.

En el presente capítulo se realiza un estudio acerca de los distintos mecanismos de seguridad de las Redes Inalámbricas de Área Local (del inglés *Wireless Local Area Network, WLAN*), se evalúan las principales vulnerabilidades asociadas a este tipo de redes y se demuestran las diversos procedimientos que siguen los atacantes.

1.1 Definición de redes inalámbricas Wi-Fi

Las redes locales inalámbricas o popularmente conocidas como redes Wi-Fi son redes privadas de mediano alcance que se encuentran en el grupo de las redes de área local. Las redes de área local son usadas para la conexión entre computadoras personales y dispositivos electrónicos para permitir el intercambio de recursos e información alrededor de un edificio o local. Están normadas bajo el estándar 802.11 de la IEEE y se utilizan como complemento indispensable de las redes locales cableadas. El nombre de redes Wi-Fi proviene de la asociación industrial no lucrativa Wi-Fi *Alliance*, cuya primicia es asegurar la interoperatibilidad de los productos WLAN a través de una prueba de certificación basada en los estándares IEEE 802.11 (Alliance, 2015, K.Sandhu, 2013).

1.1.1 Estándar IEEE 802.11

La familia 802.11 consiste en una serie de técnicas de modulación semidúplex que utilizan el mismo protocolo básico a través del espacio radioeléctrico. 802.11-1997 fue el primer estándar, pero 802.11b fue el primero ampliamente aceptado, seguido de 802.11a, 802.11g, 802.11n, y 802.11ac. Otras normas en la familia (c-f, h, j) son modificaciones de servicios y extensiones o correcciones a las especificaciones anteriores (Carbonell, 2013).

Los estándares 802.11b y 802.11g utilizan la banda ISM (del inglés *Industrial Science Medical*) de 2,4 GHz. Estos operan en los Estados Unidos bajo la Parte 15 de las Reglas Federales de Comunicaciones y Reglamentos de la Comisión de Estados Unidos. Debido a esta elección de banda de frecuencia, los equipos que soportan los estándares 802.11b y 802.11g ocasionalmente pueden sufrir interferencia de hornos microondas, teléfonos inalámbricos y dispositivos *Bluetooth*. Los estándares 802.11b y 802.11g controlan su interferencia y la susceptibilidad a la interferencia mediante la utilización de métodos de señalización de espectro expandido de secuencia directa (DSSS) y multiplexación (por división) de frecuencias ortogonales (OFDM). El estándar 802.11a utiliza la banda U-NII 5 GHz, que para gran parte del mundo, ofrece al menos 23 canales que no se superponen en lugar de la banda de frecuencia ISM de 2,4 GHz, donde los canales adyacentes se superponen.

El estándar 802.11 define varios tipos de trama que las estaciones utilizan para las comunicaciones, así como para la gestión y el control de la conexión. Cada trama tiene un campo de control que representa la versión del protocolo 802.11, el tipo de trama y diversos indicadores. Además todas las tramas contienen las direcciones MAC de fuente y destino, número de secuencia de trama, cuerpo de trama y secuencia de verificación de trama (para la detección de errores), (ver figura 1.1).

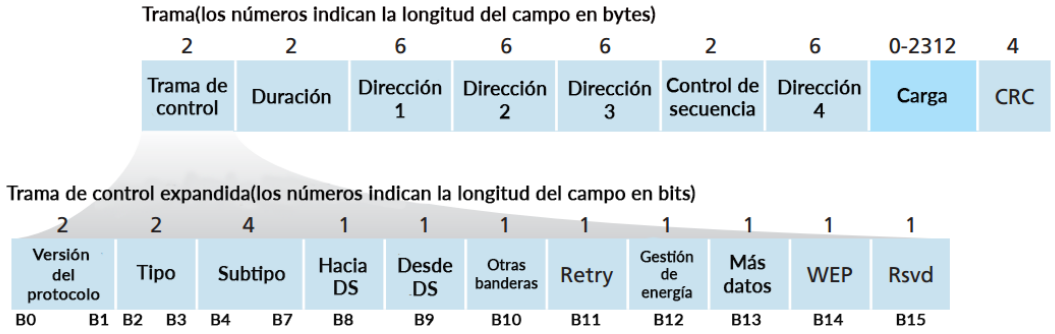

Figura 1.1. Estructura de la trama 802.11.

Las tramas de datos 802.11 transportan protocolos y datos de capas superiores dentro del cuerpo de la trama. Un ejemplo de trama de datos es la que lleva el código HTML de una página web (con cabeceras TCP/IP) hacia un usuario. Las estaciones utilizan otras tramas como las de gestión y control, las cuales llevan información específica sobre el enlace inalámbrico en el cuerpo de trama. Por ejemplo, el cuerpo de una trama baliza (del inglés

beacon), que constituye un tipo de trama de gestión, contiene el identificador de servicio (del inglés *Service Set Identifier,* SSID), fecha, hora, y otras informaciones pertinentes al punto de acceso (Geier, 2002a).

Tramas de gestión 802.11

Principalmente, 2 tipos de tramas 802.11 son las más comúnmente usadas, las de gestión y la de datos. Las tramas de datos son usadas como portadoras a protocolos superiores, mientras que las de gestión portan información específica relacionada a operaciones en el enlace o conexión. Históricamente las tramas de gestión son una debilidad fundamental en la seguridad 802.11. En la mayoría de las soluciones inalámbricas las tramas de gestión nunca han sido aseguradas, y por lo tanto son accesibles a pesar de los mecanismos de encriptación y autenticación. Este tipo de tramas permiten a las estaciones la autenticación, asociación y sincronización (Geier, 2002).

Los tipos de trama de gestión más importantes son (los cuatro bits significan el subtipo), (Systems, 2014):

- **Petición de baliza (0100):** Es una trama utilizada por cualquier STA (estación) para buscar de forma activa un punto de acceso (del inglés *Access Point*, AP) o un BSS.
- **Respuesta de baliza (0101):** Es una trama enviada en respuesta a una Petición de Baliza que contiene información acerca del emisor y de la red.
- **Autenticación (1011):** Las tramas de autenticación son enviadas y recibidas por una STA que solicite realizar una conexión. Son enviadas tanto por el punto de acceso al que la estación intenta autenticar, como por la STA que desee conectarse. El número de tramas de este tipo que se envíen depende del tipo de método de autenticación empleado.
- **Petición de asociación (0000):** Trama enviada a un AP (en un BSS) o a otro cliente (en un IBSS o red *ad-hoc*). La STA que envíe la trama debe estar autenticada.
- **Respuesta de asociación (0001):** Trama enviada desde un AP (BSS) o desde un cliente en respuesta a una petición de asociación. Si la asociación resulta exitosa, el paquete incluye el Identificador de Asociación (del inglés *Association ID*, AID) de la STA que solicitó la asociación.

- **Disociación (1010):** Funciona a modo de declaración para informar que se va a interrumpir la comunicación. Puede ser enviado por cualquier STA.
- **Deautenticación (1100):** Esta trama le indica a la STA que ya no se encuentra autenticada. Es una comunicación unidireccional desde la STA al punto de acceso que gestiona la autenticación. Tiene efecto inmediato al recibirse.

1.1.2 Estándares y enmiendas IEEE 802.11

Dentro del Grupo de Trabajo de la IEEE 802.11, existen los siguientes estándares y enmiendas:

- IEEE 802.11-1997: Estándar original WLAN que soporta razones de 1 Mbit/s y 2 Mbit/s, a través de radiofrecuencia a 2.4 GHz e infrarrojo (IEEE, 1997).
- IEEE 802.11a: Soporta razones a 54 Mbit/s, en la banda de 5 GHz (IEEE, 1999).
- IEEE 802.11b: Mejora al estándar 802.11 para soportar razones de 5.5 Mbit/s y 11 Mbit/s (IEEE, 1999a).
- IEEE 802.11g: Soporta razones hasta 54 Mbit/s, en la banda de 2.4 GHz y es compatible con 802.11b (IEEE, 2003)
- IEEE 802.11i: Enmienda al estándar original IEEE 802.11, implementado como "Acceso Protegido Wi-Fi 2" (del inglés *Wi-Fi Protected Access* II, WPA2). Especifica los mecanismos de seguridad para redes inalámbricas, para reemplazar la cláusula de autenticación y privacidad del estándar original por una cláusula detallada de seguridad. La enmienda elimina la "Privacidad Equivalente a Cableado" (del inglés *Wired Equivalent Privacy,* WEP) como especificación de seguridad, mientras se incorporaba en la publicación del estándar (IEEE, 2004).
- IEEE 802.11-2007: Una nueva entrega del estándar 802.11 con las enmiendas a, b, d, e, g, h, i, y j (IEEE, 2007).
- IEEE 802.11k: Especifica un conjunto de medidas sobre el estado del interfaz de radio (potencia, ocupación de canales, número de tramas recibidas, etc.) que pueden utilizar los equipos de red y las terminales Wi-Fi para hacer un uso más eficiente de los recursos de radio disponibles. 802.11k facilita la selección de un nuevo AP con lo que

ayuda a reducir el tiempo necesario para el cambio, complementando a los procedimientos de 802.11r (IEEE, 2008).

- IEEE 802.11n: Ofrece una velocidad de transferencia hasta 600 Mbps, en las frecuencias de 2,4 GHz y 5 GHZ. Utiliza MIMO para la mejora del desempeño (IEEE, 2009).
- IEEE 802.11p: Acceso inalámbrico en ambientes vehiculares (como ambulancias y automóviles de pasajeros), (IEEE, 2010).
- IEEE 802.11v: Gestión de redes inalámbricas (IEEE, 2011).
- IEEE 802.11w: Tramas de gestión protegidas (IEEE, 2009a).
- IEEE 802.11-2012: Nueva entrega del estándar que incluye las enmiendas k, n, p, r, s, u, v, w, y, y z (IEEE, 2012).
- IEEE 802.11ac: Estándar de WLAN aprobado el 7 de enero del 2014, conocido también como 5G Wi-Fi. De muy alto rendimiento, opera en la banda de frecuencia de 5GHz y por debajo de los 6 GHz. Con una velocidad de datos máxima que se encuentra en el rango de 1 Gbps y con potenciales mejoras sobre 802.11n en cuanto al esquema de modulación (aproximadamente un 10% de aumento en el rendimiento). El ancho de banda de los canales es mayor y se encuentran en el rango de 80 a 160 MHz. Este estándar soporta la tecnología MU MIMO (del inglés *Multi-User Multiple-Input, Multiple-Output*). Mediante el uso de la tecnología de antenas inteligentes MU MIMO permite un uso más eficiente del espectro, brinda una mayor capacidad del sistema y reduce la latencia ya que permite las transmisiones simultáneas de varios usuarios (IEEE, 2013).
- IEEE 802.11ad: Alto caudal en la banda de 60 GHz (IEEE, 2012a).
- IEEE 802.11af: TV *Whitespace*, también referido como *White-Fi* o *Super* Wi-Fi, es un estándar que permite a las redes de área local inalámbricas usar el espectro blanco de la bandas VHF y UHF entre 54 y 790 MHz (IEEE, 2013a).

Enmiendas en proceso (Chatzimisios, 2014):

- IEEE 802.11mc: Integración de 802.11-2012 con las enmiendas aa, ac, ad, ae y af Debe ser publicada como 802.11-2015, en diciembre del 2015.
- IEEE 802.11ah: Operación exenta de licencia en la banda debajo de 1 GHz. Se estima su publicación en marzo del 2016.
- IEEE 802.11ai: Establecimiento Inicial Rápido del Enlace. Pretende estandarizar una función de establecimiento rápido de enlace que permitirá al cliente alcanzar un establecimiento seguro del enlace dentro de 100ms. Se estima su publicación en noviembre del 2015.
- IEEE 802.11aj: Onda Milimétrica China. Versión de 802.11ad para el uso del espectro sin licencia en los 45 GHz, disponible en algunas regiones del mundo (específicamente en China). Se estima su publicación en junio del 2016.
- IEEE 802.11ak: Para el soporte de tránsito general dentro de redes puenteadas. Mejora la habilidad de trabajo de las redes inalámbricas 802.11 con dispositivos en la red cableada dentro de una Red Virtual de Área Local (del inglés *Virtual Local Area Network,* VLAN). Se estima su publicación en mayo del 2016.
- IEEE 802.11aq: Descubrimiento de pre-asociación. Responde al problema de como una estación (como un dispositivo móvil) descubre la disponibilidad de servicios dentro de una red donde otra estación se encuentra conectada. Se estima su publicación en julio del 2016.
- IEEE 802.11ax: Redes de área local inalámbricas de alta eficiencia. Se estima su publicación en mayo del 2018.

1.2 Beneficios de las redes inalámbricas

Las redes inalámbricas poseen múltiples beneficios en una solución de redes cualesquiera. Se pueden clasificar en su generalidad en beneficios funcionales y de seguridad.

Beneficios funcionales de las redes inalámbricas

Las principales ventajas de la implementación de las redes inalámbricas son (Bosworth, 2014, K.Sandhu, 2013, Osterhage, 2011):

- Movilidad
- Flexibilidad
- Reducción de costos

Movilidad: Las tecnologías inalámbricas permiten el acceso a la información de la red a través de terminales móviles, a medida que se mueven alrededor de su lugar de residencia o trabajo. Dentro de un entorno empresarial, las tecnologías inalámbricas ofrecen una alternativa flexible en adición a la red cableada. Usualmente en un ambiente de trabajo se cuenta con un número limitado de conexiones *Ethernet*; las tecnologías inalámbricas pueden proporcionar conexiones de red adicionales según sea necesario.

Flexibilidad: Las redes inalámbricas públicas (en inglés *hotspots*) permiten al personal, en el tiempo de inactividad, a la conectividad. Los usos típicos incluyen el acceso a la LAN corporativa, junto al acceso a internet.

Reducción de Costos: Los costos se reducen ante la inexistencia de enlaces físicos. Un enlace inalámbrico se puede configurar entre dos puntos siempre que haya línea de vista. Un único punto de acceso puede dar servicio a uno o muchos usuarios finales y escalar apropiadamente, cediendo a los usuarios a AP vecinos. Esta capacidad es exclusivamente inalámbrica. El uso de la tecnología inalámbrica con VLAN puede reducir el volumen de *hardware* de red. El ahorro de costos se puede realizar dentro de una LAN inalámbrica mediante la reducción de las tareas de gestión de red y la complejidad global, con el objetivo de maximizar los recursos y el uso del *hardware*.

Beneficios de seguridad en las redes inalámbricas

Seguridad física: Un punto de acceso se puede ocultar de los usuarios finales para protegerlo de los ataques físicos. La ubicación en lugares de difícil acceso de estos dispositivos contribuye a la seguridad, en contraste con las tomas físicas presentes en la red cableada, que deben ser accesibles para todos los usuarios que necesitan acceso a la red interna (López, 2010).

Visibilidad de segmentación: En las redes cableadas comúnmente se asignan las VLAN en función de los puertos, o en configuraciones avanzadas, se utilizan las direcciones MAC. En

la asignación de VLAN en función de los puertos, la gestión depende en gran medida de la configuración del conmutador (del inglés *switch*) y el diseño que se realice en la red. La asignación de VLAN basada en direcciones MAC requiere la presencia de autenticación MAC como característica en los conmutadores y un servidor *Radius* para asignaciones MAC→VLAN. Las asignaciones MAC basadas en VLAN pueden mejorar la capacidad de gestión y supervisión, pero plantean un riesgo de seguridad, ya que son susceptibles al ataque de suplantación de MAC. En un entorno inalámbrico, las VLAN se pueden asignar en función de cada SSID y los esfuerzos administrativos pueden reducirse en gran medida. De esta manera un usuario no se limita a una ubicación física para el acceso a una VLAN, siempre que el SSID se encuentre disponible en los APs accesibles (Bosworth, 2014).

1.3 Mecanismos de seguridad en redes Wi-Fi

Los mecanismos de seguridad que se pueden aplicar en las redes WLAN son diversos y estos actúan en las diferentes capas del modelo OSI. Estos mecanismos tienen puntos de contacto con las redes cableadas pero difieren significativamente en las dos primeras capas del modelo OSI. En esta sección se tratan algunos de los mecanismos de seguridad que presentan las redes Wi-Fi.

1.3.1 Mecanismos de seguridad de la capa de enlace

Protocolo WEP

El protocolo WEP IEEE 802.11 fue diseñado en 1999 para proporcionar autenticación y cifrado de datos entre un cliente y un punto de acceso inalámbrico mediante el uso de un enfoque de clave compartida simétrica. WEP no especifica un algoritmo de gestión de claves, por lo que se supone que el cliente y el punto de acceso han acordado de alguna manera la clave a través de un método determinado (Chaouchi y Laurent-Maknavicius, 2007, Kurose y Ross, 2013).

El algoritmo de encriptación de datos WEP se ilustra en la figura 1.2. Una clave secreta de 40 bits simétrica, K_S, se supone que es conocida tanto por el cliente y el punto de acceso. Un vector de inicialización (del inglés *initialization vector,* IV) de 24 bits se le añade a la clave de 40 bits para crear una clave de 64 bits que se utiliza para cifrar una sola trama. El vector de inicialización cambia de una trama a otra, por lo tanto cada trama se cifra con una clave diferente de 64 bits. Primeramente un valor CRC de 4 bytes se calcula para la carga de datos.

La carga útil y los cuatro bytes CRC se cifran mediante el uso del algoritmo de cifrado RC4 (del inglés *Rivest Cipher 4*).

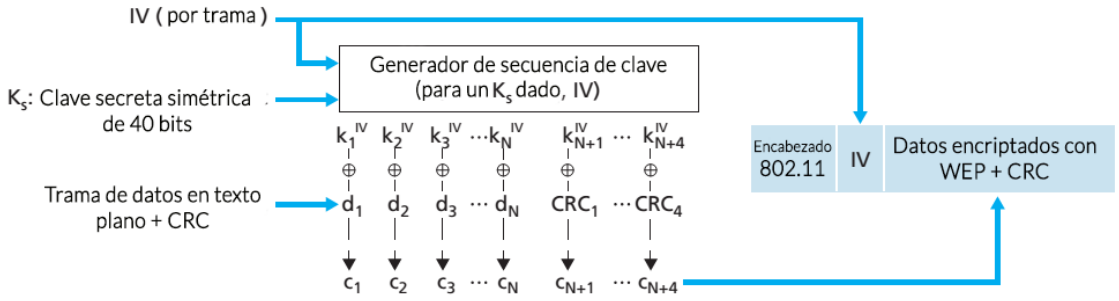

Figura 1.2. Protocolo 802.11 WEP.

Cuando se presenta una clave (en este caso, de 64 bits (K_S, IV)), el algoritmo RC4 produce una serie de valores k_1^{IV}, k_2^{IV}, k_3^{IV},. . . que se utilizan para cifrar los datos y el valor CRC en una trama. Se puede pensar que estas operaciones se están realizando un byte a la vez. El cifrado se realiza realizando un XOR en el byte i-ésimo de datos, d_i, con la clave i-ésima, k_i^{IV}, en la serie de valores de clave generados por el par (K_S, IV) para producir el i-ésimo byte del texto cifrado, c_i (ver ecuación 1), (Kurose y Ross, 2013).

$$c_i = d_i \oplus k_i^{IV} \qquad (1)$$

El valor IV cambia de una trama a la siguiente y se incluye en texto plano en la cabecera de cada trama 802.11 con encriptación WEP, como se muestra en la figura 1.2. El receptor toma la clave simétrica secreta de 40 bit que comparte con el remitente, y le añade el IV, utilizando la clave de 64 bits resultante (que es idéntica a la clave utilizada por el remitente para realizar el cifrado) para desencriptar la trama (ver ecuación 2), (Kurose y Ross, 2013).

$$d_i = c_i \oplus k_i^{IV} \qquad (2)$$

El uso adecuado del algoritmo RC4 requiere que el mismo valor de clave de 64 bits no puede utilizarse más de una vez. Debe tenerse en cuenta que la llave WEP varía de trama en trama. Para una K_S dado (que cambia raramente, o nunca), esto significa solo existen 2^{24} claves únicas. Si estas claves se eligieran al azar, se puede demostrar que la probabilidad de elegir el mismo valor IV (y por lo tanto se utiliza la misma clave de 64 bits) es más del 99 por ciento después de sólo 12000 tramas. Con 1 Kbyte de tamaño de trama y una velocidad de transmisión de datos de 11 Mbps, sólo unos pocos segundos son necesarios para llegar a

12000 tramas. Además, puesto que la IV se transmite en texto plano en la trama, un atacante sabrá siempre que se utilice un valor duplicado IV. Si un atacante realiza una petición al punto de acceso con contenido conocido d_1, d_2, d_3, d_4,. . . este observa la respuesta encriptada c_1, c_2, c_3, c_4,. . . si se realiza un XOR con c_i en ambos miembros de la ecuación 2 se tiene en la ecuación 3 (Kurose y Ross, 2013):

$$d_i \oplus c_i = k_i^{IV} \qquad (3)$$

Con esta relación, el atacante puede usar los valores conocidos de d_i y c_i para calcular k_i^{IV}. La próxima vez que el atacante observe el mismo valor de IV, este conocerá la secuencia de la clave k_1^{IV}, k_2^{IV}, k_3^{IV},. . . siendo capaz de desencriptar el mensaje encriptado.

Existen varios problemas de seguridad adicionales con WEP. Fluhrer en (Fluhrer et al., 2001) describe un ataque aprovechando una debilidad conocida en RC4 cuando se eligen ciertas claves débiles. Stubblefield en (Stubblefield et al., 2002) presenta formas eficientes para implementar y explotar este ataque. Otra preocupación con WEP implica a los bits CRC que se muestran en la figura 1.2 y transmitidos en la trama de 802.11 para detectar los bits alterados en la carga útil. Un atacante que cambie el contenido cifrado (por ejemplo, la sustitución de datos aleatorios por los datos cifrados originales), calcule una CRC sobre los datos sustituidos, y coloque el CRC en una trama WEP puede producir una trama 802.11 que será aceptado por el receptor (Kurose y Ross, 2013).

IEEE 802.11i

En junio del 2004, la IEEE publicó el estándar 802.11i para mejorar la seguridad de las redes 802.11. Este nuevo sistema se denomina Redes de Seguridad Robusta (del inglés *Robust Security Network, RSN)* y está diseñado para usuarios personales o empresariales. El uso empresarial se basa en el protocolo 802.1X para proporcionar autenticación y establecer un contexto de seguridad. Un perfil "personal" utiliza una clave pre-compartida (PSK), basado en contraseña (Weidman, 2014). RSN provee control de acceso basado en autenticación robusta de las capas superiores. El rol principal de RSN es garantizar seguridad, movilidad, integridad y confiabilidad como escalabilidad y flexibilidad. (Chaouchi y Laurent-Maknavicius, 2007)

Debido a las fallas en la seguridad de WEP, se adicionaron mecanismos adicionales a 802.11i:

– Protocolo de Integridad de Clave Temporal (del inglés *Temporal Key Integrity Protocol*, TKIP), sucesor de WEP.

– CCMP (del inglés *Counter Mode with Cipher Block Chaining Message Authentication Code Protocol*), usa algoritmo de encriptación AES en modo CCM (del inglés *Counter with CBC-MAC*) y una firma MIC (del inglés *Message Integrity Code*).

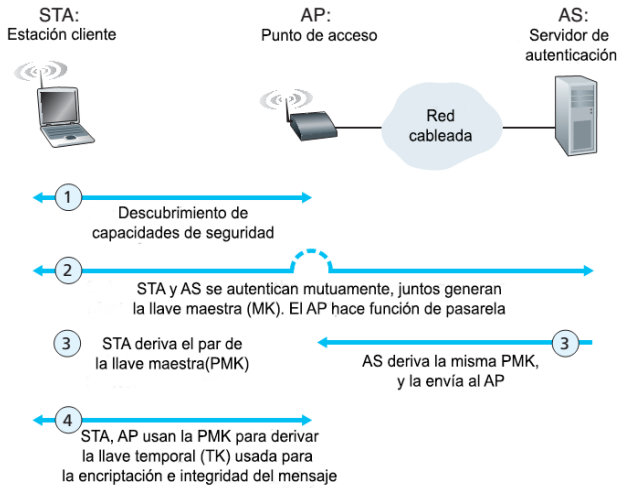

Figura 1.2. Proceso de autenticación en 802.11i (802.1X)

Protocolo TKIP

TKIP es una alternativa a WEP que repara los problemas de seguridad y no requiere *hardware* diferente que el necesario para soportar a WEP. En otras palabras el *hardware* soportado por WEP también soporta TKIP, siempre y cuando tenga el *firmware* del dispositivo actualizado. Como WEP, TKIP usa el cifrado de flujo RC4 tanto para encriptar como para desencriptar y todos los dispositivos involucrados deben conocer la misma llave secreta. Esta llave secreta es de 128 bits y es llamada clave temporal (del inglés *Temporal Key,* TK). TKIP usa un vector de inicialización (IV), denominado también contador de secuencia (del inglés *TKIP Sequence Counter*, TSC), de 48 bits, el cual es suficiente para transmitir 218474976710656 paquetes sin repetir el IV. Utiliza una jerarquía de claves lo cual le permite fortalecer aún más la seguridad. El cambio de la clave temporal, a diferencia de WEP, se hace cada 10000 paquetes. A pesar de que el TK es compartido, todos los

participantes involucrados generan una clave RC4 diferente, esto es debido a que todos los participantes de la comunicación realizan dos fases de generación de una única clave por paquete (del inglés *Packet Per Key*, PPK), para esto se usa la dirección MAC de emisor, la clave secreta, el IV de 48 bits, teniendo así claves diferentes para cada dirección que opera sobre el enlace (ver figura 1.3). Por ultimo TKIP usa un Código de Integridad de Mensaje (del inglés *Message Integrity Code*, MIC), diseñado para el *hardware* existente. Su objetivo es detectar las modificaciones en el mensaje. El MIC es una función criptográfica de un solo sentido y es calculado en base de las direcciones MAC origen, destino y texto plano (datos), (Chiu, 2006).

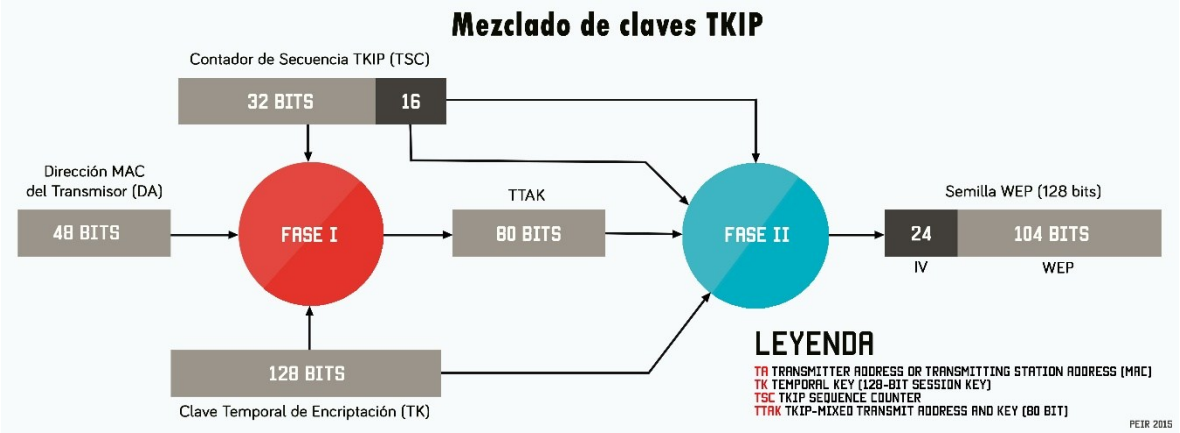

Figura 1.3. Mezclado de claves TKIP.

Protocolo CCMP

Protocolo al complementario al TKIP, que representa un nuevo método de encriptación basado en AES, un cifrado simétrico que utiliza bloques de 128 bits, con el algoritmo CBC-MAC. Así como el uso de TKIP es opcional, la utilización del protocolo CCMP es obligatorio en 802.11i. CCMP utiliza un vector de inicialización (IV) de 48 bits denominado Número de Paquete (del inglés *Packet Number*, PN) utilizado a lo largo del proceso de cifrado, junto con la información para inicializar el cifrado AES para calcular el MIC y la encriptación de la trama. En este proceso de encriptación, se utiliza la misma clave temporal tanto para el cálculo del MIC como para la encriptación del paquete. El MIC se calcula a partir de un IV formado por el PN y datos extraídos de la cabecera de la trama. El IV se convierte en un bloque AES y su salida a través de la operación XOR conformara el siguiente bloque AES (Chiu, 2006).

Acceso Protegido Wi-Fi (Wi-Fi *Protected Access*, WPA)

WPA fue desarrollado por la Wi-Fi *Alliance* para mejorar el cifrado existente en WEP así como para incorporar un método de autenticación (Alliance, 2003). WPA opera a nivel MAC y está basado en el borrador del estándar IEEE 802.11i, aunque presenta algunas carencias que la versión final de IEEE 802.11i no tiene. Las principales características de WPA son las siguientes:

- Distribución automática de claves.
- Sesión dinámica de llaves. Por usuario, por sesión, por llaves de paquetes.
- Utilización más robusta del vector de inicialización (mejora de la confidencialidad).
- Nuevas técnicas de integridad y autenticación aplicables en entornos residenciales y empresariales.
- Actualización de equipos a WPA mediante *software*.
- En WPA es posible emplear dos modos de autenticación diferentes en dependencia del entorno de aplicación.
- WPA con clave pre-compartida (WPA-PSK) para entornos residenciales.
- WPA con los mecanismos IEEE 802.1X y EAP (del inglés *Extensible Authentication Protocol*) para entornos empresariales.

En cuanto al cifrado, WPA consigue disminuir las vulnerabilidades conocidas de WEP mediante la utilización del cifrado RC4 implementando mejoras como la creación de un IV (vector de inicialización) extendido de 48 bits y reglas de secuenciamiento del mismo; la implementación de nuevos mecanismos de derivación y distribución de claves; la incorporación de TKIP, el cual se encarga de la generación de la clave para cada trama y empleando el algoritmo de cifrado RC4 elimina el problema de las claves estáticas compartidas implementado en WEP. Como el algoritmo de cifrado WEP y WPA es el mismo RC4, en el caso de una red WLAN con equipamiento WEP, únicamente es necesario una actualización del *software* en las estaciones de los usuarios y en los AP, sin llevar a cabo ningún cambio de *hardware* (Alliance, 2003).

Acceso Protegido Wi-Fi 2 (Wi-Fi Protected Access 2, WPA2)

WPA2 fue diseñado para resolver las necesidades de muchas organizaciones, buscando una tecnología interoperable y certificada basada en el estándar completo de IEEE 802.11i. IEEE 802.11i y WPA2 son virtualmente idénticos, siendo mínimas las diferencias entre ellos, ambos emplean como código de cifrado AES/CCMP y añaden, opcionalmente, pre-autenticación a WPA. La principal diferencia de WPA2 respecto a WPA es que emplea un mecanismo de cifrado más avanzado (Arana, 2006). No obstante, WPA2 es compatible con WPA, por lo que algunos productos WPA pueden ser actualizados a WPA2 por *software*. Sin embargo, en otros casos es necesario un cambio del *hardware* debido a los altos requerimientos de cómputo del cifrado AES (García, 2011).

Al igual que en el caso de los productos WPA, la Wi-Fi *Alliance* ha establecido dos modos de autenticación para los productos, WPA2 con PSK para ambientes residenciales y WPA2 con IEEE 802.1X/EAP para ambientes empresariales. A diferencia de WPA, el estándar WPA2 puede asegurar tanto WLAN en modo infraestructura como también en modo ad hoc (García, 2011). Por otra parte WPA2 carece de ciertos aspectos con los que cuenta IEEE 802.11i para proporcionar servicios de voz inalámbricos, como prevenir la latencia de la señal o la pérdida de información durante el *roaming* (García, 2011).

Tabla 1.1 Comparación entre protocolos de seguridad inalámbrica: WEP, WPA, WPA2 (Sukhija y Gupta, 2012)

	WEP	**WPA**	**WPA2**
Propósito	Proveer seguridad, comparable a redes cableadas	Soluciona las fallas de WEP sin requerir nuevo *hardware*. Implementación del estándar IEEE 802.11i.	Implementa completamente el estándar IEEE 802.11i y una mejora sobre WPA.
Privacidad de datos (Encriptación)	Cifrado Rivest 4 (RC4).	Protocolo de integridad temporal de llaves (TKIP).	*Counter Mode con Cipher block Chaining Message Authentication Code Protocol* (CCMP) mediante la utilización del cifrado de bloque AES.
Autenticación	WEP-Abierta y WEP-Compartida.	WPA-PSK y WPA-*Enterprise.*	WPA2-Personal y WPA2-*Enterprise.*

Integridad de datos	CRC-32	Michael (genera Código de Integridad de Mensaje).	*Cipher block chaining message authentication code* (CBC-MAC).
Gestión de llaves	Ausencia de gestión de llaves.	Provee gestión robusta de llaves y las llaves son generadas a través de un *four way handshake.*	Provee gestión robusta de llaves y las llaves son generadas a través de un *four way handshake.*
Compatibilidad de *hardware*	Funcionamiento en *hardware* existente.	Funcionamiento en *hardware* existente a través de actualizaciones de *firmware* en las NIC (del inglés *Network Interface Card*).	Soportado por dispositivos Wi-Fi con certificados desde el año 2006. No funciona con NICs más antiguas.
Ataques/Vulnerabilidades	*Chopchop*, Fragmentación de *Bittau*, FMS y ataque PTW, ataques DoS.	*Chopchop*, *Ohigashi-Morii*, WPA-PSK, *Beck-Tews*, ataque *Michael Reset*, vulnerabilidad *Hole* 196 y ataques DoS.	Vulnerabilidad *Hole* 196, ataques DoS debido a la gestión y tramas de control desencriptadas, personificación de dirección MAC debido a la deautenticación, Ataques offline de diccionario en *WPA2-Personal.*
Complejidad de despliegue	Fácil de instalación y despliegue.	Instalación complicada requerida para WPA-*Enterprise*.	Instalación complicada requerida para WPA2-*Enterprise*.
Protección contra ataques de reenvío	Ninguna protección contra ataques de reenvío.	Implementa contador de secuencia para protección contra ataques de reenvío	Número de paquete de 48 bit previene ataque de reenvío

Estándar 802.1X

802.1X fue diseñado originalmente para el control de acceso en redes IEEE basadas en puertos con infraestructuras *802LAN*. Estas infraestructuras incluyen *Ethernet, Token Ring*, y redes inalámbricas. 802.1X autentica y autoriza a los dispositivos conectados a un puerto LAN, y no permite el acceso a la red si la autenticación falla.

802.1X define tres roles (Chaouchi y Laurent-Maknavicius, 2007):

1. Autenticador. Un dispositivo que autentica a otro dispositivo en la red antes de permitir que acceda a recursos de la red. En una red 802.11 el AP es el autenticador.

2. Suplicante. El dispositivo que desea acceder a los recursos de red y necesita ser autenticado.

3. Servidor de autenticación (del inglés *Authentication Server*, AS). El AS realiza la autenticación real del solicitante en nombre del autenticador. El AS puede ser localizado con el autenticador, pero es comúnmente un sistema externo, como un servidor *Radius* (Weidman, 2014). Otra alternativa sería la utilización del protocolo *Diameter* recientemente estandarizado en la RFC 3588 (Kurose y Ross, 2013).

El estándar 802.1X define el objeto Entidad de Puertos de Acceso (del inglés *Port Access Entity,* PAE), que controla los algoritmos de autenticación y los protocolos entre el suplicante y el autenticador. Una visión general del 802.1X se muestra en la figura 1.4:

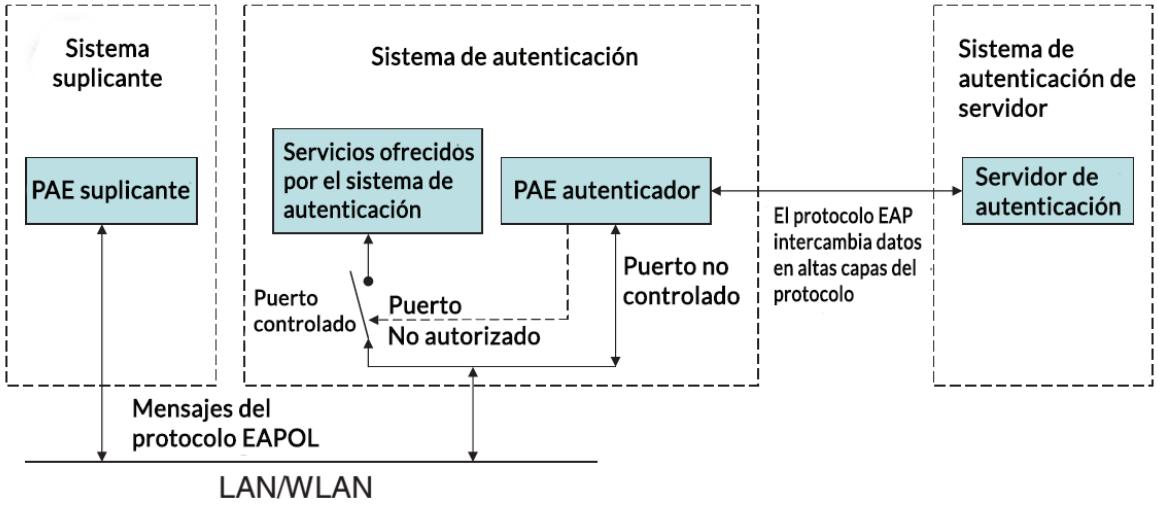

Figura 1.4. Estructura de 802.1X.

802.1X se basa en el Protocolo de Autenticación Extensible (del inglés *Extensible Authentication Protocol*, EAP) para realizar la autenticación entre el suplicante y el autenticador. EAP fue originalmente diseñado para el uso en redes *modem dial-up*; por lo tanto, la especificación 802.1X detalla la expansión de EAP (RFC 3748) a través de redes *Ethernet/Token Ring* y de la extensión EAPoL (*EAP over LAN*).

Las implementaciones más comunes de los protocolos EAP/EAPoL clasificados por el nivel de seguridad son los siguientes:

1. **EAP-TLS**: Implementación EAP que permite la autenticación basada en certificados X509 a través de la capa de Transporte de Seguridad (del inglés *Transport Layer Security*, TLS). La autenticación entre el AS y el cliente inalámbrico proporciona un alto nivel de seguridad inalámbrica, pero requiere el uso de una Infraestructura de Clave Pública (del inglés *Public Key Infrastructure,* PKI) para distribuir de forma segura y actualizar las claves del cliente de manera regular. Un problema con EAP-TLS es que la mayoría de las organizaciones no tienen la PKI necesaria para emitir los certificados TLS cliente-suplicante.

    - EAP-TTLS (TLS Tunelizado): EAP-TTLS es similar a EAP-TLS y soporta la autenticación de certificado mutuo, pero no requiere certificados de cliente. EAP-TTLS crea un túnel TLS antes de comenzar un proceso de autenticación de red y, por consiguiente puede tunelizar cualquier mecanismo de autenticación de contraseña, incluso mecanismos inseguros como PAP.

2. **EAP-PEAP** (EAP protegido): En su forma nativa, EAP-PEAP no admite autenticación mutua basada en certificados. El EAP-PEAP nativo utiliza solamente TLS para autenticar al servidor de directorio y realiza un proceso de desafío-respuesta basada en contraseña para autenticar al cliente. Extensiones PEAP posteriores contribuyen a mitigar esta debilidad. EAP-PEAP es nativa en la mayoría de las versiones de *Microsoft Windows*, por lo que es muy frecuente en la industria inalámbrica. Actualmente existen tres tipos principales de EAP-PEAP, que ofrecen distintos niveles de protección:

    - **EAP-PEAP-MS-CHAP-v2** (Protocolo de Desafío Mutuo de Microsoft de Autenticación, del inglés *Microsoft Challenge Handshake Authentication Protocol*,): El método de autenticación interna EAP-PEAP más común utiliza el protocolo CHAP-v2 de *Microsoft* para proporcionar facilidad de identificación (ID de usuario) y contraseña en una autenticación basada en desafío-respuesta.

- **EAP-PEAP-TLS:** PEAP-TLS es el segundo protocolo PEAP interior definido por *Microsoft*. PEAP-TLS tuneliza el protocolo EAP-TLS dentro de PEAP para la autenticación basada en certificados mutuos X509.
- **EAP-PEAPv1 (EAP-GTC):** La tercera implementación está definida por Cisco, que permite la autenticación mediante tarjetas genéricas señalizadas como *RSA's SecurID token* y además a través de usuario y contraseña.

3. **EAP-LEAP**: Un protocolo desarrollado por Cisco, que desarrolla MS-CHAP-v2 con desafío-respuesta, basado en autenticación por nombre de usuario y contraseña en texto plano. Un atacante cuyo objetivo sea una red que emplee EAP-LEAP sólo tiene que supervisar el tráfico para capturar los *hashes* desafío-respuesta, para luego realizar ataque *offline* de diccionario. En contraste con EAP-PEAP-MS-CHAP-v2, que requiere un ataque de personificación de punto de acceso para interceptar los *hashes* MS-CHAP v2.
4. **EAP-FAST** (Autenticación Flexible a través de Tunelización Segura): Desarrollado por Cisco como reemplazo para el protocolo LEAP. EAP-FAST introdujo túneles seguros de pre-autenticación sin uso de certificados.
5. **EAP-MD5**: Es la base del soporte EAP brindado por los dispositivos que soportan 802.1X. Es el primer tipo de EAP que duplican las operaciones CHAP. Como EAP-MD5 no provee autenticación con servidor, es vulnerable a los tipos de ataque de personificación de servidor**.** Al escoger soluciones o productos 802.1X para la red inalámbrica, debe proveerse una autenticación mutua entre autenticador y AS con el objetivo de reducir los ataques MITM.

   Protocolos EAP inseguros: Cuando EAP se integra inicialmente al estándar 802.11, se utilizó comúnmente los protocolos EAP-MD5 y EAP-LEAP de Cisco. Por desgracia, estas resultaron ser vulnerables a los ataques, lo que estimuló el desarrollo de los protocolos seguros como los mencionados. EAP-MD5 y EAP-LEAP no se deben utilizar para la implementación de una red inalámbrica empresarial (Weidman, 2014), (ver tabla 1.2).

Tabla 1.2 Resumen de métodos de autenticación EAP

Propiedad	Método de autenticación EAP					
	MD5	LEAP	**TLS**	TTLS	PEAP	FAST
Atributos de autenticación	Unilateral	Mutuo	**Mutuo**	Mutuo	Mutuo	Mutuo
Dificultad de implementación	Fácil	Fácil	**Difícil**	Moderado	Moderado	Moderado
Regeneración de llaves dinámica	No	Si	**Si**	Si	Si	Si
Certificado de servidor	No	No	**Si**	Si	Si	No
Certificado del cliente	No	No	**Si**	No	No	No
Tunelizado	No	No	**No**	Si	Si	Si
WPA compatible	No	Si	**Si**	Si	Si	Si
Seguridad WLAN	Pobre	Moderada	**Muy Fuerte**	Fuerte	Fuerte	Fuerte
Riesgos de seguridad	Exposición de identidad, ataques de diccionario, ataque MITM	Exposición de identidad, ataques de diccionario	**Exposición de identidad**	Ataque MITM	Ataque MITM, identidad escondida en fase 2 pero exposición potencial en fase 1.	Ataque MITM

Vulnerabilidades en el protocolo 802.1X

En el año 2005, Steve Riley, perteneciente a la Unidad Tecnológica y de Seguridad de Negocios de *Microsoft* publica un artículo, donde se detalla una vulnerabilidad grave en el protocolo 802.1X, el cual involucraba el ataque de hombre en el medio.

En resumen, la falla radica en el hecho que 802.1X solo autentica al comienzo de la conexión, pero después que ocurre esta, existe la posibilidad de que un atacante use el puerto autenticado, si este posee la habilidad de insertarse físicamente entre la computadora autenticada y el puerto (Riley, 2005).

La especificación 802.1X-2010, que comienza como 802.1af, soluciona las vulnerabilidades en las versiones previas de 802.1X, mediante el uso de MACSec IEEE 802.1ae para la encriptación de datos entre puertos lógicos (IEEE, 2010a, IEEE, 2010b).

1.3.2 Mecanismos de seguridad del nivel de red

Red Privada Virtual utilizando IPsec (IPsec VPN)

Los protocolos de IPsec (del inglés *IP Security*) se definieron originalmente en las RFCs 1825 (Atkinson, 1995) y 1829 (IETF, 1995), publicadas en 1995, pero posteriormente fue actualizado en 1998 y 2005 RFC 4301 (IETF, 2005) y RFC 4309 (IETF, 2005a). El objetivo principal de IPsec es proporcionar protección a los paquetes IP, es decir, establece comunicaciones IP con seguridad de extremo a extremo, lo que significa que los nodos intermedios utilizan el protocolo IP sin necesidad de una implementación específica para IPsec. Al establecer la comunicación, IPsec permite utilizar dos protocolos diferentes: el AH (del inglés *Authentication Header*) y ESP (del inglés *Encapsulation Security Payload*). AH únicamente verifica la integridad del paquete (mediante firma digital), y ESP cifra la información (DES (Daley, 1999), 3DES, AES) y opcionalmente verifica la integridad del paquete. Existen dos modos de operación de IPsec, el modo transporte y el modo túnel (García, 2011, Pellejero et al., 2006). IPsec no define algoritmos específicos de cifrado, sino que mediante ISAKMP, permite utilizar IKE (Internet Key Exchange) la versión 2 (IKE v2), definido en la RFC 4306 (IETF, 2005a), para realizar una auto-negociación del algoritmo a emplear y del intercambio de claves. La protección del modo de transporte IPSec es transparente para los usuarios, es independiente del hardware de WLAN y no limita la utilización de los algoritmos de cifrado (García, 2011). La utilización de IPSec en lugar de la seguridad de WLAN nativa presenta algunos inconvenientes, por ejemplo el uso de IPSec únicamente para la autenticación a nivel de equipo; la exigencia del cifrado de todo el tráfico de un extremo a otro; algunos dispositivos pueden ser incompatibles con IPSec (García, 2011). Las VPN basadas en IPSec constituyen una solución excelente para muchas

situaciones de seguridad (García, 2011), pero no afrontan la seguridad de la WLAN, debido a que no proporcionan seguridad a nivel de enlace y solo protegen el tráfico IP (García, 2011, Pellejero et al., 2006).

1.3.3 Mecanismos de seguridad del nivel de transporte

Protocolos SSL/TLS

Los protocolos SSL (del inglés *Secure Socket Layer*) y TLS (del inglés *Transport Layer Security*) son protocolos de la capa de transporte que proporcionan comunicaciones seguras en Internet. Los protocolos SSL versión 3.0 y TLS versión 1.0 son muy parecidos. SSL fue diseñado por *Netscape* en 1996 y, a pesar de que no es un protocolo estandarizado por el IETF, este lo estandarizó en 1999 con ligeras modificaciones, aunque el protocolo funcionaba de la misma manera. La primera definición de TLS apareció en el RFC 2246 (IETF, 1999), aunque se han ido publicando otras definiciones relacionadas con la compatibilidad de TLS con otros protocolos y técnicas criptográficas. SSL/TLS permite la autenticación tanto de cliente como servidor, usando claves públicas y certificados digitales, proporcionando comunicación segura mediante el cifrado de la información entre emisor y receptor (García, 2011). SSL/TLS funciona por encima del protocolo de transporte (normalmente TCP) y por debajo de los protocolos de aplicación. Este protocolo está muy extendido para realizar actividades de comercio electrónico y debido a esto muchas de las principales instituciones financieras han aprobado SSL para el comercio sobre Internet (García, 2011). SSL/TLS se compone de cuatro protocolos. Estos protocolos funcionan de manera idéntica en SSL y en TLS, pero incorporan algunos detalles en TLS para su mejor funcionamiento los cuales son *Record Protocol, Handshake Protocol, Change Cipher Spec Protocol y Alert Protocol* (García, 2011).

Red Privada Virtual con protocolo SSL (SSL VPN)

Las VPN basadas en el protocolo SSL posibilitan el acceso a aplicaciones Web, aplicaciones cliente/servidor y a aplicaciones con ficheros compartidos a través de un Gateway SSL VPN que ejerce como proxy (García, 2011, Pellejero et al., 2006). Las SSL VPN aportan ventajas significativas respecto a las IPsec VPN, aunque también implican complejidades y consideraciones a tener en cuenta. La principal ventaja que aportan las SSL VPN es que no es necesario tener instalado ningún software cliente extraen el terminal de usuario, pues lo

único que se necesita para establecer una conexión segura es disponer de un navegador web en el terminal de usuario y de un Gateway SSL VPN en la red corporativa (García, 2011, Pellejero et al., 2006). Se puede decir que las conexiones VPN basadas en SSL son igual de seguras que las basadas en IPsec (García, 2011, Pellejero et al., 2006).

1.3.4 Mecanismos de seguridad del nivel de aplicación

Protocolo SSH (del inglés *Secure Shell*)

SSH es un protocolo que se utiliza para acceder a equipos remotos a través de una red, de forma similar a como lo realiza TELNET. La diferencia principal es que SSH usa técnicas de cifrado para proteger la conexión durante toda la sesión (García, 2011, Pellejero et al., 2006). Además de ser útil para conectarse de forma segura a un equipo remoto, SSH también permite copiar datos de forma segura (tanto ficheros como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectarse a las máquinas y transmitir los datos de cualquier otra aplicación a través de un canal seguro SSH (García, 2011, Pellejero et al., 2006). Existen dos versiones del protocolo SSH. La versión 1 de SSH es un protocolo monolítico, SSH-2 es un protocolo de 4 capas: de transporte, autenticación de usuario, conexión y una llamada "SSHFP DNS" que se encarga de las firmas de los servidores (García, 2011). El protocolo SSH proporciona los siguientes tipos de protección (Hat, 2005):

- Después de la conexión inicial, el cliente puede verificar que se está conectando al mismo servidor al que se conectó anteriormente.
- El cliente transmite su información de autenticación al servidor usando una encriptación robusta de 128 bits.
- Todos los datos enviados y recibidos durante la sesión se transfieren por medio de encriptación de 128 bits, siendo para un atacante extremamente difícil de descifrar y leer.
- El cliente tiene la posibilidad de reenviar aplicaciones X11 desde el servidor. Esta técnica, llamada reenvío por X11, proporciona un medio seguro para usar aplicaciones gráficas sobre una red.

Debido a que el protocolo SSH encripta todo lo que envía y recibe, se puede usar para asegurar protocolos inseguros. El servidor SSH puede convertirse en un conducto para

convertir en seguros los protocolos inseguros mediante el uso de una técnica llamada reenvío por puerto, incrementando la seguridad del sistema en general y de los datos (García, 2011).

1.4 Principales vulnerabilidades y ataques en las redes Wi-Fi

Las vulnerabilidades en las redes Wi-Fi se pueden agrupar en dos grupos principales: en el primer grupo se encuentran las vulnerabilidades causadas por malas configuraciones de seguridad y en el segundo se encuentran las causadas por fisuras en los protocolos que conforman la capa de enlace encargados de la autenticación y confidencialidad. Los ataques y las vulnerabilidades son dos aspectos inseparables, ya que la manera más eficiente de probar una determinada debilidad es a través de la realización de un ataque. Muchos autores dividen a los ataques en dos tipos: los de tipo pasivo y los de tipo activo. Entre los de tipo pasivo se puede mencionar el espionaje (de inglés *eavesdropping)* y entre el tipo activo se encuentra el ataque de personificación, de renvío, modificación del mensaje y de denegación de servicio. A continuación se muestra con detalle las vulnerabilidades y ataques más relevantes.

1.4.1 Redes inalámbricas abiertas

Las redes abiertas (en inglés *open wireless*) son vulnerables desde una perspectiva de seguridad, ya que cualquier personal dentro del alcance del punto de acceso puede conectarse a la misma. Además, los paquetes inalámbricos que viajan a través de una red abierta no están cifrados, y cualquier usuario que se encuentre a la escucha puede capturar todos los datos en texto plano. Los datos sensibles pueden ser encriptados mediante protocolos como SSL, aunque esta no es la solución más recomendada. Por ejemplo, el tráfico FTP en una red inalámbrica abierta se encuentra sin cifrar, donde se incluyen la información de acceso. En este caso no es necesario utilizar envenenamiento ARP o de caché DNS para capturar los paquetes. Cualquier tarjeta inalámbrica en modo monitor puede analizar el tráfico sin cifrar (Weidman, 2014).

1.4.2 Extensiones en la red no autorizadas

Puntos de acceso no autorizados (del inglés *Rogue Access Point* o *RAP*): Es un punto de acceso que se une a la red sin autorización o que personifica a un dispositivo conectado (ver figura 1.5). Los puntos de acceso no autorizados se pueden encontrar en varias formas (Bosworth et al., 2014):

- **Interno no malicioso:** Este tipo de punto de acceso no autorizado es comúnmente implementado por un empleado o funcionario sin malas intenciones, pero utilizado por un atacante externo.
- **Interno malicioso:** Un atacante puede hacerse pasar como un AP legítimo de una empresa utilizando el mismo SSID que los puntos de acceso autorizados.
- **Externo malicioso:** Un AP que usa tácticas similares a un punto de acceso interno para obligar a los clientes desprevenidos para conectarse a este. La principal diferencia con el interno es la cercanía física. Por lo general en las configuraciones en el cliente se exige la conexión al AP que posea la señal más potente y que coincida con la deseada SSID y requerimientos de autenticación.

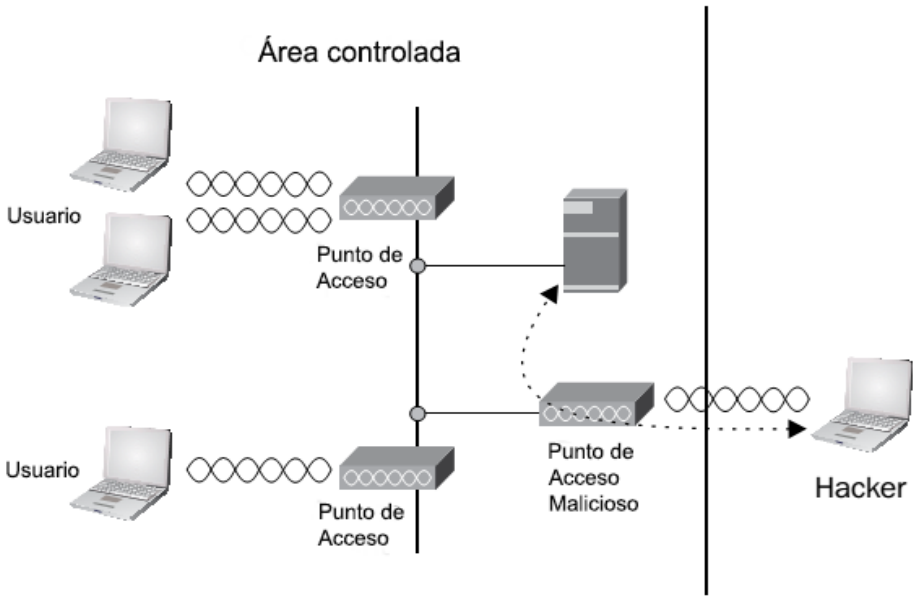

Figura 1.5. Punto de acceso malicioso.

Los RAP permiten la existencia de varios escenarios para el hacker o atacante; se pueden mencionar:

Escucha de tráfico: Cualquier *sniffer* de redes como *Wireshark* o *Ettercap* puede interceptar el tráfico.

Phishing: Al conectarse un usuario, puede forzarse al uso de un servidor DNS no legítimo.

Puerta trasera (del inglés *backdoor*): Mediante la implementación de un dispositivo como un *RaspberryPi* que utiliza la red cableada, puede establecerse una red inalámbrica no legítima que brinda servicio.

1.4.3 Vulnerabilidades en la capa física

La capa física es la capa de menor jerarquía del modelo OSI. Dicha capa define las interfaces mecánica, eléctrica y de temporización de la red. Mediante la variación de algunas propiedades físicas, como el voltaje o la corriente, es posible transmitir información (Kurose y Ross, 2013).

Las redes Wi-Fi utilizan el espacio radioeléctrico como medio de trasmisión. El espectro que emplean estas redes no requiere de una licencia de uso, aunque la IEEE ha creado estándares tales como 802.11d, 802.11h, 802.11j y 802.11y, los cuales especifican aspectos particulares de la operación de las redes Wi-Fi en Europa, Japón, EE.UU y otros dominios regulatorios (Álvarez-Campana et al., 2009, Holt y Huang, 2010). La ausencia de un medio privado (físico) permite los ataques a la capa física de la señal inalámbrica, independientemente de los controles de seguridad en niveles superiores. Esta es la principal amenaza en cualquier forma de comunicación inalámbrica, no necesariamente de 802.11 (Bosworth et al., 2014).

A pesar de lo anterior estas redes presentan una serie de riesgos de seguridad, entre los cuales se encuentran:

Interferencia: Los estándares IEEE 802.11 operan sobre las bandas de frecuencias 2.4 Ghz y 5 Ghz. En estas bandas se encuentran las frecuencias de trabajo de otros dispositivos inalámbricos (teléfonos inalámbricos, trasmisores de video, microondas, otro punto de acceso de mayor potencia, entre otros) u otras tecnologías inalámbricas como *Bluetooth*, los cuales al encontrarse en el área de cobertura pueden interferir la trasmisión de la señal. Por ello, estas bandas son consideradas por algunos autores como “bandas basuras” (Carballeiro, 2013) por encontrarse en esta gama de frecuencia. Otras fuentes de interferencia en estas redes, aunque de menor repercusión, son la topología del terreno y las características arquitectónicas de las edificaciones. Dentro de esta última influye la estructura y los materiales empleados en las edificaciones, los cuales pueden atenuar la señal (K.Sandhu, 2013). El espectro en la frecuencia 2.4 GHz solo provee 3 canales sin solapar. Este se

encuentra en el rango de 2.412 GHz a 2.462 GHz (Tanenbaum, 2011). Un AP cercano puede causar una situación de negación de servicio mediante el uso de un canal solapado.

***Hardware* al descubierto:** Los dispositivos que conforman una red Wi-Fi de una organización o empresa y que se encuentren en exteriores, tienen un mayor riesgo de ser dañados físicamente por personas malintencionadas. De esta manera puede facilitar el descubrimiento de la topología de la red por atacantes en potencia. Lo anterior es un aspecto muy importante que se debe tener en cuenta, pues el uso de AP en exteriores es una práctica común en campus de universidades, parques y demás áreas de acceso público (Gast, 2005, Pietrosemoli et al., 2013).

Las siguientes secciones muestran diferentes tipos de métodos y técnicas de ataques inalámbricos.

1.4.4 Ataques al control de acceso

Los ataques al control de acceso consisten en la penetración a la red mediante la evasión de las medidas de control de acceso como los filtros MAC en los AP y controles en los puertos de acceso 802.11 (Johns, 2015).

1.4.4.1 *War Driving*

War driving es la acción de descubrimiento o reconocimiento de las redes inalámbricas mediante la escucha a las tramas balizas o a través del envío de peticiones para proveer un punto de partida para futuros ataques. *War driving* es usualmente realizado por dos individuos, uno que maneja el automóvil y el otro escanea y descubre las redes inalámbricas en el área. Este término no significa que el usuario atacante necesariamente se encuentre conduciendo, sino que puede estar en libre movimiento (ejemplo: caminando), alrededor de un parque tecnológico, por el centro de una ciudad, dentro de una edificación, entre otros lugares, con una computadora portátil o un Asistente Personal Digital (del inglés *Personal Digital Assistant*, PDA). Con las correctas herramientas y aplicaciones, se puede establecer configuraciones GPS para apuntar a estas locaciones Wi-Fi y guardarlas para posteriores ataques (Fleck, 2002, Johns, 2015).

1.4.4.2 Falsificación o *spoofing* de dirección MAC e IP

El estándar IEEE 802.11, como es conocido es una extensión del estándar IEEE 802. Este utiliza el mismo esquema de direccionamiento de 48 bits como las LANs pertenecientes al estándar 802.3. Sin embargo, existen varias diferencias en la subcapa Medio de Control de Acceso (del inglés *Media Access Control*, MAC) y en la capa física, con respecto al estándar anteriormente citado. Existen vulnerabilidades inherentes al formato y uso de las tramas MAC, en dependencia del tipo de trama que se analice, por lo que puede ser posible el cambio o falsificación de las direcciones en las tramas de datos. En otras palabras la falsificación de MAC es un término utilizado cuando el atacante reconfigura su propia dirección MAC con el objetivo de personificar un cliente o AP (Johns, 2015). Estas vulnerabilidades constituyen una de las bases de las amenazas a las cuales están sometidas las WLANs. La capa de red es la encargada de llevar los paquetes desde el origen hasta el destino, la cual puede requerir muchos saltos en dependencia de la cantidad de enrutadores (del inglés *routers*) intermedios. Esta función contrasta con la que ocupa la capa de enlace de datos, que solo tiene como objetivo mover las tramas de la misma subred. Por lo tanto, la capa de red es la capa más baja que maneja la transmisión de extremo a extremo (Tanenbaum, 2011). Una de las vulnerabilidades presentes en esta capa es la técnica de falsificación de dirección IP (del inglés IP *Spoof*), la cual puede hacer uso de una dirección IP existente en la red o de una nueva (Corletti, 2011), (ver figura 1.6).

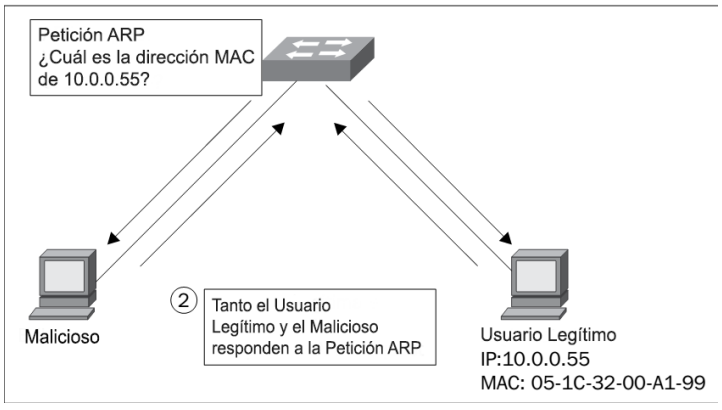

Figura 1.6. Falsificación de MAC.

1.4.5 Ataques a la confidencialidad de las redes Wi-Fi

Estos ataques intentan interceptar información privada enviada a través de redes inalámbricas, como la intercepción de texto sin cifrar, cifrado por 802.11 o por protocolos superiores.

1.4.5.1 Espionaje

Espionaje en seguridad informática significa la acción de captura y decodificación de datos, para luego obtener información potencialmente sensible. Es exactamente como espiar una llamada telefónica. Usted escucha, graba, y luego obtiene información sensible de la conversación. Herramientas como *Ettercap, Kismet* y *Wireshark* pueden utilizarse para este tipo de ataque (Johns, 2015).

1.4.5.2 Punto de acceso gemelo malicioso

Un AP gemelo malicioso (del inglés *Evil Twin AP*) es similar al punto de acceso malicioso. El atacante crea un punto de acceso inalámbrico falso con el mismo SSID de un AP de confianza de la red para engañar a los usuarios. Amplifican su señal de manera que el cliente automáticamente se conecte a ellos. Herramientas como *Honeypot, CqureAP, D-Link G200, HermesAP, Rogue Squadron,* y *WifiBSD* pueden realizar estos ataques (Song, 2010).

1.4.5.3 AP *Phishing*

Phishing es un término informático que denomina un tipo de abuso informático y que se comete mediante el uso de un tipo de ingeniería social caracterizado por intentar adquirir información confidencial de forma fraudulenta. En este caso particular el atacante ejecuta un portal web o un servidor web falso en un AP falso para realizar el *phishing* de inicios de sesión, cuentas bancarias y números de tarjetas de crédito (Martinovic, 2007). Este es uno de los ataques más peligrosos debido a que el usuario medio, no nota que se encuentra bajo un ataque. Los usuarios creen que el sitio web es legítimo cuando en realidad el atacante sólo espera que inicien sesión para captar su información en el otro extremo. Herramientas como *airpwn, Airsnarf, Hotspotter, Karma, Wifiphisher* y *RGlueAP* pueden realizar estos ataques.

1.4.5.4 Ataque hombre en el medio

Un ataque hombre en el medio (del inglés *man in the middle*) consiste en la intercepción del tráfico de red por un atacante. El atacante podría utilizar este ataque en una red cableada o

inalámbrica para obtener los nombres de usuario, contraseñas, ver correos electrónicos, sesiones de sitios web, u otros. Los ataques MITM son probablemente uno de los ataques más potentes sobre un sistema WLAN. Existen diferentes configuraciones que pueden utilizarse para llevar a cabo el ataque. El más común es cuando un atacante se conecta a internet mediante una LAN cableada y crea un punto de acceso malicioso. Este punto de acceso transmite un SSID similar al del punto de acceso local de las inmediaciones. Un usuario puede conectarse accidentalmente a este punto de acceso falso y puede creer que se encuentra conectado al punto de acceso legítimo (Ramachandran, 2011). Herramientas como *dsniff, Ettercap-NG*, y *sshmitm* pueden realizar estos ataques. Este ataque se ilustra en la figura 1.7.

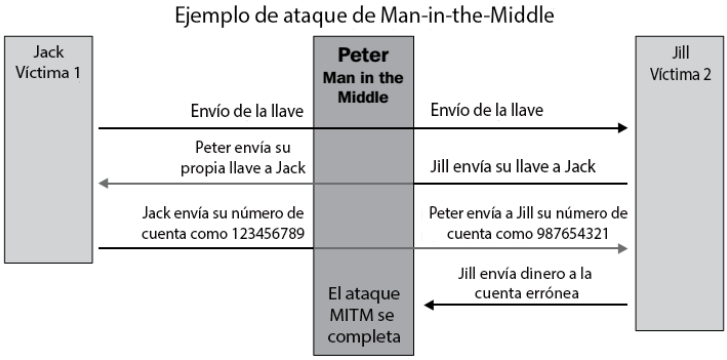

Figura 1.7. Ataque hombre en el medio.

1.4.5.5 Ataque de personificación de servidor *Radius*.

Las soluciones inalámbricas, que no utilicen RSN, configuraciones de confianza con específicos tipos de AP, y además no empleen autenticación basada en certificados mutuos, son vulnerables a la combinación de AP no autorizado y personificación de servidor *Radius*.

Este ataque recae en la habilidad del atacante de forzar al cliente a la conexión a un AP malicioso con un método de autenticación EAP. En una solución típica RSN EAP, en respuesta a una petición de autenticación, el servidor *Radius* responde con un desafío *Radius* de acceso, que es reenviado por el AP al cliente. En este ataque, el AP malicioso es configurado para usar un servidor *Radius* brindado por el atacante. El servidor *Radius* porta el proceso de la autenticación EAP y obliga al cliente a proveer el *hash* de su contraseña en respuesta a un desafío fijo y conocido. Dependiendo del tipo de implementación EAP en uso, el nombre usuario asociado con el cliente se podrá observar en texto plano dentro la *EAP-*

Identity-Response enviado en este proceso. Conocido que el AP no autorizado usa un desafío fijo, se puede realizar un ataque de fuerza bruta mediante el uso de ataques de diccionario (Bosworth et al., 2014). Este ataque solo se aplica a implementaciones EAP, que no utilicen certificados en el cliente para su autenticación, como EAP-PEAP con MSCHAP-v2 usado en la red UCLV-WIFI.

1.4.6 Ataques a la autenticación de las redes Wi-Fi

Los atacantes utilizan los ataques de autenticación para el robo de identidades y credenciales de usuarios legítimos para acceder a las redes privadas y sus servicios.

1.4.6.1 Predicción de clave compartida (del inglés *shared key guessing*)

El atacante intenta adivinar la clave compartida 802.11 para la autenticación mediante el uso de credenciales por defecto del proveedor o generadores de claves compartidas. Las claves compartidas no deben dejarse con el valor por defecto y debe ser cambiado inmediatamente después de la instalación y configuración del dispositivo. Cualquier herramienta de craqueo como *aircrack-ng* puede realizar este ataque (Johns, 2015).

1.4.6.2 Ataque a la encriptación WEP

Debido a las vulnerabilidades presentes en el protocolo WEP (ver el tópico de Mecanismos de seguridad de la capa de enlace), la encriptación WEP, con el *software* y *hardware* disponible en la actualidad puede ser craqueada en un breve intervalo de tiempo. La encriptación WEP solo debe ser usada en casos donde el *hardware* se encuentre obsoleto. Herramientas como *aircrack-ng, airSnort, airoway, chopchop,* y *dwepcrack* pueden vulnerar la encriptación WEP.

1.4.6.3 Craqueo de WPA/WPA2 PSK

WPA/WPA2 PSK es un subconjunto del estándar IEEE 802.11 WPA/WPA2 que no ejerce la compleja tarea de la distribución de llaves y la autenticación del cliente mediante la asignación de la misma clave compartida a cada suplicante involucrado. Esta clave maestra es derivada de una contraseña, pre-configurada por el administrador de la red. Cuando un dispositivo crea una conexión con el punto de acceso, una nueva clave de sesión se deriva de la clave maestra para cifrar y autenticar el tráfico entre suplicante y punto de acceso. Esta "facilidad" de utilizar una llave maestra en lugar claves únicas por usuario, facilita el

despliegue de las redes WPA/WPA2 en el hogar y oficina. La utilización de esta configuración es un riesgo conocido, pues convierte a este protocolo vulnerable a los ataques de fuerza bruta en su fase de negociación de claves; que permite revelar en última instancia, la contraseña que protege la red. Esta vulnerabilidad debe ser considerada desastrosa tanto el protocolo permita que la derivación de claves sea pre-calculada, haciendo simple, los ataques de fuerza bruta para el atacante.

El ataque de craqueo PSK recupera el WPA/WPA2 PSK de una trama *key handshake* mediante el uso de una herramienta de ataque de diccionario, sea en forma de lista de palabras o *rainbow tables*. La diferencia entre una lista de palabras y una *rainbow table* está dada en que la última consiste en una lista pre-computada de *hashes*. Otra alternativa para la obtención de las contraseñas PSK es el ataque por fuerza bruta, el cual es menos práctico en término del tiempo necesario para aplicarlo.

Este ataque depende realmente de la fortaleza de la encriptación de la clave WPA/WPA2. Si la clave es muy fuerte, podría tomar semanas romperla. A medida que la contraseña sea más larga, es menos probable convertirse en un objetivo. Herramientas tales como *coWPAtty, pyrit, genpmk, KisMAC* y *wpa_crack* pueden realizar estos ataques (Bosworth et al., 2014, Johns, 2015, Ramachandran, 2011).

1.4.6.4 WPA/WPA2: Wi-Fi Protected Setup (WPS)

Wi-Fi Protected Setup permite a los usuarios con poco conocimiento de configuraciones de seguridad inalámbrica facilitar la configuración de una clave pre-compartida sin la necesidad de autentificarse en los enrutadores. WPS describe el protocolo y número de identificación personal (PIN) usado para negociar una clave pre-compartida WPA/WPA2 sin la necesidad de una configuración manual en el cliente y AP. WPS solo existe en *hardware* a nivel de pequeñas infraestructuras, no a nivel empresarial (Weidman, 2014).

WPS contempla cuatro tipos de configuraciones diferentes para el intercambio de credenciales, Número de Identificación Personal (del inglés *Personal Identification Number,* PIN), PBC (del inglés *Push Button Configuration*), NFC (del inglés *Near Field Communications*) y Bus Serial Universal (del inglés *Universal Serial Bus,* USB), (Kuo, 2007):

- PIN: Tiene que existir un PIN asignado a cada elemento que vaya a asociarse a la red. Este PIN tiene que ser conocido tanto por el enrutador (registrador o del inglés *registrar)*, como por el usuario (registrado o del inglés registrado). Es necesaria la existencia de una interfaz para que el usuario pueda introducir el mencionado PIN.
- PBC: La generación y el intercambio de credenciales son desencadenados a partir que el usuario presiona un botón (físico o virtual) en el AP (o en otro elemento registrador) y otro en el dispositivo. Notar que en el corto lapso de tiempo entre que se presiona el botón en el AP y se presiona en el dispositivo, cualquier otra estación próxima puede ganar acceso a la red.
- NFC: Intercambio de credenciales a través de comunicación NFC. La tecnología NFC, basada en RFID (del inglés *Radio Frequency IDentification*) permite la comunicación sin hilos entre dispositivos próximos (0-20 cm). Entonces, el dispositivo registrado se tiene que situar al lado del registrador para desencadenar la autenticación. De esta manera, cualquier usuario que tenga acceso físico al registrador, puede obtener credenciales válidas.
- USB: Con este método, las credenciales se transfieren mediante un dispositivo de memoria flash desde el registrador al registrado.

Los métodos PBC, NFC y USB pueden usarse para configurar dispositivos sin pantalla ni teclado (ej. impresoras, cámaras, etc.), pero aunque el estándar contempla NFC y USB, todavía no se certifican estos mecanismos. Actualmente sólo el método PIN es obligatorio en todas las estaciones para obtener la certificación WPS; PBC es obligatorio sólo en APs.

El PIN puede ser requerido por el AP en el agente WPS del cliente o este pudiera proveer el PIN para introducirlo en el *prompt* WPS del AP. La autenticación WPS está basada en 802.1X EAP con implementación específica de EAP en dependencia del fabricante, el cual usa intercambio de llaves *Diffie-Hellman* para probar la posesión de la primera mitad del PIN entre el AP y el cliente, y eventualmente la segunda. Como WPS divide el PIN en dos pasos separados de autenticación, limita severamente el número de combinaciones para cada mitad del PIN. Un atacante puede derivar la primera y la segunda parte del PIN, mediante la utilización de los mensajes EAP, particularmente “EAP-NACK”. Este mensaje es brindado por el AP al fallar la introducción de los correctos valores de la primera y segunda mitad del

PIN. Si se realiza un ataque de fuerza bruta aplicada en cada porción separada del PIN se requiere solo 10^4 valores lo sumo, por lo tanto quedan $10^4+10^4=20000$ posibles combinaciones en vez del valor del pin original de 8 dígitos, 10^8 combinaciones. En promedio, las herramientas de craqueo WPS pueden recuperar una clave WPA/WPA2 en texto plano en un intervalo de 4 a 10 horas (Weidman, 2014).

1.4.6.5 Ataque a MS-CHAPv2

Algunas implementaciones EAP MS-CHAPv2 son conocidas como vulnerables a los ataques de diccionario como el uso de EAP-LEAP o EAP-PEAP-MSCHAP con malas configuraciones suplicante. La manera tradicional de obtención de las credenciales de la víctima, en este tipo de configuración, es a través del uso de un programa que procesa el *hash* desafío-respuesta y una lista de palabras o diccionario. Este es el único enfoque viable para explotar esta vulnerabilidad.

Este proceso se ha acelerado recientemente a través de una nueva comprensión de la funcionalidad del protocolo MS-CHAPv2. Durante MS-CHAPv2, el hash MD4 de la contraseña de un usuario es dividida bit a bit para construir tres claves individuales DES (siete bytes cada uno). Una herramienta típica de pruebas penetración como *Asleep* toma el hash MD4, lo separa en tres claves DES y usa esas claves para cifrar un valor en texto plano presente en un diccionario y luego compararlo con el *hash* del cifrado inicial. En un acercamiento a la metodología de craqueo tradicional se tendría un tamaño de clave de $2^{(56+56+56)}$ o alrededor de 10^{50} lo cual hace que el craqueo por fuerza bruta sea poco viable. Esta es la razón por la que los diccionarios se utilizan como un enfoque de "mejor estimación", porque el ataque de fuerza bruta incremental (carácter por carácter) no es una opción para una cantidad razonable de tiempo. El uso de tres claves DES es primordial para la explotación. Un *hash* MD4 solo posee 16 bytes, mientras que tres claves DES de siete bytes equivale a un total de 21 bytes ¿Cómo se logra componer esta diferencia? El relleno, reduce el tamaño de la clave efectiva de la tercera clave DES a dos bytes. Si cada una de las 2 llaves de 7 bytes y la llave de 2 bytes es craqueada individualmente, el tamaño de llave se reduce a $2^{56}+2^{56}+2^{16}$ o alrededor de 10^{17} (lo que disminuye el tamaño de clave por un factor de 10^{32}). La parte interesante de la rutina de MS-CHAP-v2 es que cada clave DES se utiliza para cifrar el mismo texto plano; por lo tanto, durante el proceso de craqueo se puede utilizar

la misma entrada de clave en cada una de las dos funciones DES, a fin de reducir efectivamente la cantidad de funciones DES para cada iteración a uno, dejando la única operación dual como la comparación de la salida DES en cada uno de los dos textos cifrados previamente determinados. Esto resulta un tamaño de clave de $2^{56}(10^{16})$, el cual en 1998 podía ser craqueado en 4.5 días como promedio, y en los días actuales, con las arquitecturas disponibles en el mercado basadas en el principio de paralelismo, en 12 horas aproximadamente (Bosworth et al., 2014).

1.4.7 Ataques de denegación de servicio (DoS)

El ataque de denegación de servicio en las redes Wi-Fi, consiste en privar al usuario legítimo de la red inalámbrica o causar la indisponibilidad de la red a partir del bloqueo de la misma. Las redes inalámbricas son extremadamente susceptibles a los ataques de denegación de servicio, y es difícil localizar al atacante en una red inalámbrica distribuida (Beggs, 2014).

Ejemplos de ataques de denegación de servicio son (Beggs, 2014, Bellardo, 2003, Gu, 2007):

- Inyección de comandos de red a una red inalámbrica, como los comandos de reconfiguración, puede provocar un fallo en los enrutadores, conmutadores u otros dispositivos de red.
- Algunos dispositivos y aplicaciones pueden reconocer un ataque en proceso y responder de manera automática mediante la desactivación de la red. Un atacante malicioso puede lanzar un ataque obvio y luego dejar que el objetivo cree su propia DoS.
- Bombardeo de la red inalámbrica con una inundación de paquetes de datos puede ocasionar la indisponibilidad de la red; por ejemplo, un ataque de inundación HTTP realizando miles de peticiones de páginas en un servidor web puede agotar su capacidad de procesamiento. De la misma manera, mediante la inundación de la red con paquetes de autenticación y asociación se puede ocasionar el mismo efecto.
- Los atacantes pueden crear comandos específicos de deautenticación y disociación, que se utilizan en las redes inalámbricas para cerrar una conexión autorizada o para inundar la red y evitar que los usuarios legítimos mantengan su conexión a un punto de acceso inalámbrico.

1.5 Estándar de *pentesting*

El estándar de ejecución de pruebas de penetración que se utiliza como modelo consiste en siete secciones principales. Estas abarcan el proceso desde la comunicación inicial y el razonamiento detrás de una prueba de penetración, desde la recopilación de inteligencia hasta la fase de reportes. Los especialistas tratan de obtener una mejor comprensión de la organización evaluada, a través de la investigación de vulnerabilidades, explotación y posterior explotación, donde la experiencia en seguridad de los especialistas en pruebas de penetración se combina con el entendimiento del negocio de la contratación. Finalmente se procede a la presentación de informes, que captura todo el proceso, de manera que tenga sentido para el cliente y ofrezca más valor a la misma. Para más información, este estándar se encuentra en su página web oficial (Nickerson, 2014), actualmente en la versión 1.0 y licenciado por *GNU Free Documentation License* 1.2.

Estas son las secciones que componen el proceso de pruebas de penetración:

1. Interacciones iniciales
2. Recopilación de inteligencia
3. Modelado de las amenazas
4. Análisis de vulnerabilidad
5. Explotación
6. Post-Explotación
7. Reportes

Las interacciones iniciales es la primera sección al realizar una prueba de penetración. En esta sección se define el enfoque general de la situación existente. Se realizan una serie de preguntas que caracterizarán el sistema para luego comenzar las pruebas de penetración. Se establecen para cada escenario existente una serie de interrogantes, sea para un *pentest* en la red cableada, a una red inalámbrica, página web o a la red física. Además se interrogan a los encargados de la empresa y administradores de red. Estas son las preguntas que deben realizarse en una red Wi-Fi.

- ¿Cuántas redes inalámbricas posee la institución?

- Si se utilizan redes inalámbricas:
    - ¿La red inalámbrica requiere autenticación?
    - ¿Qué tipo de encriptación es usada?
    - ¿Cuál es el área de cobertura de la red inalámbrica?
    - ¿Es necesaria la enumeración de puntos accesos maliciosos?
    - ¿Es necesaria la evaluación de los ataques a los clientes?
    - ¿Cuántos clientes aproximadamente utilizarán la red inalámbrica?

Recopilación de inteligencia

La recopilación de inteligencia consiste en reunir la mayor cantidad de información posible para ser utilizada al penetrar el objetivo durante las fases de evaluación de la vulnerabilidad y explotación. Cuanta más información sea capaz de reunirse durante esta fase, más vectores de ataque pueden ser utilizados en el futuro.

Modelado de las amenazas

El estándar no especifica un modelo a seguir pero requiere que el modelo a utilizar debe ser consistente en términos de representación de las amenazas, sus capacidades, sus calificaciones cuando la organización ha sido evaluada y la habilidad de ser aplicado repetidamente en futuras pruebas con los mismos resultados. El estándar se enfoca en dos elementos clave del modelado tradicional de amenazas: los activos y atacantes (amenazas). La idea es simular una situación de amenaza en la que un atacante toma el control de los activos de la red.

Análisis de vulnerabilidad

El análisis de vulnerabilidad es el proceso de descubrimiento de fallas en los sistemas y aplicaciones que pueden ser aprovechados por un atacante. Estos defectos pueden oscilar entre el cliente y una mala configuración de servicios, o el diseño de aplicaciones inseguras. Aunque el proceso usado para buscar las vulnerabilidades varía y depende en gran medida del componente particular que está siendo probado, algunos principios fundamentales se aplican al proceso. Al realizar el análisis de la vulnerabilidad el especialista en pruebas de

penetración debe enfocar correctamente el alcance de las pruebas y la amplitud de aplicación para cumplir los objetivos o requisitos de los resultados deseados.

Explotación

La fase de explotación de una prueba de penetración se centra en el establecimiento del acceso a un sistema o recurso, burlando las restricciones de seguridad.

Si el análisis de vulnerabilidad se ha realizado correctamente, se debe garantizar la precisión y buena planificación de la fase de explotación. El objetivo principal es identificar el punto de entrada principal a la organización e identificar los activos de alto valor. Si la fase de análisis de vulnerabilidad fue debidamente realizada, se debe obtener una lista de objetivos de alto valor. En última instancia, el vector de ataque debe tener en cuenta la probabilidad de éxito y el mayor impacto en la organización.

Post-explotación

El propósito de la fase de post-explotación es determinar el valor del activo comprometido y mantener el control de este para su uso posterior. El valor de la activo está determinado por la sensibilidad de los datos almacenados en él y los activos presentes en la red para futura explotación. Los métodos descritos en esta fase tienen el propósito de ayudar al especialista en pruebas de penetración a identificar y documentar los datos sensibles, identificar los valores de configuración, los canales de comunicación y las relaciones con otros dispositivos de red que se pueden utilizar para ganar más el acceso a la red, y la configuración de uno o más métodos de acceder a la máquina en un momento posterior.

Reportes

El reporte se encuentra dividido en dos secciones principales con el propósito de comunicar los objetivos, métodos y resultados de las pruebas de penetración.

La primera sección de los reportes es el resumen ejecutivo donde se comunica al lector los objetivos específicos de la prueba de penetración y los resultados de la misma. En esta debe encontrarse los antecedentes, donde se explica los propósitos de la prueba de penetración. Deben estar presentes los términos identificados en la primera sección de la prueba relacionada a los riesgos, medidas y los objetivos ya que estos conducen al cliente a los objetivos y a los resultados relativos. El riesgo de las amenazas será identificado en el área

de Ranking de Riesgos. El riesgo se define por un sistema de puntuación tal como se muestra en la figura 1.8.

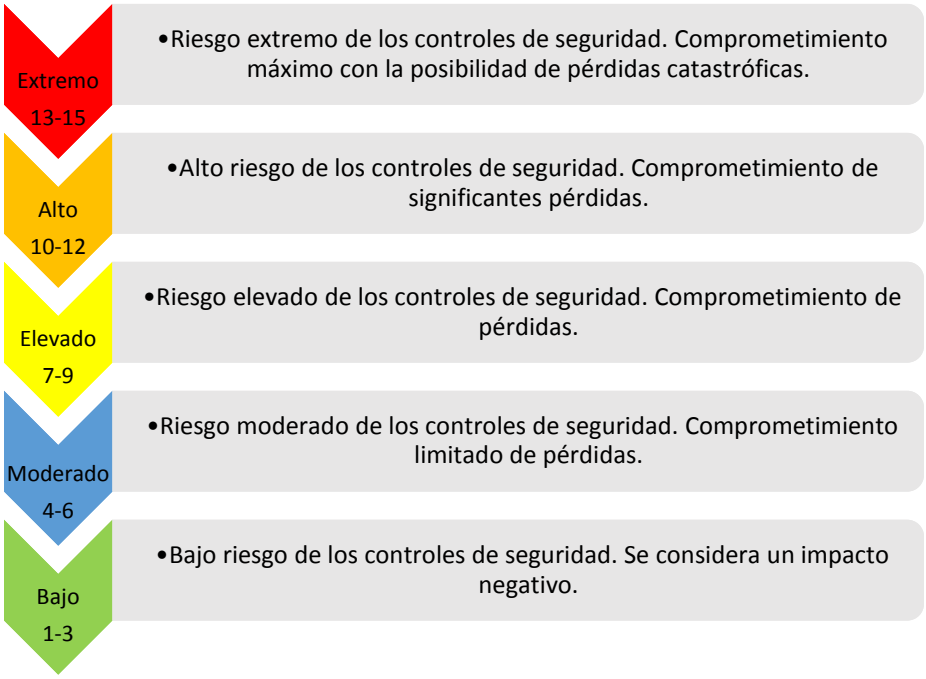

Figura 1.8. Escala de riesgo de seguridad de la información

El estándar cuenta con una sección de vulnerabilidades encontradas donde se definen a partir de gráficos e información, cada una de las fallas explotadas, además cuenta con una sección de recomendaciones donde se le informa al usuario las tareas necesarias para resolver los riesgos identificados. El especialista debe proveer una estrategia a seguir por el cliente en un período a largo y a corto plazo donde se especifica un plan de contingencia a seguir. Por último se brinda un reporte técnico donde se comunica al lector los detalles técnicos de las pruebas de penetración. Este reporte plasma en detalle los objetivos, información, los tipos de ataques utilizados, el impacto y las medidas sugeridas por el especialista en pruebas de penetración. Esta sección es una de las más relevantes en el proceso de *pentesting* pues se entrega los resultados finales de las pruebas al cliente.

1.6 Conclusiones parciales

Los beneficios de las redes WLAN son innegables e indispensables en la implementación de una solución de redes cualesquiera en los días actuales. Su versatilidad y facilidad de despliegue le permiten desempeñar un papel importante como complemento de la red

cableada. La seguridad toma un papel rector en este tipo de redes, y su presencia se evidencia en los diversos mecanismos de seguridad que habitan en las diferentes capas del modelo OSI. Según el nivel de seguridad requerido, el servicio deseado y el coste de gestión se establecen estos mecanismos de seguridad. A pesar de los esfuerzos de los especialistas en redes inalámbricas cada una de las capas del modelo OSI presenta vulnerabilidades, que son originadas por configuraciones inseguras o por vulnerabilidades inherentes al protocolo utilizado en particular. De estas fisuras nacen diversos ataques que son llevados a cabo por un especialista en pruebas de penetración con fines correctivos o simplemente por un usuario con objetivo de comprometer la red inalámbrica. El *pentesting* en redes inalámbricas es una tarea bien compleja llevada a cabo solo por expertos en seguridad. El proceso se encuentra estandarizado para su uso correcto y claridad en el trato con los clientes que solicitan el servicio en su empresa. Se concluye que debe existir una correlación entre las vulnerabilidades presentes en las redes WLAN y los mecanismos de seguridad a utilizar, siendo una labor compleja debido a las diversas situaciones que se pueden presentar en una infraestructura inalámbrica, sea en modo infraestructura o *ad-hoc*.

CAPÍTULO II. Herramientas empleadas en las pruebas de penetración de redes Wi-Fi

El uso de herramientas de auditoría y explotación de vulnerabilidades informáticas para garantizar los elevados requerimientos de seguridad de las redes Wi-Fi es de vital importancia. El estudio de las características de estas herramientas permite el uso eficiente y adecuado de las mismas. En la actualidad este tipo de herramientas vienen instaladas y pre-configuradas en varias distribuciones *Linux* diseñadas para la Seguridad Informática, lo cual permite ahorrar tiempo y esfuerzo a los especialistas en pruebas de penetración u otro personal interesado, como administradores de red, en la realización de las auditorías de seguridad.

2.1 Distribuciones *Linux* vinculadas a la Seguridad Informática

Linux es un núcleo libre de sistema operativo basado en Unix. Es uno de los principales ejemplos de *software* libre y de código abierto. Se encuentra licenciado bajo GPLv2 y se encuentra desarrollado por colaboradores de todo el mundo. El desarrollo del día a día tiene lugar en la *Linux Kernel Mailing List Archive*. El núcleo *Linux* fue concebido por el entonces estudiante de ciencias de la computación finlandés Linus Torvalds en 1991. Normalmente *Linux* se utiliza junto a un empaquetado de *software*, llamado distribución *Linux*. Una distribución *Linux* (coloquialmente llamada distro) es una distribución de *software* basada en el núcleo *Linux* que incluye determinados paquetes de *software* para satisfacer las necesidades de un grupo específico de usuarios, dando así origen a ediciones domésticas, empresariales y para servidores. Por lo general están compuestas, total o mayoritariamente, de *software* libre, aunque a menudo incorporan aplicaciones o controladores propietarios.

Las distribuciones de seguridad están principalmente diseñadas para realizar tareas de seguridad en la red como auditar la seguridad y hacer pruebas de penetración con los fines de prevenir y monitorear accesos no autorizados, abusos, alteraciones o denegación de servicios de red. La mayoría de ellas están disponibles como *Live CD*, por lo que es posible su prueba y uso, sin alterar nuestro sistema instalado. A continuación se describe brevemente las distribuciones de seguridad que se encuentran activas:

***BackTrack*:** Basada en *Ubuntu*, es considerada como una de las distribuciones más populares entre los *hackers* y entusiastas de la seguridad de redes. Fue creada combinando dos distros principales: *Auditor Security Linux* (basada en *Knoppix*) y WHAX (anteriormente *Whoppix,* basada en *Slax*).

DEFT (Set de Herramientas Forenses y de Evidencia Digital o en inglés *Digital Evidence & Forensic Toolkit*): Es una distribución personalizada de la distribución *Kubuntu Live Linux*. Es un sistema de uso muy sencillo que incluye excelente detección de *hardware* y de las mejores aplicaciones de código abierto dedicadas a respuesta inmediata a incidentes e informática forense. *Helix* es otra distribución similar a DEFT la única diferencia radica que esta es basada en *Knoppix Live Linux*.

***Network Security Toolkit* (NST):** Basada en *Fedora*, es un Live CD equipado con herramientas de análisis de seguridad de redes, programas de validación y monitoreo que puede ser utilizado en servidores virtuales que albergan máquinas virtuales. Su principal objetivo es proveer a los administradores de red de un set completo de herramientas de seguridad de código abierto.

Ophcrack: Basada en SLAX6, es un *Live CD* para la creación de tablas arcoíris de contraseñas alfanuméricas. El *Live CD* craquea contraseñas automáticamente, sin necesidad de instalación, ni credenciales de administración, entre otras funcionalidades.

***OSWA Assistant*:** Distribución dedicada a la auditoría de redes inalámbricas.

***Samurai*:** Es un *Live CD* enfocado a las pruebas de penetración en aplicaciones web.

WiFiSlax: Distribución GNU/*Linux* con funcionalidades de *Live* CD y *Live* USB, diseñada para la auditoría de seguridad y relacionada con la seguridad informática en general. Incluye una larga lista de herramientas de seguridad y auditoría listas para ser utilizadas, entre las que destacan numerosos escáneres de puertos y vulnerabilidades, herramientas para creación y diseño de *exploits, sniffers*, herramientas de análisis forense y herramientas para la auditoría inalámbrica, además de añadir una serie de útiles lanzadores.

Tails: Es un sistema operativo basado en *Debian* que utiliza *Tor* en todo su tráfico de internet. Su objetivo principal es brindar seguridad a través del anonimato. Con este, es posible la navegación de manera anónima a través de conexiones encriptadas.

***Qubes*:** Es un sistema operativo con ambiente de escritorio basado en *Fedora* que garantiza la seguridad a través de aislamiento. *Qubes* asume la inexistencia de un sistema verdaderamente seguro, por lo que en vez de ejecutar todos los procesos dentro del mismo sistema, los ejecuta dentro de máquinas virtuales. Al ser víctima de un ataque malicioso se garantiza que la agresión no se extienda por todo el sistema operativo.

Existen múltiples sistemas operativos dedicados a la seguridad cuyo uso es similar, se puede mencionar *Chaox-Ng, GnackTrack, Matriux, Katana, NodeZero, BackBox Linux, BlackUbuntu, WeakerTh4n, Caine, Bugtraq, BlackArch,* entre otros.

2.1.1 Distribución *Kali Linux*

Al referirse a las pruebas de penetración, *Kali Linux* es el sistema operativo preferido por los profesionales de la seguridad informática (ver figura 2.1). *Kali* es un sistema operativo avanzado basado en *Linux* que contiene una colección de herramientas de código abierto para realizar diversas tareas de pruebas de seguridad, informática forense y auditorías de seguridad. *Kali Linux* se distribuye en imágenes ISO compiladas para diferentes arquitecturas (32/64 bits) que puede ser usado como un *live* o como un sistema operativo independiente (Singh, 2013).

Figura 2.1. Distribución de seguridad *Kali Linux*.

Fue fundada y es mantenida por *Offensive Security Ltd*. Mati Aharoni y Devon Kearns, ambos pertenecientes al equipo de *Offensive Security*, desarrollaron la distribución a partir de la reescritura de *BackTrack,* por lo que se puede denominar como una completa reconstrucción de *BackTrack Linux*, adherido completamente a los estándares de desarrollo de *Debian Linux*. La infraestructura de *Kali* está organizada adecuadamente, las herramientas fueron revisadas y embaladas, se modificó el *Git* para un VCS (del inglés *Video Computer System*) propio. *Kali* reúne las siguientes características:

- Tiene más de 300 herramientas de pruebas de penetración.
- El modo de adquisición es gratis.
- *Git* (árbol de código abierto).
- Cumple con los requerimientos del Estándar de Jerarquía del Sistema de Archivos (del inglés *Filesystem Hierarchy Standard*, FHS).
- Amplio soporte a dispositivos inalámbricos.
- *Kernel* personalizado con parches de inyección.
- Tiene un entorno de desarrollo seguro.
- Los paquetes están firmados por Privacidad Bastante Buena (del inglés *Pretty Good Privacy*, PGP) y por los repositorios.
- Multi-lenguaje. Soporta ambientes de desarrollo en C, *Python*, y *Ruby*
- Totalmente personalizable.
- Soporte ARMEL y ARMHF (sistemas basados en *Advanced RISC Machines*, ARM).

La última versión disponible *Kali Linux* 1.1.0 se encuentra en su página oficial (Linux, 2015) con un tamaño de 3 Gb aproximadamente. Esta nueva versión ofrece:

- *Kernel* 3.18 parcheado para ataques de inyección inalámbrica.
- Soporte mejorado para *drivers* inalámbricos, debido a las actualizaciones en el *kernel* y en el *firmware*.
- Soporte para NVIDIA *Optimus hardware.*
- Actualizada la herramienta *virtualbox*, *openvm* y paquetes de *vmware-tools.*
- Correcciones y actualizaciones en el *changelog* y cambios en las imágenes del *grub* y fondos de pantalla.

Por su versatilidad, amplio soporte y preferencia entre los especialistas en pruebas en penetración se escoge *Kali Linux* como la distribución ideal para realizar el proyecto de tesis.

2.2 Herramientas para las pruebas de penetración en redes Wi-Fi

Para el desarrollo de pruebas de penetración en redes Wi-Fi se utilizan diversas herramientas para los análisis de seguridad, como herramientas de descubrimiento o de monitoreo, de forma automatizada o manual, confeccionada por el especialista o pre-configuradas para detectar las vulnerabilidades presentes.

2.2.1 Herramientas para descubrimiento

Las herramientas de descubrimiento son utilizadas con el objetivo de obtener huellas para detectar e investigar redes inalámbricas y APs. El tipo de ataque que realizan estas herramientas por lo general es considerado de carácter pasivo y conocido *war driving*. Aunque son consideradas herramientas de ataque pasivo, estos datos pueden ser utilizados por otras herramientas para provocar posteriormente los ataque activo con diferentes propósitos en una red (Mcclure et al., 2012). A continuación se describen algunas herramientas de descubrimiento.

***Airodump-ng*:** Es usado para la captura de paquetes inalámbricos 802.11 y para la acumulación de vectores de inicialización, con el fin de usarlos con *aircrack-ng* u otra utilidad. Con un receptor GPS conectado al ordenador, *airodump-ng* es capaz de mostrar las coordenadas de los puntos de acceso identificados en su exploración (Aircrack-ng, 2009d). *Airodump-ng* es una buena alternativa para el descubrimiento de redes inalámbricas, su velocidad en la captura de tramas lo hace favorito en comparación con *kismet*, puesto que este ejecuta un menor número de funcionalidades (Mcclure et al., 2012). El método de ejecución de *airodump-ng* es mediante una tarjeta inalámbrica, configurada en "modo monitor" con la utilización de *airmon-ng* (Aircrack-ng, 2009d). Esta herramienta permite analizar el tráfico inalámbrico e inyectar tramas en el aire, para descubrir todos los APs y clientes inalámbricos en el espectro de 2.4 Ghz mediante saltos a través de canales y la observación de datos obtenidos (Aircrack-ng, 2009d, Mcclure et al., 2012).

***Kismet*:** Es una de las herramientas de descubrimiento más poderosas de redes Wi-Fi. Dentro de sus características incluye la detección de redes inalámbricas, *sniffer* y sistema de detección de intrusos. *Kismet* opera con cualquier tarjeta inalámbrica que soporte el modo de

monitor, y puede procesar tráfico 802.11b, 802.11a, 802.11g y 802.11n. Posee una arquitectura basada en *plugins* que permite la decodificación adicional de protocolos no 802.11. Identifica las redes inalámbricas mediante la recopilación pasiva de paquetes y la detección de redes, lo que le permite identificar los nombres de redes ocultas y la presencia de redes que no utilicen tramas balizas de identificación a través de tráfico de datos. Los *plugins* de *Kismet* pueden ejecutarse como cualquier proceso nativo de *Kismet*. Esto incluye el incremento de la capacidad de registro, alertas IDS, la definición de nuevas fuentes de captura (dentro de ciertas limitaciones), y la adición de nuevas características para el *Kismet* UI (Kismet, 2011). *Kismet* utiliza GPS en la identificación de redes inalámbricas, pero como herramienta en general esta ofrece un mayor número de características (Mcclure et al., 2012). Soporta filtrado básico; las redes pueden ser excluidas del seguimiento, del registro en los archivos **.pcap**, o de los registros en general, basado en la BSSID o en la dirección MAC fuente o destino. Proporciona una herramienta para la elaboración de redes, sobre mapas descargados llamados "*gpsmap*". *Gpsmap* analiza los archivos **netxml** y **gpsxml** y "sanitiza" los datos. *GPSMap* puede descargar mapas de varias fuentes en línea (*MapBlast, tigre, Terraserver, Earthamaps*, y más), así como el uso de los gráficos proporcionados por el usuario, siempre y cuando se conozca la escala y el centro de coordenadas. Otra funcionalidad es como sistema de detección de intrusos inalámbrico con el soporte de alertas basadas en trazas y tendencias como tramas balizas inusuales, disociación excesiva, entre otros. *Kismet* se centra en la capa de red 802.11 (capa 2 del modelo TCP/IP), y proporciona la integración con sistemas capa 3 + IDS como *Snort* (Kismet, 2011).

***Airgraph-ng*:** Es una herramienta escrita en *python* que se utiliza para la creación de gráficos de relación entre cliente y AP. Con el fin de ofrecer una visión gráfica de la red Wi-Fi, *airgraph-ng* utiliza la información obtenida de *airodump-ng*. *Airgraph-ng* no se encuentra incorporado en la suite *Aircrack-ng*, se debe descargar por separado. Este depende para su correcta instalación y funcionamiento de:

- *Graphviz* con soporte png
- *Airodump-ng*
- *Python* > 2.7

La sintaxis de uso es la siguiente:

```
# airgraph-ng -i <archivo .csv> -o <salida en formato png> -g [CAPR|CPG]
```

Donde CAPR y CPG significan 2 tipos de gráfico:

CAPR o relación cliente a AP: Muestra los clientes vinculados a un punto de acceso en particular (ver Figura A.1 del Anexo A).

CPR o gráfico de tramas balizas del cliente: Muestra todos los clientes que se encuentran en el envío de tramas balizas de petición para un mismo ESSID (Aircrack-ng, 2010). El código fuente de *airgraph-ng* se puede descargar desde *Github* (Github, 2014).

Para el correcto funcionamiento de *airgraph-ng* es necesario agregar en la carpeta **support**, el archivo **oui.txt**, que contiene un listado de las OUI registradas en la IEEE. Este archivo se puede descargar desde IEEE (IEEE, 2015). En la figura A.2 del Anexo A se muestra un *script* escrito en *bash* para la facilitación del uso de esta herramienta.

2.2.2 Herramientas para monitoreo

Las herramientas de monitoreo están diseñadas para capturar información sensible de la red (por ejemplo, la dirección MAC, dirección IP origen y destino, identificadores de usuario, contraseñas, clave WEP, puertos etc.) como un paso previo a ataques posteriores. Las herramientas de monitoreo son identificadas como s*niffers,* derivado del mencionado *war driving* ejecutado por las herramientas de descubrimiento. Ambos ataques son de carácter pasivo.

***Wireshark*:** Es una herramienta de monitoreo, que se ha convertido en el estándar de mercado en el análisis de protocolos en redes TCP/IP. Se ejecuta en sistemas operativos como *Windows*, *Linux*, OSX, *Solaris*, *FreeBSD, NetBSD*, entre otros. Los datos capturados de la red se pueden consultar a través de una interfaz gráfica de usuario, mediante sus filtros de visualización. Puede analizar y guardar en una larga lista formatos de captura como texto plano (*.txt), *PostScript* (*.ps), *Comma Separated Values* (*.csv), C *Arrays to Packet Bytes* (*.c), PSML o XML *Packet Summary* (*.psml), PDML-XML *Packet Details* (*.pdml). Captura archivos comprimidos con gzip que pueden ser descomprimidos sobre la marcha. A través de *Wireshark* se pueden leer datos en tiempo real en *Ethernet*, IEEE 802.11, PPP con HDLC, ATM, *Bluetooth*, USB, *Token Ring, Frame Relay*, FDDI, entre otros, en dependencia de la plataforma que se utilice. Es útil para analizar múltiples protocolos, como *IPsec*,

ISAKMP, *Kerberos*, SNMPv3, SSL/TLS, WEP, WPA/WPA2 (Wireshark, 2015, Orzach, 2013).

***Tcpdump*:** Es una herramienta estándar de monitoreo de red, es uno de los *sniffers* más populares, basado en el sistema operativo *Unix*. *Tcpdump* realiza la decodificación de información de la trama 802.11 en sus 2 últimas versiones, mediante la instalación de *libpcap* para apoyarlas. Puede usarse para imprimir todas las cabeceras de los paquetes o para ver todas las cabeceras exactas del tráfico de una red, con el objetivo de supervisar y evitar los ataques de desestabilización del sistema (Mcclure et al., 2012).

***Airbase-ng*:** Es una utilidad "multi-propósito" perteneciente a la *suite Aircrack-ng* dirigida a los ataques a clientes conectados a un punto de acceso. Es una herramienta muy versátil que se ejecuta en sistemas operativos de *Linux* y *Windows*. Entre sus funcionalidades más importantes se pueden encontrar (Aircrack-ng, 2009a):

- Implementación del ataque "*Caffe Latte WEP*".
- Implementación del ataque "*Hirte WEP client attack*".
- Captura del *handshake* WPA/WP2.
- Punto de acceso *ad-hoc*
- Punto de acceso normal
- Filtrado por SSID o dirección MAC del cliente
- Manipulación y reenvío de paquetes
- Encriptación de paquetes enviados y desencriptación de los recibidos.

La idea principal de esta utilidad es que los clientes no puedan prever el acceso a un punto de acceso legítimo ante la asociación a un AP falso. Una interfaz (atX) se crea cuando *airbase-ng* se ejecuta. Esta se puede usar para recibir paquetes desencriptados o enviar paquetes encriptados. En muchos casos los clientes envían tramas balizas de petición, para la conexión con un AP legítimo. Estos paquetes son importantes para la conexión de un cliente al punto de acceso malicioso, puesto que el AP responde a cualquier trama baliza de petición con la correspondiente trama baliza de respuesta, el cual le informa al cliente que se encuentra autenticado al BSSID de *airbase-ng*. Esta herramienta puede afectar el

funcionamiento correcto de otros APs que se encuentren en el mismo canal (Aircrack-ng, 2009a).

2.2.3 Herramientas para explotar las vulnerabilidades de autenticación

***Aireplay-ng*:** Es una herramienta perteneciente a la *suite Aircrack-ng*, que se ejecuta en los sistemas operativos de *Linux* y *Windows* (Wirelessdefence.org, 2010b, Wirelessdefence.org, 2010a, Aircrack-ng, 2013). Esta realiza una variedad de ataques de DoS pero fundamentalmente el de autenticación y deautenticación mediante la inyección de paquetes y reinyección de paquetes ARP, entre otros. En conjunto, con las herramientas de la *suite Aircrack-ng* realiza la inyección de paquetes para contribuir al ataque al cifrado en la identificación de claves (Aircrack-ng, 2013, McClure et al., 2012, Ramachandran, 2012, Wireless, 2014, McClure et al., 2009).

Aircrack-ng Suite: *Suite* de *software* de seguridad informática diseñada para trabajar con distribuciones *Linux*, aunque también existe una versión para *Windows* que no es muy estable debido a conflictos con los *drivers*. Es utilizada para realizar el craqueo de los protocolos WEP y WPA/WPA2-PSK. La *suite Aircrack-ng* está compuesta por una gran lista de herramientas para auditorías de redes inalámbricas (las herramientas *airbase-ng, airmon-ng, airgraph-ng, aircrack-ng, aireplay-ng* y *airodump-ng* son pertenecientes a la *suite Aircrack-ng*; no se mencionan pues son tratadas en este capítulo) como (Aircrack-ng, 2009c):

- *Airdecap-ng*: Herramienta para desencriptar archivos de captura WEP/WPA/WPA2.
- *Airdecloak-ng*: Herramienta para remover el WEP *Cloaking*™ de una archivo de captura.
- *Airdrop-ng*: Herramienta para la deautenticación en redes inalámbricas basada en reglas.
- *Airolib-ng*: Precomputa claves WPA/WPA2 en una base de datos para su uso posterior con *aircrack-ng*.
- *Airserv-ng*: Servidor inalámbrico TCP/IP que permite el uso de múltiples aplicaciones por la interfaz inalámbrica.
- *Airtun-ng*: Herramienta para la creación de túneles virtuales.

- *Packetforge-ng*: Herramienta para la creación de paquetes encriptados usados para la inyección de datos.

  Se encuentran otras herramientas experimentales como:

- *Easside-ng*: Herramienta automática que permite la comunicación con un punto de acceso encriptado con clave WEP sin conocer la clave en sí.
- *Tkiptun-ng*: Implementación del ataque a WPA/TKIP mediante la inyección de tramas dentro de una red WPA/TKIP con QoS.
- *Wesside-ng*: Herramienta automática que incorpora varias técnicas para obtener la clave WEP en cuestión de minutos.

Asleap: Es una herramienta de seguridad inalámbrica diseñada para captar y descifrar contraseñas débiles LEAP de *Cisco*. Como LEAP utiliza una variante de MS-CHAPv2 para la autenticación, es susceptible a ataques de diccionario *offline*. *Asleap* permite el ataque al protocolo de Tunelización Punto a Punto (del inglés *Point to Point Tunneling Protocol*, PPTP), como cualquier intercambio MS-CHAPv2 donde se pueda especificar los valores de desafío y respuesta (SUSHI, 2015). Pertenece al índice de herramientas *Linux*. Los rasgos principales que definen a la herramienta *asleap* son (Wright, 2003):

- Analiza el tráfico desde cualquier archivo **.pcap** o cualquier interfaz inalámbrica activa en el modo RFMON con el *libpcap*.
- Monitorea un solo extremo o canal mediante saltos, en busca del funcionamiento de las redes a analizar.
- Realiza ataques de deautenticación para obligar a los usuarios a la reautenticación, y así la captura de contraseñas.
- Realiza deautenticación a un usuario que ha sido anteriormente registrado en la red y a los usuarios que no usen la misma red con cierta regularidad.
- Usa una gran base de datos dinámica y un índice que le permite la consulta rápida en grandes archivos.
- Escribe solo la información de intercambio de LEAP, en un archivo de *libpcap.*

***Bully*:** Es una implementación de ataque de fuerza bruta WPS, escrito en C. *Bully* es conceptualmente idéntico a otros programas, en que se explota el fallo en el diseño de la

especificación WPS. Tiene varias ventajas sobre el código original de *reaver*. Estos incluyen un menor número de dependencias, la mejora del uso de memoria, rendimiento de la CPU y un conjunto más robusto de opciones. Ha sido desarrollado específicamente para funcionar en sistemas basados en *Linux*, independientemente de la arquitectura. Requiere las librerías *libpcap* y *libssl*. Debido a que *Bully* almacena *pins* aleatorios y los datos de sesión en archivos de texto, no existe la necesidad de utilizar bases de datos. Ofrece varias mejoras en la detección y manejo de escenarios anómalos. Ha sido probado contra los puntos de acceso de numerosos vendedores, y con diferentes configuraciones, con mucho éxito (Larson, 2014).

***MACFaker v1.1*:** Es una herramienta automatizada para la personificación de direcciones MAC en redes abiertas que utilicen filtrado por MAC como método de seguridad. Escrita en *Python*, identifica todas las redes inalámbricas presentes y monitorea a todos los usuarios que se encuentran conectados o intentan la conexión a un punto de acceso, para luego realizar un ataque de deautenticación y utilizar la MAC de este para entrar en la red. Es una muestra de la implementación de *scripts* por parte del especialista en pruebas de penetración para agilizar el proceso de *pentesting* (ver muestra del código fuente en la figura A.3 del Anexo A). Es resultado de la investigación del autor del presente trabajo. Como requisito debe encontrarse instalado:

Requisitos:

- *Airodump-ng*
- *Python 2.6*
- *Scapy*
- Librería *PrettyTable* de *Python*

2.2.4 Herramientas para explotar las vulnerabilidades del cifrado

***Aircrack-ng*:** Es una herramienta perteneciente a la *suite Aircrack-ng*, que opera sobre sistemas operativos *Linux* y *Windows*. Se caracteriza por el craqueo de claves mediante ataques de fuerza bruta. La función principal de *aircrack-ng* es el rompimiento de la encriptación WEP, WPA/WPA2-PSK mediante ataques de diccionario (Aircrack-ng, 2009b, Ramachandran, 2011).

2.2.5 Herramientas de ingeniería social

***Wifiphisher*:** Es una herramienta de seguridad que realiza ataques *phishing* automatizados contra las redes Wi-Fi con el fin de obtener claves de autenticación u otras credenciales. Es un ataque de ingeniería social que a diferencia de otros métodos, no incluye ninguna fuerza bruta. Es una manera sencilla de obtener credenciales de portales cautivos y páginas de acceso o claves secretas WPA/WPA2. *Wifiphisher* se encuentra en *Kali Linux* y se encuentra disponible bajo licencia MIT (Chatzisofroniou, 2014).

Desde la perspectiva de la víctima, el ataque hace uso de tres fases:

1. **La víctima se encuentra deautenticada de su punto de acceso**: *Wifiphisher* ataca continuamente a todos los dispositivos Wi-Fi que se encuentran vinculados al punto de acceso de destino dentro de su alcance, mediante el envío de paquetes de deautenticación.
2. **La víctima se une a un punto de acceso no autorizado**: *Wifiphisher* analiza el área y copia la configuración del punto de acceso de destino. A continuación, crea un AP no autorizado malicioso con esta configuración. También establece un servidor NAT/DHCP y redirige el tráfico a los puertos correctos. En consecuencia, debido a la interferencia, los clientes comienzan a conectarse al punto de acceso no autorizado. Después de esta fase, la víctima se le realiza un MITM.
3. **Página de configuración de enrutador de aspecto realista**: *Wifiphisher* emplea un servidor web minimalista que responde a peticiones HTTP y HTTPS. Tan pronto como la víctima solicita una página de internet, *Wifiphisher* responde con una página falsa realista que solicita credenciales (por ejemplo una que pide la confirmación de contraseña WPA debido a una actualización del *firmware* del enrutador).

Wiphisher se encuentra disponible en *GitHub* (Chatzisofroniou, 2014). Esta herramienta es muy útil y puede ser modificada para un tipo de escenario en particular, mediante la alteración del código de la aplicación o de las páginas web emergentes.

Para el correcto funcionamiento de esta herramienta se necesitan dos tarjetas inalámbricas, una para realizar las funciones de AP malicioso y otra que soporte inyección de paquetes. Además las tarjetas inalámbricas que se utilicen como AP malicioso deben ser compatibles

con *hostapd*, utilizado para la creación del punto de acceso. *Hostapd* solo soporta los siguientes *drivers*/tarjetas inalámbricas (Wireless, 2015):

- *Drivers Linux* mac80211
- *Driver Host AP* para *Prism2/2.5/3*
- *Madwifi* (*Atheros* ar521x)
- *BSD net80211 layer* (ej. *driver Atheros*) (*FreeBSD 6-CURRENT*)

Las tarjetas inalámbricas disponibles para las pruebas de penetración utilizadas en este trabajo de diploma no son compatibles con *hostapd*, por lo que elimina la posibilidad de realizar pruebas con esta herramienta.

APFaker v1.0: Es una herramienta automatizada para la creación de puntos de acceso maliciosos en redes Wi-Fi. Existe otra herramienta escrita en *bash* llamada *Wifi Honey*, la cual difiere de *APFaker* en que no detecta el SSID y configuración de seguridad de las redes inalámbricas existentes (ver figura B.52 del Anexo B).

Requisitos:

- *Airmon-ng*
- *Airdump-ng*
- *Airbase-ng*
- *Python 2.6*
- Librería *PrettyTable* de *Python*

APFaker puede ser modificado para la realización de MITM en redes Wi-Fi, solo debe tenerse en cuenta que los *drivers* cableados soporten *ip forwarding* y la compatibilidad de los *drivers* inalámbricos con *Airbase-ng* (no se pudo implementar por estas cuestiones). *APFaker* es un ejemplo de las posibilidades del especialista en pruebas de penetración para confeccionar herramientas automatizadas, y así el ahorro en esfuerzo y tiempo en la realización de pruebas de seguridad. Un fragmento del código fuente de esta herramienta se encuentra en la figura A.4 del Anexo A. Es un resultado de la investigación del autor del presente trabajo.

2.2.6 Herramientas para el craqueo y creación de diccionarios

***CoWPAtty*:** Es una implementación de ataque de diccionario *offline* contra redes inalámbricas con encriptación WPA/WPA2 que usen autenticación PSK (ej. WPA-Personal). Su autor es Joshua Wright y se encuentra licenciado por GPLv2. *Cowpatty* puede implementar un ataque acelerado si un fichero pre-calculado PMK se encuentra disponible para el SSID que es asediado. Puede encontrar más información en su página oficial (Suchi, 2015).

***Pyrit*:** Es una potente herramienta basada en *Python* que permite la creación de bases de datos para pre-calcular claves PSK explotando el poder computacional de muchas de las plataformas que se encuentran actualmente disponibles en el mercado, tales como *Nvidia CUDA*, *OpenCL,* entre otras. Su instalación es simple, solamente es necesario descargar la última versión disponible desde su página web oficial (Pyrit, 2011). Se integra fácilmente con otras herramientas como *Cowpatty*, con características de importación de bases de datos y varias facilidades. *Pyrit* se considera la herramienta más veloz en el craqueo de ficheros *handshake* WPA/WPA2. *Pyrit* es *software* libre y licenciado por *GNU General Public License* v3+. Se compila y ejecuta en una gran variedad de plataformas como *FreeBSD*, *MacOS X* y *Linux* y en arquitecturas x86-, *alpha-, arm-, hppa-, mips-, powerpc-*, s390 y procesadores *sparc*.

***OclHashcat*:** Es un *multi-hash cracker* basado en GPGPU que utiliza ataques de fuerza bruta, ataque combinado, ataque de diccionario, ataque híbrido, ataque de máscara, y ataque basado en reglas. Este *GPU cracker* se encuentra fusionado con la versión de *oclHashcat-plus* y *oclHashcat-lite*, ambos muy utilizados pero se encuentran descontinuados. Actualmente se encuentra disponible en su página web oficial *Hashcat* (Hashcat, 2015b).

***Hashcat*:** Es la herramienta de recuperación de claves basada en CPU. No es tan rápida como su contraparte *GPU oclHashcat*, pero puede disminuir el trabajo fácilmente mediante el uso de un buen diccionario Actualmente se encuentra disponible en su página web oficial *Hashcat* (Hashcat, 2015a).

El beneficio de usar *Hashcat* es que permite la creación de reglas personalizadas que coincidan con un patrón de palabra y realizar un ataque de fuerza bruta. Esta es una alternativa al uso del ataque de diccionario ya que permite la prueba de todas las posibles

combinaciones existentes dado un set de caracteres. *Hashcat* puede craquear contraseñas WPA/WPA2 y se puede utilizar para obtener contraseñas MD5, *phpBB*, MySQL y SHA1. El uso de *Hashcat* es una buena opción si se conoce uno o dos caracteres de una contraseña.

***Crunch*:** Es un generador de listas de palabras basado en un conjunto de caracteres brindados por el usuario. Genera todas las combinaciones y permutaciones posibles para su uso posterior en sus herramientas de diccionario o de fuerza bruta. Soporta caracteres alfanuméricos bajos y altos, así como caracteres especiales. También posee la capacidad de dividir la salida en varios archivos, basado en el número de líneas o tamaño de archivo designado, además tiene la posibilidad de pausar y reanudar, lo cual es útil cuando se generan grandes listas de palabras que pueden tardar un tiempo para compilar completamente. Esta herramienta se encuentra por defecto en la instalación de *Kali Linux*. Actualmente se encuentra disponible en *Sourceforge* (Sourceforge, 2013).

***Genpmk*:** La instalación de *Cowpatty*, cuenta con esta herramienta bastante útil que permite la generación de un fichero con los *hashes* de cada clave PMK (PSK) que es posible obtener de un diccionario de claves con su respectivo SSID.

2.2.7 Herramientas integradoras

***Fern Wifi Cracker*:** *Software* de auditoría de seguridad escrito en *Python*. Utiliza la biblioteca de *Python* Qt GUI. El programa es capaz de descifrar y recuperar claves WPA/WEP/WPS, además ejecutar otros ataques basados en sistemas inalámbricos o en redes *Ethernet*. El *software* funciona en cualquier computadora con distribución *Linux* con los requisitos previos.

Requisitos:

- *Aircrack-ng*
- *Python-Scapy*
- *Python Qt4*
- *Python*
- *Subversion*
- *Xterm*

- *Reaver* (para ataques WPS)
- *Macchanger*

Fern Wifi Cracker brinda soporte para las siguientes características:

- Craqueo de WEP con fragmentación, *Chop-Chop, Caffe-Latte, Hirte,* reenvío de peticiones ARP o ataque WPS.
- Craqueo WPA/WPA2 con diccionario o ataques WPS.
- Salva automática de clave en la base de datos ante un ataque exitoso.
- Sistema automático de ataques a puntos de acceso.
- Secuestro de session.
- Geo-localización de dirección MAC de punto de acceso.
- Motor interno MITM.
- Ataques de fuerza bruta (HTTP, HTTPS, TELNET, FTP).
- Soporte de actualizaciones.

***Wifite*:** Es un proyecto de *Google Code* realizado en *Python* para automatizar el proceso de auditoría en redes Wi-Fi. El *software* fue diseñado y probado para distribuciones *Linux* (Wifite, 2014). Actualmente se encuentra disponible en *GitHub* (Merkler, 2015), (ver figura A.3 del Anexo A).

Características:

- Ordenamiento de objetivos por intensidad de señal (en dB). Brinda prioridad de ataque a los puntos de acceso más cercanos.
- Deautentica clientes de redes ocultas para revelar SSID.
- Numerosos filtros para especificar tipos de ataques (WEP/WPA/ambos, por encima de ciertas intensidades de señal, canales, entre otros).
- Ajustes personalizables (tiempos de espera, paquetes/seg, entre otros).
- Característica de "anónimo"; cambia la dirección MAC a una dirección aleatoria antes de atacar, y vuelve a su estado inicial al terminal el ataque.

- Todos los WPA *handshake* capturados se copian en el directorio actual.
- WPA inteligente de deautenticación; existencia de ciclos entre clientes y deautenticaciones por difusión.
- Muestra el resumen de la sesión en la salida; muestra las claves obtenidas.
- Todas las contraseñas obtenidas se guardan en **cracked.txt**.
- Permite la actualización mediante el comando**: ./wifite.py -upgrade**

Requerimientos

- Sistema operativo *Linux*
- *Python* 2.6 o 2.7
- *Suite aircrack-ng* (v1.1)
- *Reaver* para ataques WPS
- *Pyrit, cowpatty, tshark*: no son necesarios, pero contribuye a las capturas de los *handshake* WPA

2.3 Conclusiones parciales

El estudio y análisis de las principales herramientas para el descubrimiento, monitoreo y explotación de vulnerabilidades informáticas en redes inalámbricas es de vital importancia para realizar *pentesting* a las redes Wi-Fi de empresas e instituciones, de tal manera que sea posible el despliegue de estas redes de forma segura. El lenguaje de programación *Python* ha tomado un lugar cimero en lo que respecta a la confección de herramientas de seguridad, es recomendado su uso por los especialistas en programación por su versatilidad y eficiencia. Entre las distribuciones *Linux* diseñadas para la seguridad informática se escoge a *Kali Linux* por ser uno de los sistemas operativos más populares y efectivos para los profesionales de seguridad informática. En el proyecto se desarrollan algunas aplicaciones en *Python* que contribuyen a la aplicación más eficiente del proceso de *pentesting* en las redes Wi-Fi.

CAPÍTULO III. Pruebas de penetración en redes inalámbricas Wi-Fi

En el capítulo se realizan pruebas de penetración en entornos Wi-Fi, con el objetivo de vulnerar configuraciones y mecanismos de seguridad planteados en el capítulo 1 mediante la utilización de las herramientas descritas en el capítulo 2. Estas pruebas se han realizado en un ambiente controlado para cumplir con las normativas del *hacking* ético, solo con fines educacionales, cuyo principal objetivo ha sido dirigirlo al estado presente de la red inalámbrica Wi-Fi presente en la Universidad Central "Marta Abreu" de las Villas, aunque se trata de manera general otros escenarios.

3.1 Configuraciones iniciales

El proceso de pruebas de penetración requiere un correcto funcionamiento del *hardware* a utilizar y de las herramientas a aplicar, de ahí el éxito del mismo. Una mala configuración puede ocasionar lentitud en el proceso y resultados poco convincentes.

3.1.1 Comprobación de adaptadores inalámbricos para el *pentesting*

Para realizar el *pentesting* en redes Wi-Fi es necesaria la comprobación del *hardware* a utilizar. Muchas veces el adaptador inalámbrico que posee una *laptop* u otro dispositivo no presenta las características de modo monitor (modo promiscuo), o sea la capacidad de inyección y captura de paquetes simultáneamente para realizar las pruebas de seguridad. Para ello se debe comprobar la lista de compatibilidad en *Aircrack-ng* (Aircrack, 2009f). Se recomienda el uso de los adaptadores inalámbricos Alfa AWUS036NHR, Alfa AWUS036H y TL-WN722N para una correcta compatibilidad con *Kali Linux*.

Otra manera de comprobar si el *chipset* Wi-Fi es funcional, es a través de la realización de una prueba de inyección con *aireplay-ng*. Se escribe en consola:

```
# aireplay-ng -9 <interfaz inalámbrica>
```

Para la realización de las pruebas de penetración se utilizan los siguientes controladores de red inalámbricos:

- Ralink corp. RT5390 *Wireless* 802.11n (Tarjeta inalámbrica interna de la computadora portátil ASUS X401U con procesador AMD E2 1.8Ghz y 4GB de memoria RAM)

- Ralink corp. RT2573 Wireless 802.11bg (*Dongle* externo)

Para obtener la información de los controladores inalámbricos se escribe en consola:

```
# lscpi –v | grep –iE 'Wire'
```

Además se utiliza el punto de acceso *TrendNet* TEW-430APB (Trendnet, 2015), actualmente descontinuado por *Trendnet*, usado comúnmente en la red UCLV-WIFI, para realizar las pruebas de penetración.

3.1.2 Conexión a una red inalámbrica abierta.

1. Se comprueban las redes inalámbricas que detecta la interfaz inalámbrica. Para mostrar el listado de las redes activas se procede con el comando:

   ```
   # iwlist wlan0 scanning
   ```

2. Se observa la red inalámbrica "TESIS-ITURRIA". El campo ESSID contiene el nombre de la red (ver figura B.1 del Anexo B).

3. Puede suceder que múltiples puntos de acceso posean el mismo SSID, debe verificarse la dirección MAC del AP para evitar confusiones.

4. Se procede con los comandos para comprobar el estado de la conexión:

   ```
   # iwconfig wlan0 essid "TESIS-ITURRIA"
   # iwconfig wlan0
   ```

   Si la conexión es satisfactoria, puede observar la dirección MAC del AP en el campo Punto de Acceso (ver figura B.2 del Anexo B).

5. En el manual del punto de acceso *TrendNet*, se conoce la dirección IP de gestión "192.168.1.100". Alternativamente, esta es la misma que la dirección IP del enrutador por defecto cuando se corre el comando en consola **route -n**. Se dispone la dirección IP de la computadora del cliente en la misma subred a través del comando **ifconfig wlan0 192.168.1.101 netmask 255.255.255.0 up**. Para verificar se escribe en consola **ifconfig wlan0** y se comprueba la salida.

6. Se realiza un **ping** al punto de acceso mediante la emisión del comando **ping 192.168.1.100.** Si la conexión de red se ha configurado correctamente, entonces se

debe observar las respuestas del punto de acceso. Se puede verificar que la respuesta proviene del punto de acceso a través del comando **arp –a** (Ramachandran, 2011). Debe observarse que la dirección MAC de la IP 192.168.1.100 es la dirección MAC del punto de acceso que se ha señalado anteriormente. Es importante tener en cuenta que algunos de los puntos de acceso más recientes pueden tener desactivados la respuesta a la solicitud de paquetes ICMP *Echo*. Esto sucede puesto que algunos de los puntos de acceso actuales sólo poseen ajustes de fábrica de configuración mínimos. En tal caso, se accede a la interfaz web para verificar la conexión.

3.1.3 Creación de una interfaz en modo monitor

1. Iniciar *Kali Linux* con la tarjeta inalámbrica conectada. En la consola, se introduce **iwconfig** para confirmar que la tarjeta ha sido detectada y el controlador se ha cargado correctamente.
2. Utilizar el comando **ifconfig wlan0 up** para activar la tarjeta inalámbrica. Verificar que la tarjeta está activa mediante la ejecución de **ifconfig wlan0**. Se debe observar la palabra **Up** en la segunda línea de la salida (ver figura B.3 del Anexo B).
3. Para establecer la tarjeta inalámbrica en modo monitor, se utiliza la utilidad *airmon-ng* que se encuentra disponible de forma predeterminada en *Kali Linux*. Primeramente se verifica si se detectan las tarjetas disponibles a través del comando **airmon-ng**. Debe observar la interfaz **wlan0** listada en la salida (ver figura B.4 del Anexo B).
4. Debe asegurarse que no existan procesos activos que interfieran con el modo monitor a través del comando **airmon-ng check**. Para detener todos los procesos interferentes, entre el comando **airmon-ng check kill**.
5. Se escribe en consola **airmon-ng start wlan0** para crear una interfaz en modo monitor correspondiente al dispositivo **wlan0**. Esta nueva interfaz en modo monitor será nombrada **mon0**. Puede verificar que se ha creado mediante la ejecución de **airmon-ng**. También, si se ejecuta **ifconfig** debe observarse la nueva interfaz **mon0** (ver figura B.5 del Anexo B).

3.1.4 Estado de la red inalámbrica Wi-Fi en la UCLV

La red inalámbrica Wi-Fi en la UCLV, es una red de gran envergadura. Cuenta con 66 puntos accesos distribuidos en las 12 facultades y otras locaciones, como puntos de accesos exteriores en posiciones estratégicas. Básicamente se brinda el acceso por Wi-Fi en "la Puerta", Rectorado, Facultad de Ingeniería Eléctrica, Construcciones y la presencia en instituciones ligadas directamente con la UCLV, como el Instituto de Biotecnología de las Plantas (IBP), CEI (Centro de Estudios de Informática), Centro de Bioactivos Químicos (CBQ), Dirección de Informatización y Comunicaciones (DIC), Centro de Estudios de Electrónica y Tecnologías de la Información (CEETI), Empresa de Automatización Integral (CEDAI), Biblioteca (CDICT) entre otros, cuyo objetivo es brindar servicio a la comunidad universitaria.

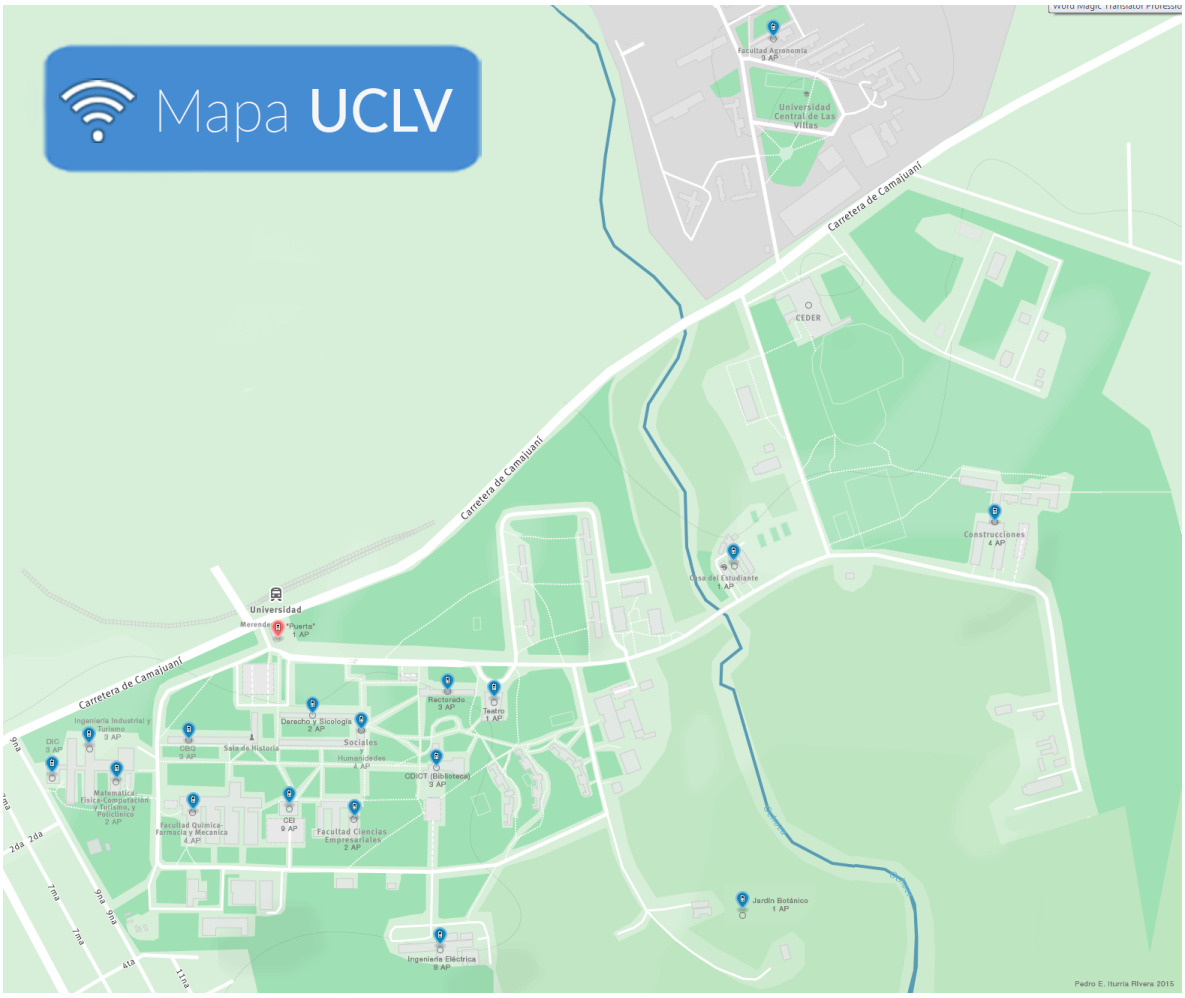

Figura 3.1. Mapa de distribución de los puntos de acceso en la UCLV.

Actualmente la red inalámbrica de la Universidad Central "Marta Abreu" de las Villas cuenta con 3030 usuarios activos con al menos un dispositivo registrado, así como 4411 dispositivos registrados entre PCs de escritorio, *laptops* y dispositivos inteligentes (ver figura 3.2). Existen numerosos SSID en la red UCLV, como el inicial UCLV-WIFI, WIFITL, TrendNet, UCLV-MFC, entre otros que dificultan el reconocimiento de una conexión legítima. La red cuenta con un servicio de DHCP que asigna a todos los dispositivos conectados al rango de direcciones 10.12.240.0/22 (1022 direcciones IP) para garantizar la concentración de todas las conexiones en una VLAN determinada. Este es un aspecto que se encuentra en propuesta de cambio, puesto que se garantiza organización, pero se sacrifica control. La propuesta es realizar una segmentación por facultad o zona de la red Wi-Fi ya que así se garantiza con seguridad en que sector ocurre algún incidente.

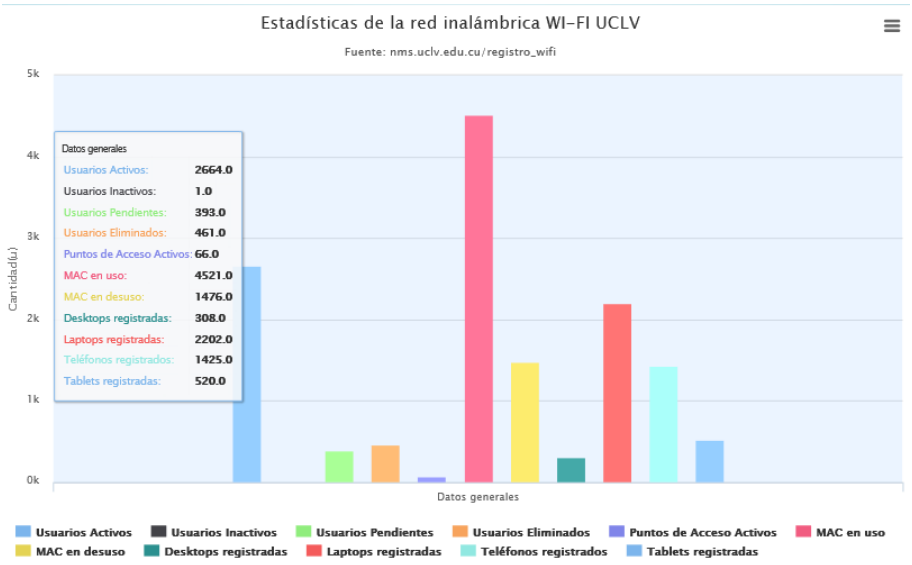

Figura 3.2. Estadísticas de la red inalámbrica Wi-Fi de la UCLV

Debido a la variedad de fabricantes, a la poca calidad de los puntos de acceso presentes en la red UCLV-WIFI, además de otros factores, se utiliza en la generalidad una configuración abierta y en algunos casos puntuales la configuración EAP-PEAP-MS-CHAP-v2 con servidor *Radius*. Las dos configuraciones presentan filtrado MAC como método de seguridad adicional. Todos los usuarios que deseen utilizar los servicios de la red necesitan inscribir sus dispositivos de manera legal, a partir de un documento con la información personal del usuario, del dispositivo a inscribir, así como la aceptación del código de ética de la red

universitaria. Además se presta el servicio de Registro Wi-Fi, un portal web, donde el usuario puede acceder a información personal, a estadísticas generales y agregar nuevos dispositivos.

3.2 Ataques al control de acceso

Los ataques al control de acceso son de los más comunes en las redes inalámbricas. Los atacantes toman ventaja de diversas vulnerabilidades, como configuraciones simples de seguridad que permiten la burla de las medidas perimetrales, que por su facilidad de implementación, son escogidas frecuentemente por los usuarios.

3.2.1 Ataque de personificación de MAC en redes abiertas

La red inalámbrica UCLV-WIFI actualmente presenta una configuración de acceso abierta (sin autenticación) con filtrado MAC como mecanismo de seguridad. Para llevar a cabo este ataque se utiliza la herramienta *MACfaker v1.1*. El uso de esta herramienta es muy sencillo, solo debe otorgar permisos de ejecución al script y luego ejecutar en consola:

```
# python macfaker1.1.py
```

La herramienta comienza a escanear las redes inalámbricas mediante la utilización de *airodump-ng* y luego captura las tramas balizas de usuario con *Scapy* (ver figura 3.3) para realizar el ataque de denegación de servicio y el cambio de dirección MAC (ver figura B.6 del Anexo B). Solo el usuario debe conectarse a la red atacada y utilizar los servicios de esta. El objetivo fundamental de la confección de la herramienta es la demostración del uso de varias herramientas, en conjunto con el lenguaje de programación *Python*, para la facilitación de las pruebas de penetración.

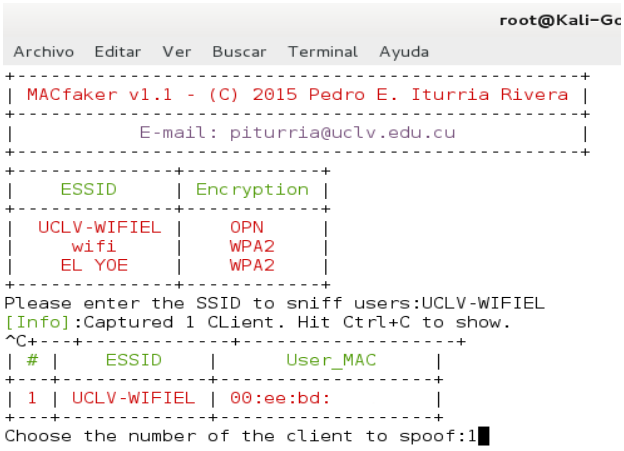

Figura 3.3. Ataque de personificación de MAC (Captura de credenciales).

3.3 Ataques a la autenticación en las redes Wi-Fi

Los ataques a la autenticación en las redes inalámbricas son un punto fundamental en las pruebas de penetración. Las credenciales en las redes inalámbricas y de manera general son el máximo objetivo a alcanzar por un atacante y lo es también para un especialista en seguridad, puesto que permite evaluar la solidez de la infraestructura. Existen diversas configuraciones que son conocidas por su vulnerabilidad y a pesar esto muchos dispositivos inalámbricos Wi-Fi brindan soporte, para garantizar así la existencia de redes vulneradas por el mero desconocimiento de los usuarios que la utilizan.

3.3.1 Ataque a la encriptación WEP

Las vulnerabilidades del protocolo WEP son conocidas desde del año 2000, pero sorprendentemente, este es usado por los usuarios de manera regular, y no solo esto, los puntos de acceso modernos todavía incluyen las características WEP. Existen muchas debilidades criptográficas en este protocolo descubiertas por Walker, Arbaugh, Fluhrer, Martin, Shamir, KoreK, y otros (Ramachandran, 2011). Para atacar la encriptación WEP se procede:

1. Se configura el punto de acceso *Trendnet* con el mecanismo de encriptación WEP, el nombre de la red inalámbrica creada se le denominó “TESIS-ITURRIA”, como se observa en la figura B.7 del Anexo B.
2. En el punto de acceso, luego de configurar el modo de seguridad WEP, se selecciona la longitud de la clave a utilizar. Como se observa en la figura B.7 se selecciona WEP con una longitud de clave de 128 bit. Se escoge la clave en ASCII “labiturria14$” de 128 bit.
3. En la máquina atacante se habilita la interfaz inalámbrica (si esta se encuentra apagada). Luego se escribe en consola **airmon-ng start wlan0** con el objetivo de crear **mon0**, la interfaz en modo monitor. Verifique que la interfaz **mon0** ha sido creada con el comando **iwconfig.**
4. Se inicia *airodump-ng* con el objetivo de localizar la red de prueba mediante el comando **airodump-ng mon0**. Como se observa en la figura B.8 del Anexo B, la red inalámbrica “TESIS-ITURRIA” con encriptación WEP.

5. Se enfoca el análisis en la red inalámbrica "TESIS-ITURRIA". Con el objetivo de monitorear los paquetes provenientes de este canal y red, se ejecuta en consola:

```
# airodump-ng --bssid 00:14:D1:30:01:B0 --channel 1 --write
WEPCrackingTesis mon0
```

   Adicionalmente, se capturan los paquetes en un fichero llamado **WEPCrackingTesis.cap** mediante el uso de la directiva **–write** (ver figura B.9 en el Anexo B).

6. Al realizar la captura el atacante nota que el número de paquetes de datos capturados debajo de la columna **#Data** de *airodump-ng* son muy pocos. En el rompimiento de la encriptación WEP, se necesita un gran número de paquetes encriptados con la misma clave, con el objetivo de explotar las vulnerabilidades del protocolo. Es por esto que se debe obtener más paquetes de datos, ya que conlleva mucho tiempo el proceso de captura de paquetes. Para acelerar este proceso se utiliza la herramienta *aireplay-ng.*

7. La herramienta *aireplay-ng* captura los paquetes ARP en la red inalámbrica y los inyecta en la misma, con el objetivo de simular respuestas ARP. Se utiliza *aireplay-ng* en una ventana separada. A través del reenvío de estos paquetes unos cientos de veces, se logra generar una gran cantidad de tráfico en la red. Aun cuando *aireplay-ng* no conoce la clave WEP, es capaz de identificar los paquetes ARP, a través del análisis del tamaño de los paquetes. El encabezado fijo del protocolo ARP y el tamaño de un paquete ARP puede ser determinado y usado para identificarlos incluso en tráfico encriptado. La opción **-3** detalla el reenvío de paquetes ARP, **-b** especifica la BSSID del punto de acceso, **–h** especifica la dirección MAC del cliente que se le aplica la falsificación y por último **--ignore-negative-one** resuelve el problema del canal fijo en **mon0**. Como es un ataque de reenvío o repetición solo funciona para direcciones MAC asociadas o autenticadas. Para ello se escribe en consola:

```
# aireplay-ng -3 –b <MAC del AP> -h <MAC falsificada> mon0 --ignore-
negative-one.
```

Como se observa en la figura B.10 del Anexo B, *aireplay-ng* es capaz de monitorear los paquetes ARP y reenviarlos hacia la red. Por otro lado, *airodump-ng* realiza el registro de los paquetes de datos, los cuales comenzarán a aumentar de manera abrupta. Todos los paquetes monitoreados son guardados en el fichero "*WEPCrackingTesis"*.

8. Se procede con el ataque activo real. Se ejecuta en consola *aircrack-ng* con la opción **WEPCrackingTesis-01.cap** en una nueva ventana. El *software aircrack-ng* comienza la obtención de la clave WEP usando los paquetes de datos del fichero. Note que es buena idea tener *airodump-ng* activo tomando los paquetes WEP, *aireplay-ng* realizando el ataque de reenvío, y *aircrack-ng* obteniendo la clave llave WEP basado en los paquetes capturados, todo al mismo momento. El resultado del ataque debe parecerse a la figura B.11 del anexo B.

El número de paquetes para la obtención de la clave WEP no está determinado, pero generalmente está en el orden de las decenas de miles (en el caso particular alrededor de 51475 IVs). En una red rápida (o mediante la utilización de *aireplay-ng*), debe tomar de 5 a 10 minutos como máximo (en este caso, en menos de 5 min). Si el número de paquetes no es suficiente, entonces *aircrack-ng* realiza una pausa y espera a que se capturen más de ellos y se reinicia el proceso, aclarando que se prueba con un número de paquetes superior (Ramachandran, 2011).

Es importante aclarar que la encriptación WEP es totalmente vulnerable (no importa cuán compleja) será craqueada por *aircrack-ng*. El único requerimiento es poseer un gran número de paquetes, encriptado con su clave (Ramachandran, 2011, Weidman, 2014).

3.3.2 Ataque a la encriptación WPA/WPA2-PSK.

Al descubrirse las vulnerabilidades del protocolo WEP en los comienzos del año 2001, WPA fue introducido como un estándar en la industria, el cual usaba TKIP para la encriptación de datos. Luego, WPA2 se convierte en el estándar de la industria ya que introdujo la encriptación AES (aunque también soporta la encriptación TKIP), la cual es más potente que TKIP. La clave WPA/WPA2 que se utiliza para la autenticación en una red inalámbrica es usada para generar otra clave única. Cinco parámetros adicionales son añadidos a la clave para generar una única clave por sesión. Los parámetros son el SSID del autenticador de red,

el *Nounce* del autenticador (ANounce), el *Nounce* del suplicante (SNounce), la dirección MAC del autenticador (*access point* MAC), y la dirección MAC del suplicante (Ramachandran, 2011).

Debe aclararse que un *nounce e*s un número arbitrario usado solo una vez en una comunicación criptográfica. Comúnmente es un número aleatorio o pseudo-aleatorio usado en un protocolo de autenticación para asegurar que las comunicaciones antiguas no puedan reutilizarse en ataques de reenvío.

Desde la perspectiva de un *hacker*, se puede usar la fuerza bruta, un ataque de diccionario, o *rainbow tables* para craquear una red WPA/WPA2, obviamente un ataque de diccionario consume menor tiempo que los otros ataques, por lo tanto esta se convierte en la primera preferencia.

Método Básico I: Ataque de diccionario de lista de palabras

El éxito de este ataque depende de la lista de palabras que se utilice. Otro requerimiento para este ataque es el *four-way handshake*, que ocurre entre el cliente y el punto de acceso, que se captura con un ataque de deautenticación (Baloch, 2015). Para obtener una clave WPA/WPA2 se prosigue con los siguientes pasos:

1. Se configura el punto de acceso con SSID "TESIS_ITURRIA" y clave *"wpacrack"* (ver figura B.12 del Anexo B)
2. Se habilita la tarjeta inalámbrica en modo monitor. Se monitorea la interfaz **mon0** para detectar redes inalámbricas que utilicen WPA o WPA2. Se utiliza el comando en consola **airmon-ng mon0** para ello. Se observa debajo de la columna **ESSID**, la red "TESIS_ITURRIA" a atacar, con encriptación WPA y cifrado TKIP (ver figura B.13 del Anexo B). Se toman los datos de dirección MAC del autenticador y el canal utilizado para su posterior utilización.
3. Con otra terminal se salvan los datos asociados al punto de acceso en un archivo **.cap**. Las entradas que se debe especificar son el canal, el BSSID y el archivo a escribir (ver figura B.14 y B.15 del Anexo B).

```
# airodump-ng –c 6 –w wpa_tesis --bssid 00:14:D1:30:03:B7 mon0
```

Donde:

-c → Canal

-w → Archivo

4. Para craquear exitosamente el protocolo WPA, se necesita capturar el *four-way handshake.* Se debe usar un ataque de deautenticación para forzar a los clientes a la reconexión con el punto de acceso (ver figura B.16 del Anexo B). Para ello se prosigue con la siguiente estructura:

```
# aireplay-ng --deauth 10 –a <AP objetivo AP> –c <Dirección MAC de un cliente> <interfaz en modo monitor>

# aireplay-ng --deauth 10 –a 00:14:D1:30:03:B7 –c  1E:3E:84:45:E4:95 mon0
```

Como se observa en la figura B.16 del Anexo B, *aireplay-ng* comienza el envío de paquetes falsos de deautenticación al punto de acceso, hasta lograr la desconexión del cliente. El cliente se reconecta y en este instante *airodump-ng* captura el WPA *handshake* (ver figura B.17 del Anexo B).

5. Para lograr una mejor comprensión, se utiliza *Wireshark*, para el análisis del f*our-way handshake*. Se observa que el tercer mensaje del f*our-way handshake* es el que porta información relevante respecto a este (ver figura B.18 del Anexo B).

6. Para la obtención de la clave WPA, se utiliza *aircrack-ng* y se especifica un archivo que contiene una lista de palabras (en inglés *wordlist*). Este archivo es utilizado en contra del archivo **wep_crack.cap** generado anteriormente. Esta lista de palabras puede ser confeccionada por el especialista en pruebas de penetración, cualquier atacante o pudiera ser descargado desde internet .Si no se especifica el destino del archivo **.cap** este se guardará en **/root** (ver figura B.20 del Anexo B).

Estructura:

```
# aircrack-ng <fichero capturado con extensión .cap> –w <lista de palabras>
```

Comando:

```
# aircrack-ng wpa_crack.cap  –w /root/Desktop/pass_list.lst
```

7. El ataque es exitoso como se muestra en la figura B.20 del Anexo B. Debe tenerse en cuenta, el prerrequisito de un ataque de diccionario, es que la clave se encuentre en el archivo de diccionario **.lst** que se utiliza en *aircrack-ng*. Existen varias herramientas para la confección de diccionarios personalizados, como *crunch*, la cual se hizo mención en el capítulo anterior. Si la clave no se encuentra en el diccionario, el ataque no es exitoso.

Método básico II: *Rainbow tables* o tablas arcoiris

En este método se utiliza la herramienta *Wifite* para la captura del *four-way handshake,* la cual es mucho más sencilla en cuanto a su uso en lo que respecta a la introducción de comandos, *genpmk* para la creación de la *rainbow table* y *cowpatty* para la obtención de la contraseña.

1. Como en el método anterior, se debe capturar el *four-way handshake*. Para detectar todas las redes existentes se escribe el siguiente comando en la terminal: **wifite** (ver figura B.21 del Anexo B). Si se desea listar solo las redes con una encriptación dada se agrega como argumento **–wep** o **–wpa**. Puede consultar más opciones mediante el comando **wifite --help**.

2. Luego se seleccionan las redes inalámbricas a las cuales se desea obtener el *four-way handshake*. Se selecciona 1 como se observa en la figura B.22 del Anexo B. Se procede con un enter, y se observa en la figura B.23 del Anexo B, la obtención del *four-way handshake*. Este archivo es guardado en **/root/hs/** bajo un nombre identificativo de la red escogida.

3. Para la generación de la tabla arcoíris se utiliza el comando **genpmk**. El *hash* de la contraseña se "hashea" con SHA1 y una semilla del SSID. Esto significa que una misma contraseña en diferentes SSID genera diversos *hashes*. El uso de este mecanismo previene la utilización de una misma *rainbow table* contra todos los APs. Se procede con el comando en consola (ver figura B.24 del Anexo B):

```
# genpmk -f <lista de palabras> -d  <nombre de la rainbow table > -s <SSID>
```

4. Para la obtención de la contraseña se utiliza la herramienta *cowpatty*. Su uso puede ser muy simple, pero también muy lento. *Cowpatty* toma la lista de contraseñas

provistas por un usuario y computa el *hash* con el SSID para cada palabra, lo cual requiere un alto procesamiento de CPU. Por esta razón se genera una *rainbow table* para agilizar el proceso. *Cowpatty* soporta el uso de tablas arcoiris, una mejora ante una lista de palabras, ya que aumenta la velocidad de obtención de las claves WPA/WPA2 PSK unas 1000 veces aproximadamente. Se procede con el comando:

```
# cowpatty –d <nombre de la rainbow table> -r <fichero del four-
handshake> -s <SSID>.
```

Como se observa en la figura B.24 del Anexo B la obtención de la contraseña "wpacrack" fue exitosa.

Método básico III: Ataque por fuerza bruta. Ataque a WPA/WPA2 con *Hashcat*

Las herramientas *cudaHashcat, oclHashcat o hashcat* presentes en *Kali Linux* son capaces de atacar y desencriptar los archivos *handshake* WPA/WPA2. La única limitante es la necesidad de convertir el fichero **.cap** al formato **.hccap**. En este método se utiliza *hashcat* ya que permite la confección de ataques personalizados con reglas predefinidas y máscaras.

Hashcat permite el uso de los siguientes juegos de caracteres para atacar al archivo *handshake* WPA/WPA2.

Juego de caracteres:

```
?l = abcdefghijklmnopqrstuvwxyz

?u = ABCDEFGHIJKLMNOPQRSTUVWXYZ

?d = 0123456789

?s = !”#$%&'()*+,-./:;⇔?@[\]^_`{|}~

?a = ?l?u?d?s
```

1. Se captura el *four-way handshake*. Se toma el mismo archivo **.cap** capturado por *wifite* en el método anterior.
2. Se limpia el fichero **.cap** con **wpaclean**. Este paso nos permite la conversión del archivo **.cap** en un formato que tanto *cudaHashcat, oclHashcat o Hashcat*

comprendan. Para convertir los ficheros **.cap** manualmente en *Kali Linux* se utiliza el siguiente comando:

```
# wpaclean <out.cap> <in.cap>
```

Note que las opciones de *wpaclean* son de orden contrario <out.cap> <in.cap> en vez de <in.cap> <out.cap> lo cual puede causar cierta confusión (ver figura B.25 del Anexo B).

3. Se convierte el archivo **.cap** al formato **.hccap**. Se necesita convertir este fichero a un formato que *cudaHashcat, oclHashcat* o *Hashcat* en *Kali Linux* comprenda. Para convertir al formato **.hccap** con *aircrack-ng* se necesita utilizar la opción **-J**.

```
# aircrack-ng <out.cap> -J <out.hccap>
```

Note que **-J** es una **J** mayúscula, no minúscula (ver figura B.26 del Anexo B).

4. Las herramientas *cudaHashcat*, *oclHashcat o hashcat* son muy flexibles. Para craquear el *handshake* WPA/WPA2 con *hashcat* se presentan dos escenarios comunes:

a. Ataque de diccionario

b. Ataque de máscara

4.a Se detecta cuál es la opción que se necesita para el craqueo de encriptación WPA, para ello se ejecuta en consola: `hashcat --help | grep WPA` y como resultado se obtiene **2500 = WPA/WPA2**. Luego se procede al ataque de diccionario con la siguiente sintaxis:

```
# hashcat -m 2500 <archivo .hccap> <lista de palabras>
```

Se obtiene la clave "*wpacrack*" como resultado del ataque (ver figura B.27 del Anexo B).

4.b Para implementar un ataque de máscara se procede de manera similar al ejemplo anterior, con la excepción que se debe establecer determinadas opciones. Se procede en la terminal con la siguiente sintaxis:

```
# hashcat -m 2500 -a 3 <archivo .hccap> <máscara>
```

Donde:

-m = 2500. Ataque a un fichero *handshake* WPA/WPA2.

-a = 3. Modo de ataque de fuerza bruta (compatible con el ataque de máscara).

La máscara para una contraseña: **ABc99 es: ?a?a?u?u?l?d?d, donde a (caracteres alfanuméricos) , u (caracteres mayúsculas), l (caracteres en minúscula) y d (números). En este caso, se supone que se conoce que la contraseña solo contiene caracteres en minúsculas (ver figura B.28 del Anexo B). Para más información en el uso de las máscaras consulte la página web oficial de *Hashcat* (Hashcat, 2015c). Existe otra alternativa, a través de la creación de archivos **.hcmask** los cuales contienen determinados caracteres (opcional) y máscaras (ej. **?1?1?1?1?d?d**) línea por línea. La ventaja del uso de archivos **.hcmask**, los cuales son archivos en texto plano, es que estos permiten al usuario de *hashcat* poseer un set predefinido de máscaras funcionales guardadas en un fichero. La sintaxis es similar al anterior ataque como se muestra:

```
# hashcat -m 2500 -a 3 <archivo .hccap> <archivo .hcmask>
```

El uso de cualquiera de las variantes de *hashcat*, permite el uso total de todo el procesamiento de los microprocesadores, en oposición al uso de *cowpatty/genpmk* que solo permite el uso de un simple hilo de procesamiento. La preferencia de *hashcat* es innegable, sobre todo con la existencia de poderosas unidades de procesamiento en los días actuales.

3.3.3 Ataques a la autenticación en enrutadores inalámbricos.

Básicamente los ataques mostrados a continuación son basados en el uso de *reaver*.

Ataques a WPS con *Reaver*

1. Establecer la tarjeta inalámbrica en modo monitor a través del comando:

```
# airmon-ng wlan0 start
```

2. En consola se ejecuta el comando **reaver** con la opciones **-i** de interfaz, **-b** la dirección MAC del enrutador a atacar y **–vv** de *verbose* para observar el proceso. La estructura debe quedar de esta manera:

```
# reaver -i mon0 -b 00:24:17:3B:7E:C9 –vv
```

Reaver comienza a atacar la clave WPS (ver figura B.29 del Anexo B).

Se observa como *reaver* inicialmente restaura los ataques realizados con anterioridad hacia el mismo objetivo, se asocia al punto de acceso y prueba claves de 8 dígitos. Este envía una petición EAPOL y el punto de acceso responde con una negación (WSC NACK), (ver figura B.30 del Anexo B). Al terminar con el ataque, guarda automáticamente la sesión.

Ataques a WPS con Fern WiFi Cracker v1.9.

1. Ejecute en consola: **fern-wifi-cracker.**
2. En el menú desplegable de la interfaz, se escoge la tarjeta inalámbrica a utilizar (WlanX). Esta debe mostrar que se encuentra habilitado el modo monitor: "*Monitor Mode Enabled on Mon(x)*" (ver figura B.31 del Anexo B).
3. Se escanean los puntos de accesos existentes en "*Scan for Access Points*".
4. Se selecciona WEP o WPS cuando este encuentre los puntos de accesos.
5. Se escoge el punto de acceso al cual desea dirigir el ataque, en el ejemplo se utiliza la red "GET_".
6. Se elige el ataque WPS (*en* inglés *WPS Attack*) justo debajo de las opciones de ataque para realizar el ataque a esta vulnerabilidad.
7. Para finalizar se inicia el ataque en "*Wi-Fi Attack*" (ver figura B.32 del Anexo B).

Conclusiones: Si es satisfactorio el ataque, se puede observar el pin WPS y la clave secreta WPA (en inglés *WPA passphrase*).

Ataques a WPS con *Wifite*

1. Ejecute en una terminal el comando: **wifite.**
2. Wifite inicializa la tarjeta inalámbrica y habilita el modo monitor, para realizar un escaneado de las redes inalámbricas. Al encontrar la red de su interés se presiona "Ctrl+C" para la selección de objetivos. *Wifite* muestra la habilitación de WPS en color verde debajo de la columna "**WPS?**" (ver figura B.33 y B.34 del Anexo B).

El craqueo WPS no es rápido, se conoce que puede tomar hasta 10 horas para completarse este tipo de ataque. Además, los enrutadores modernos poseen protección ante estos ataques

y algunos bloquean su WPS después de intentos fallidos. En muchos casos es más rápido craquear un enrutador a través de WPS que mediante la obtención de la clave WPA2.

3.3.4 Ataque a *WPA-Enterprise* y *Radius*

La red inalámbrica UCLV-WIFI presenta una configuración de autenticación WPA2-*Enterprise* y *Radius* con método de autenticación EAP-MsCHAPv2. Para un ambiente institucional, no es recomendable. Como demostración se muestra el siguiente escenario:

1. Configure el punto de acceso para usar WPA-*Enterprise*. Luego en la sección 802.1X, escriba la dirección IP del servidor *Radius*. Se utilizó en la simulación la dirección IP 192.168.1.101 Esta dirección IP se le asignó a la interfaz **eth0**, para que estuviera en la misma subred que el punto de acceso, el cual posee una dirección IP 192.168.1.100. El SSID es "UCLV-WIFI" ya que el objetivo es realizar una suplantación de identidad de la red inalámbrica universitaria. Se aclara que todas estas pruebas se realizaron en un ambiente controlado, sin atacar en realidad la infraestructura, solo con fines académicos (ver figura B.35 del Anexo B).
2. Como clave compartida (en inglés *Shared Secret*) se utilizó "*tesis*", una clave sencilla; ya que para el atacante no es importante la seguridad con el servidor *Radius*.
3. *FreeRadius* no viene instalado por defecto en *Kali Linux*, se procede su instalación a través del comando en consola **apt-get install freeradius**. En el directorio **/etc/freeradius** se encuentran todos los archivos de configuración de *FreeRadius-WPE* (ver figura B.36 del Anexo B).
4. Se procede a la edición del archivo **eap.conf**, donde se encuentra como tipo de EAP por defecto MD5, para los intereses de la prueba se debe cambiar a PEAP (ver figura B.37 del Anexo B).
5. Se edita el archivo **clients.conf**. Aquí es donde se define la lista de clientes que pueden conectarse al servidor *Radius*. La dirección IP del AP como se conoce es 192.168.1.100, se observa la clave compartida (*secret*), que debe coincidir con la configurada en el punto de acceso (ver figura B.38 del Anexo B).

6. En el directorio **/etc/freeradius/modules** se encuentran los módulos configurables que posee *Freeradius*, como se observa figura B.39 del Anexo B. Para simular EAP-MsCHAPv2 se debe modificar el archivo **mschap**.

   Se debe descomentar y habilitar las siguientes opciones:

```
use_mppe = yes

require_encryption = yes

require_strong = yes

with_ntdomain_hack = yes
```

7. Se accede al archivo **radius.conf** dentro de **/etc/freeradius** y se dispone los valores a los cuales el servidor *Freeradius* va a escuchar. Es necesario especificar el tipo de operación que este realizará (type = auth), la dirección IP de la PC atacante y el puerto de escucha (1812), este último es el puerto por defecto pero pudiera cambiarse a decisión del atacante (ver figura B.40 del Anexo B).

8. Se inicia el servidor *Radius*, a través del comando **service freeradius start**. No se observa ningún mensaje de advertencia como se observa en la figura B.41 del Anexo B.

9. Para la comprobación del servidor *Radius* se realiza una pequeña prueba a través del comando **radtest**. Para ello se utiliza un usuario de prueba establecido en el archivo **users**.

   Nota: La configuración del archivo **users** no es vital para el ataque, si se desea, el *Freeradius* provee un usuario de prueba, así como un cliente. Como se observa en la figura B.42 del Anexo B, la prueba es exitosa.

10. En la PC víctima, se observa la red UCLV-WIFI, tal como si fuese una legítima (ver figura B.43 del Anexo B).

11. La configuración de seguridad de la red UCLV-WIFI de los clientes posee varios errores que la gran mayoría de los usuarios cometen. Estos errores conllevan a un éxito rotundo de estos ataques. En la figura B.44 del Anexo B se observa habilitada la opción de recordar credenciales y en la figura B.45 del Anexo B se observa

habilitado la reconexión rápida. Esto no obstaculiza los resultados finales del atacante, solo los facilita y los acelera. El cliente solo tiene que realizar un intento de conexión a la red para brindar sus credenciales. Los certificados se encuentran deshabilitados. No existe un certificado o entidad certificadora que valide la autenticidad del servidor *Radius* con el cliente. Se debe tener en cuenta que la carencia de una entidad certificadora que sea validada por los clientes es producto del bloqueo impuesto por los Estados Unidos; a pesar de esta justificación no deja de existir la presencia de esta vulnerabilidad.

12. Se supone que las opciones previamente comentadas no se encuentran habilitadas. Al usuario se le pregunta por usuario y contraseña, como se muestra en la figura B.46 del Anexo B. En este ejercicio el usuario ingresa las credenciales usuario "peter" y contraseña "peter1610". Como es de suponer, este usuario no se registra en la red, por la inexistencia de este en el sistema falsificado.

13. Se abre el archivo **radius.log**, que se encuentra en el directorio **/var/log/freeradius**. Este archivo guarda cada instancia del servidor *Radius*, desde el proceso de encendido hasta los usuarios registrados correcta o erróneamente. En el caso de la prueba, se observa que el usuario "peter" se autenticó incorrectamente como era de esperarse (ver figura B.47 del Anexo B)

14. *Freeradius* ante cada intento de autenticación genera un archivo **.log** donde se genera el desafío y una respuesta para cada usuario. Este archivo se encuentra en el mismo directorio que el archivo de **radius.log** como se observa en la siguiente figura B.48 del Anexo B.

15. Si se analizan las últimas 7 líneas del archivo **freeradius-server-wpe.log** se observa el nombre de usuario, desafío y respuesta del usuario "peter" al intentar la conexión (ver figura B.49 del Anexo).

16. Por último para craquear MsCHAPv2, se utiliza *asleap* con la siguiente estructura:

```
# asleap –C <desafío o challenge> -R <respuesta o response> -W <archivo
de diccionario>
```

El ataque es exitoso como se observa en la figura B.50 del Anexo B, la clave del usuario es “peter1610”.

Al igual que el ataque a WPA/WPA2 el éxito final de este ataque depende directamente de la calidad del diccionario a utilizar.

3.4 Ataque de denegación de servicio

En los ataques de denegación de servicio existen dos variantes fundamentales, la denegación de servicio a un cliente específico o a un punto de acceso. Para llevar a cabo un ataque de denegación de servicio a un punto de acceso se prosigue:

1. Se habilita el modo monitor en la tarjeta de red inalámbrica con el comando en consola:

```
# airmon-ng start <interfaz inalámbrica>
```

2. Se detecta el canal del punto de acceso al cual se desea denegar el servicio. Se escribe en consola:

```
# airmon-ng -c <número del canal> <interfaz en modo monitor>
```

3.a. Con *aireplay-ng* se realiza el envío de paquetes de deautenticación en un bucle infinito (**-0 0**) al punto de acceso.

```
# aireplay-ng -0 0 -a <MAC del punto de acceso> <interfaz en modo monitor>
```

Para llevar a cabo el ataque de denegación de servicio a un cliente se prosigue:

3.b. Con *aireplay-ng* se realiza el envío de paquetes de deautenticación en un bucle infinito (-0 0) a un cliente en específico.

```
# aireplay-ng -0 0 -a <MAC del punto de acceso> -c <MAC del cliente>
<interfaz en modo monitor>
```

Aireplay-ng permite el desarrollo de ataques de denegación de servicio por varias vías. Otra variante se observa en la figura B.51 del anexo B donde se emplea la siguiente sintaxis:

```
# aireplay-ng –deauth 1000 -a <MAC del punto de acceso> -e <SSID> -c <MAC del cliente> -h <MAC del atacante> <interfaz en modo monitor> --ignore-negative-one
```

3.5 Ataque de hombre en el medio o MITM

Los ataques MITM son potencialmente peligrosos en cualquier sistema inalámbrico. Existen diversas configuraciones que pueden utilizarse para desplegar este tipo de ataque. Existen dos maneras básicas de realizarlo, la primera a través del uso de la red cableada y un punto de acceso falso y otra variante es utilizar una conexión inalámbrica legítima y un punto de acceso falso (en las pruebas se utiliza la primera variante, pero es el mismo procedimiento). Al utilizar un SSID similar a uno existente, el usuario puede conectarse al punto de acceso malicioso por cuestiones de equivocación o por conexión al SSID que porte mayor potencia de señal. Para realizar este ataque se utiliza la herramienta *APfaker* v1.0 que facilita la creación del punto de acceso falso. Se prosigue:

1. Se ejecuta el script en consola: **python apfaker.py.** Se escoge el SSID y tipo de seguridad que se requiere para el punto de acceso falso (ver figura B.52 del Anexo B).

2. Con el punto de acceso establecido se procede a la creación del puente o *bridge* entre la interfaz inalámbrica y la cableada. Se observa en la figura B.52 que se ha creado una interfaz tap **at0**. Note que las utilidades de puente en *Kali Linux* deben ser instaladas (**apt-get install bridge-utils**) con el objetivo de utilizar el comando **brctl**. La sucesión de comandos en consola se muestran (ver figura B.53 del Anexo B):

   ```
   # brctl addbr <nombre del puente>

   # brctl addif <nombre del puente> eth0

   # brctl addif <nombre del puente> at0
   ```

   Los IPs de **eth0** y **at0** pueden dejar de utilizarse puesto que las dos interfaces se encuentran integradas en el puente virtual y no necesitan dirección IP.

```
# ifconfig eth0 down
# ifconfig eth0 0.0.0.0 up
# ifconfig at0 down
# ifconfig at0 0.0.0.0 up
```

3. Se le asigna una dirección IP al puente para garantizar conectividad. Existen dos maneras de realizar esta acción, manual (como se procede en la prueba) o mediante la utilización del servicio de DHCP (si lo posee) de la red cableada (**dhclient <nombre del puente>**).

```
# ifconfig <nombre del puente> <dirección IP> netmask <máscara de red>
broadcast <dirección IP de difusión> up
# route add default gw <dirección IP de la compuerta>
```

4. Se habilita el *IP Forwarding* en el *kernel* para permitir el correcto funcionamiento del ruteo y la redirección de paquetes. Para ello se sobrescribe el archivo:

```
# echo > 1 /proc/sys/net/ipv4/ip_forward
```

5. Luego cuando un cliente desee conectarse (ver figura B.54 del Anexo B), el DHCP interno de la red (si existe) le asigna una dirección IP para asegurar la conexión. El atacante puede ahora con un simple *sniffer* como *tcpdump* o *wireshark*, analizar el tráfico del cliente.

3.6 Evaluación del estado de la red UCLV-WIFI y acciones de mitigación de vulnerabilidades.

3.6.1 Evaluación del estado de la red inalámbrica UCLV-WIFI

El nivel de seguridad de la infraestructura inalámbrica de la Universidad Central "Marta Abreu" de las Villas se considera de nivel mediano-bajo por las deficiencias presentes en el control de acceso, en la autenticación y legitimidad de los puntos de acceso existentes. La utilización de redes abiertas con filtrado MAC, la variedad de identificadores de red, la ausencia de una entidad certificadora en el método EAP-PEAP MsCHAPv2 de 802.1X y la inexistencia de sistemas de detección o prevención de intrusos, hacen propicio diversos

ataques a la estabilidad de la red, como robo de credenciales, ataques de denegación de servicio, personificación de puntos de acceso y servidor *Radius*.

3.6.2 Acciones de mitigación de vulnerabilidades.

Para lograr una infraestructura según los estándares de seguridad, cualquier red inalámbrica debe proveer soporte RSN en WPA2 y soporte 802.11i en el estándar WPA para garantizar la confidencialidad y la integridad inalámbrica. Para la implementación de RSN, cualquier AP que realice autenticación 802.1X es suficiente. El soporte para 802.1X en un punto de acceso consiste en que este solo comprenda el protocolo EAPoL y acepte un servidor *Radius* como un parámetro de configuración. El AP no necesita estar configurado para un tipo de EAP en particular, ya que este aspecto es completamente transparente para el AP. En un entorno inalámbrico típico, hay dos funciones fundamentales de gestión en juego: la gestión de autenticación de usuarios y la gestión de claves de cifrado, clave inicial, y la distribución de claves. En una implementación no empresarial las funciones de usuario y de gestión de claves pueden ser combinadas. En el libro "*Computer Security Handbook"*, del autor Seymour Bosworth se tratan diversas consideraciones para la correcta implementación de una red inalámbrica Wi-Fi con el cumplimiento de las exigencias de seguridad requeridas. Estas consideraciones y medidas se plantean a continuación (Bosworth et al., 2014):

3.6.2.1 Consideraciones de autenticación y gestión de claves en el control de acceso

- Una gestión sólida de las claves de autenticación es necesaria para establecer una comunicación segura, en contraste con un entorno de clave pre-compartida. Una clave puede presentarse en forma de una combinación usuario/contraseña, certificado, u otro material de identificación como un número RSA SecurID y PIN.
- Aprovechar las ventajas que presenta una infraestructura basada en directorio (LDAP o *Active Directory)* y utilizar la misma para autenticar a los usuarios en la red LAN inalámbrica o desplegar una infraestructura PKI. El uso de EAP y un repositorio de usuarios para la autenticación mediante el protocolo *Radius,* es el método estándar de autenticación. Una infraestructura PKI, proporciona claves de clientes públicas/privadas para cada punto final inalámbrico y elimina efectivamente la posibilidad que un atacante pueda capturar un *hash* del desafío de la contraseña.

- Las implementaciones robustas EAP incluyen un túnel TLS para establecer entre el cliente y el servidor de autenticación antes de que cualquier información de identificación se intercambie.

3.6.2.2 Consideraciones de gestión de encriptación de claves

Los algoritmos de cifrado modernos se han perfeccionado para lograr algoritmos robustos, como el AES, y son sólo tan débiles como su eslabón más débil: su entrada. Las mejores funciones de gestión de claves sólo están disponibles con las últimas evoluciones de las tecnologías inalámbricas. WPA2-Enterprise AES/CCMP es el estándar y debe ser el requisito mínimo para cualquier nueva implementación. WEP, WPA y WPA2 presentan múltiples métodos de gestión de claves con la mejora en cada generación de 802.11 de los esquemas de autenticación o cifrado (Ramachandran, 2011).

Servidor de autenticación y certificados de clientes

La autenticación basada en certificado entre un servidor de autenticación y un cliente es el estándar de autenticación inalámbrica. Los certificados X.509 son utilizados para la autenticación de servidor al cliente, del cliente hacia el servidor o ambos. Para utilizar de manera ventajosa los certificados basados en punto final en lugar de autenticación por contraseña basada en directorios se requiere la implementación de una infraestructura de clave pública. En un PKI es necesario la actualización regular de los certificados asociados con dispositivos de punto final y firmarlos mediante el uso de un Certificado de Autoridad Raíz (del inglés *Certificate Authority*, CA) o fuera del CA raíz.

3.6.2.3 Usos más comunes de los certificados de clientes o de los puntos de accesos dentro de las implementaciones inalámbricas empresariales modernas

1. Autenticación de certificados solo en servidor

Una implementación RSN EAP debe exigir, como mínimo, un certificado de servidor de autenticación firmado. Esto asegura que los clientes que intentan establecer una conexión, la realicen con un servidor y AP confiables. En ausencia de un certificado de servidor, los clientes son susceptibles a ataques MITM, que pueden ser utilizados para capturar *hashes* EAP desafío-respuesta. El protocolo RSN asegura que un cliente debe autenticar al servidor de autenticación antes de autenticarse el mismo ante el servidor. Esta secuencia protege los

ataques de personificación de servidor *Radius* y puntos de acceso maliciosos si se configura correctamente. Existen implementaciones inalámbricas que toman las ventajas de los certificados de autenticación de servidor, pero no la hacen cumplir a nivel del suplicante, por lo que renuncian a toda garantía de seguridad por este control. Un atacante puede personificar un servidor de autenticación legítimo a través del uso del mismo SSID de la red inalámbrica y realizar un secuestro al proceso de autenticación.

2. Autenticación basada en certificados mutuos cliente/servidor:

Cada cliente posee un certificado firmado por la autoridad raíz de la empresa e instalado en el almacén de certificados del sistema operativo. EAP-TLS se utiliza habitualmente en este tipo de infraestructura PKI para autenticar el certificado del servidor y el certificado del cliente mediante la validación de la firma del certificado. Este certificado de cliente también contiene información para identificar al usuario y cualquier otra información pertinente para su uso por el servidor de autenticación.

3. Autenticación basada en cliente y servidor con certificados de cliente externos:

El certificado del cliente (clave privada) se puede almacenar en una *smart card* o tarjeta inteligente, para requerir que el usuario posea el certificado separado del sistema operativo, en el caso de que el ordenador portátil se pierda o sea robado. La tarjeta inteligente debe insertarse en el equipo antes de intentar la autenticación.

Configuraciones recomendadas para el suplicante

La configuración del suplicante es un componente crucial y comúnmente pasado por alto en una implementación inalámbrica segura. Desplegar una red inalámbrica segura dentro de la empresa o institución protege la red interna. Habilitar la conectividad inalámbrica en los niveles de la organización significa que los ordenadores portátiles, que se encuentran fuera de los límites de la institución, puedan brindar información de las características de la red inalámbrica a la que normalmente acceden, solo con poseer sus interfaces inalámbricas encendidas. Una interfaz inalámbrica encendida no implica un riesgo elevado, pero cuando se combina con configuraciones por defecto presentes en los sistemas operativos y *drivers*, pueden proporcionar fácilmente todo tipo de información de conectividad a los atacantes en escucha. La mayoría de los suplicantes inalámbricos son susceptibles a algún tipo de ataque,

lo que potencialmente puede comprometer al punto final y reestablecer la conectividad del atacante cuando el cliente acceda a la red inalámbrica de la empresa o institución.

3.6.2.4 Recomendaciones finales para asegurar una configuración suplicante

1. Desactivar solicitudes de las tramas balizas SSID (del inglés SSID *Probe Requests*): Muchos sistemas operativos y suplicantes de terceros exploran de forma continua para encontrar los SSID conocidos. Este proceso se realiza para asegurar que el cliente se pueda conectar a un punto de acceso o si el cliente ha perdido el intervalo de baliza del AP. El intervalo de baliza se define como el período de tiempo entre cada envío de tramas de señalización a todos los clientes no asociados, con el objetivo principal de la identificación ante un AP cercano. Desactivar esta función evita que un atacante monitoree pasivamente el tráfico 802.11 y que adquiera la lista de SSID a la que un cliente desee conectarse. Esta información puede ser aprovechada para coaccionar al cliente que se conecte a un AP no autorizado.

2. Perfiles inalámbricos: Los perfiles inalámbricos se deben configurar para separar los perfiles corporativos de los perfiles de redes abiertas o del hogar. Los suplicantes comúnmente proporcionan esta capacidad para evitar la situación en la que un empleado de una empresa utiliza su computadora fuera del ámbito laboral y su controlador inalámbrico transmite el SSID corporativo sin este conocerlo. Esta situación que incita a un ataque de *AP* no autorizado para capturar el *hash* de la clave pre-compartida WPA/WPA2 o el *hash* del desafío de la clave basada en directorio.

Configuración del servidor para autenticación segura:

Autoridad certificadora: Un suplicante sólo debe aceptar los certificados de servidor de autenticación, que fueron firmados por el CA raíz designado. Esto impide la capacidad de conectarse a un punto de acceso y servidor de autenticación, que muestre una combinación válida de CN (del inglés *Common Name*) firmado legítimamente pero con una diferente CA raíz.

Certificado de validación: Los usuarios no deben tener la capacidad de sobrescribir o habilitar el uso de un certificado de servidor no confiable durante el proceso de autenticación. De forma predeterminada, muchos suplicantes poseen la opción de ignorar la advertencia de certificado no válido, lo que permite la autenticación. El suplicante debe configurarse para

no solicitar las credenciales y eliminar cualquier posibilidad de éxito en los intentos de autenticación con un AP con un certificado no válido o no confiable.

Nombre común (CN): Asegurar que el nombre común del servidor de autenticación se especifica en el cliente. Esto evita la conexión a un AP y servidor de autenticación que poseen un certificado de confianza (suponiendo una CA raíz específica *non-enterprise*) pero utiliza un nombre de host incorrecto/FQDN.

3.6.2.5 Sistema de detección de intrusos inalámbrico (WIDS)

Un WIDS (del inglés *Wireless Intrusion Detector System*) es capaz de monitorear el espectro 802.11 en la capa de enlace de datos y la capa MAC, independientemente del cifrado o autenticación presente. Aunque un WIDS no se puede utilizar en lugar de un IDS cableado, puede proporcionar visibilidad en un segmento de la red donde existe un control prácticamente nulo. Un WIDS no es exclusivo de las empresas que despliegan una red inalámbrica, este puede proporcionar un valor para las que no posean este tipo de infraestructura, ya que sirve para asegurar la detección del uso no autorizado de las conexiones inalámbricas y alertar adecuadamente ante un AP no autorizado. Existe otra variante como los Sistemas de Prevención de Intrusos (del inglés *Intrusion Prevention Systems*, IPS) que se diferencian básicamente con los WIDS en que estos poseen la habilidad adicional de bloquear el tráfico sospechoso de manera automática. Se propone la utilización de estos sistemas como complemento indispensable en cualquier solución inalámbrica moderna y su implementación inmediata en la red UCLV-WIFI por el incremento constante de los usuarios inalámbricos de esta red.

Usos

Debido a la encriptación, que tiene lugar en la capa de enlace de datos y por encima de este, un WIDS no puede inspeccionar el tráfico de la capa de datos. El valor fundamental de un WIDS reside en la capacidad para proporcionar seguridad en el control del espectro inalámbrico, a través del análisis de cambios no autorizados.

Los WIDS dedicados pueden ofrecer los siguientes servicios:

- Identificar los puntos de acceso no autorizados en todas las ubicaciones físicas (puntos de acceso maliciosos).

- Detectar y bloquear el uso de diversas herramientas de "hackeo" inalámbrico
- Identificar y monitorear continuamente la encriptación y los mecanismos de autenticación en los puntos de acceso autorizados.
- Detectar y bloquear las conexiones de clientes a puntos de acceso internos con tarjetas inalámbricas no autorizadas.
- Determinar los parámetros de configuración del cliente solicitante en relación con la transmisión de tramas balizas SSID conocidas.

Métodos de detección de AP no autorizados

***White lists* de AP corporativos:**

1. Por fabricante: Los AP no autorizados se pueden detectar mediante la inspección del modelo o marca. La realización de una búsqueda de la base OUI de la dirección MAC puede proveer esta información.

2. SSID: Los AP no autorizados pueden ser detectados a través de la identificación de las tramas balizas de transmisión SSID que contienen un SSID empresarial no estándar (como "Linksys" o "Trendnet").

3. Dirección MAC: Una lista estática de direcciones MAC autorizadas de los AP puede obtenerse a través de SNMP o por otros medios y ser proporcionada al WIDS como puntos de acceso autorizados.

4. Tipo de cifrado y autenticación: Los WIDS pueden intentar conectarse a los posibles AP no autorizados y enumerar los niveles de cifrado disponibles y las peticiones de autenticación. Un nivel o método no estándar podría indicar un punto de acceso ilícito.

5. Clientes asociados: Un WIDS puede realizar análisis de los clientes conectados a un potencial AP no autorizado para determinar si este está en la *white lists* basada en el **OUI**.

6. Intensidad de la señal: Un WIDS se encuentra constantemente en el análisis del espectro y puede alertar sobre la detección de anomalías de un nuevo AP en estrecha proximidad al sensor mediante la determinación de la intensidad de la señal del supuesto punto de acceso no autorizado medido en dBm. Si la intensidad en dBm cae dentro de un umbral configurado por el usuario, puede ser activada una alerta.

Contramedidas

Al detectar un AP no autorizado, los WIDS pueden configurarse para desactivar el puerto del conmutador al cual este se encuentra conectado, a través de comandos SNMP o para enviar una alerta a la consola central. Un sensor WIDS también se puede colocar en modo bloqueo, mediante la utilización de paquetes deautenticación falsificados 802.11 para evitar que los clientes establezcan una conexión exitosa con el AP no autorizado.

Medidas menos efectivas

Además de estos controles, existen algunos controles que por sí mismos no ofrecen mucha protección, pero en conjunto pueden reducir la tasa de éxito de un atacante. Estos ofrecen poca resistencia a un atacante experimentado pero causan la toma de varios pasos antes de la exposición de su objetivo.

1. Restricciones basadas en dirección MAC: En función de las capacidades de cada punto de acceso, es posible restringir la asociación a un AP, basado en la dirección MAC, incluso en un entorno empresarial. Existen métodos para actualizar dinámicamente las asignaciones de direcciones MAC, similar a la asignación de VLAN por cable basada en MAC.

2. SSID oculta: El encubrimiento del SSID es un ejemplo perfecto de seguridad por oscuridad. La presencia de un punto de acceso y del BSSID puede ser identificado a través de diversas herramientas de *pentesting* capaces de analizar el tráfico 802.11, ya que los sistemas operativos actuales no muestran la presencia de un punto de acceso que no emita su SSID.

3. Aislamiento del cliente: Es un control específico de los AP, donde se controla la dirección MAC de destino del tráfico de entrada. Si el tráfico está destinado a otro cliente en el mismo AP, o SSID en un entorno de cliente ligero, se le niega el acceso. El propósito de este control es limitar la capacidad de comunicación de un atacante, con otros dispositivos conectados al AP. Esta característica es común en los puntos de acceso públicos en los que el usuario final no es examinado, ni autenticado. Dependiendo del uso empresarial de un SSID/VLAN en particular, esta función puede activarse para evitar el acceso a otros puntos finales en un entorno donde los servicios se prestan a través de otros medios inalámbricos, como una impresora; un análisis debe ser realizado para permitir que sólo un subconjunto de tipos de comunicación con dichos dispositivos sea permitido.

Propuesta de diseño empresarial seguro

Estas consideraciones de diseño deben ir acompañadas por técnicas o controles de mitigación propuestas anteriormente para proporcionar un enfoque integral.

La integración e implementación de varios tipos de controles de mitigación como los métodos EAP, SSID, infraestructuras PKI, certificados de AP, o configuraciones suplicantes son vitales durante la fase inicial de diseño de una red empresarial inalámbrica.

3.7 Conclusiones parciales

Las pruebas de penetración en redes Wi-Fi contribuyen a la mejora de la seguridad informática en una empresa o institución. Las buenas prácticas en este proceso agilizan el resultado brindando veracidad y seriedad al mismo. Conocer la metodología, las herramientas que se utilizan de manera consciente o inconsciente en el vulnerado de redes inalámbricas, permite al especialista o administrador de red, tomar medidas acordes a la gravedad de la situación. Un diseño empresarial seguro garantiza la seguridad de la red inalámbrica; es por ello la importancia de conocer la vulnerabilidad existente en cada protocolo e implementación de seguridad, para lograr, según el escenario presente y las condiciones de *hardware* y *software*, la solución más adecuada posible. Se determinó, según las pruebas de penetración realizadas a la seguridad de la infraestructura inalámbrica de la Universidad Central “Marta Abreu” de las Villas, que esta presenta un nivel mediano-bajo por las deficiencias presentes en el control de acceso, en la autenticación legitimidad de los puntos de acceso existentes. Además se propone como medida de seguridad adicional el uso de sistemas WIDS/WIPS en el campus universitario para el descubrimiento de brechas en la seguridad, así como puntos de acceso maliciosos y ataques de denegación de servicio.

CONCLUSIONES

1. Se identificaron las vulnerabilidades presentes en el control de acceso, confidencialidad, autenticación, capa física, que permitieron la implementación de ataques de hombre en el medio, denegación de servicio, WEP, WPA/WPA2-*Personal*, WPA/WPA2-*Enterprise*, de personificación de MAC, IP y servidor *Radius*, en las redes Wi-Fi.
2. Se evaluaron herramientas como la suite *Aircrack-ng, Kismet, Hashcat, Wifite, Reaver, Pyrit, Wireshark*, presentes en la distribución de seguridad *Kali Linux*, así como el uso de lenguajes de programación como *Python y Bash* para realizar pruebas de penetración en redes Wi-Fi.
3. Se determinó, que el nivel de seguridad de la infraestructura inalámbrica de la Universidad Central "Marta Abreu" de las Villas, se clasifica como mediano-bajo por las deficiencias presentes en el control de acceso, en la autenticación y legitimidad de los puntos de acceso existentes, así como por la ausencia de mecanismos de seguridad adicionales como sistemas de detección de intrusos inalámbricos.
4. Se estableció el uso de RSN EAP con certificado de autenticación firmado y configuraciones suplicantes seguras, como la medida mínima para garantizar la seguridad en una infraestructura inalámbrica Wi-Fi moderna.
5. Se determinó que la configuración WPA2-*Enterprise* con método de autenticación EAP-TLS es la más robusta para el entorno universitario por lo que se propone como método de seguridad en la red inalámbrica UCLV-WIFI.
6. A partir de la identificación de las vulnerabilidades, la evaluación de las herramientas y los procedimientos para la realización de pruebas de penetración, se realizó un manual de pruebas de penetración en entornos Wi-Fi.

RECOMENDACIONES

1. Implementación de WPA2-*Enterprise* con método de autenticación EAP-TLS en redes inalámbricas Wi-Fi.
2. Implementación de WIDS/WIPS basado en *software* libre como medida de seguridad adicional para el descubrimiento de brechas en la seguridad, así como puntos de acceso maliciosos y ataques de denegación de servicio.
3. Extender la investigación en redes como *Bluetooth* y redes telefónicas celulares.

REFERENCIAS BIBLIOGRÁFICAS

AIRCRACK-NG. 2009a. *Airbase-ng* [En línea]. Disponible: http://aircrack-ng.org/doku.php?id=es:airbase-ng [Accedido 6 de abril del 2015].

AIRCRACK-NG. 2009b. *Aircrack-ng* [En línea]. Disponible: http://www.aircrack-ng.org/doku.php?id=aircrack-ng [Accedido 6 de abril del 2015].

AIRCRACK-NG. 2009c. *Aircrack-ng suite* [En línea]. Disponible: http://www.aircrack-ng.org/documentation.html [Accedido 6 de abril del 2015].

AIRCRACK-NG. 2009d. *Airodump-ng* [En línea]. Disponible: http://www.aircrack-ng.org/doku.php?id=es:airodump-ng [Accedido 1 de abril del 2015].

AIRCRACK-NG. 2009e. *Aireplay-ng* [En línea]. Disponible: http://www.aircrack-ng.org/doku.php?id=es:aireplay-ng [Accedido 1 de abril del 2015].

AIRCRACK-NG. 2009f. *Compatibility drivers* [En línea]. Disponible: http://www.aircrack-ng.org/doku.php?id=compatibility_drivers. [Accedido 24 de enero del 2015].

AIRCRACK-NG. 2010. *Airgraph-ng* [En línea]. Disponible: http://www.aircrack-ng.org/doku.php?id=airgraph-ng [Accedido 1 de abril del 2015].

ÁLVAREZ-CAMPANA, M., COLMENAREJO, J. B., VIDAL, F. G., LEAL, R. P., MARTÍNEZ, I. & GALLO, E. V. 2009. *Tecnologías de banda ancha y convergencia de redes,Madrid, Centro de Publicaciones*.

ALLEN, L., HERIYANTO, T. & ALI, S. 2014. Kali Linux – Assuring Security by Penetration Testing. Packt.

ALLIANCE, W.-F. 2003. Wi-Fi Protected Access: Strong, standards-based, interoperable security for today's Wi-Fi networks

ALLIANCE, W.-F. 2015. *Who we are* [En línea]. Disponible: http://www.wi-fi.org/who-we-are [Accedido 28 de abril del 2015].

ARANA, P. 2006. Benefits and Vulnerabilities of Wi-Fi Protected Access 2 (WPA2)

ASSOCIATION, I. S. 2015. *IEEE Get Program* [En línea]. Disponible: http://standards.ieee.org/about/get/802/802.11.html [Accedido 20 de mayo del 2015].

ATKINSON, R. 1995. *Security Architecture for the Internet Protocol* [En línea]. Disponible: http://www.ietf.org [Accedido 8 de Abril del 2014].

BALOCH, R. 2015. Ethical Hacking and Penetration Testing Guide. CRC Press.

BEGGS, R. 2014. Mastering Kali Linux for Advanced Penetration Testing. Packt.

BELLARDO, J. & SAVAGE, S. 802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions. USENIX security, 2003. 15-28.

BIONDI, P. 2006. Scapy Documentation.

BOSWORTH, S., KABAY, M. E. & WHYNE, E. 2014. Computer Security Handbook *In:* WILEY (ed.) 6ta ed.

CACHE, J., WRIGHT, J. & LIU, V. 2010. Hacking Exposed Wireless: Wireless Security Secrets & Solutions. McGraw Hill.

CANIDO. 2013. Manual funcionamiento básico del programa más conocido en la auditoría wireless: Kismet. [En línea]. Disponible: http://hwagm.elhacker.net/htm/kismet.htm [Accedido 15 de enero del 2015].

CANO, J. J. 2004. Inseguridad informática: Un concepto dual en seguridad informática.

CARBONELL, X. T. 2013. Cómo conocer el uso actual de las redes WLAN basadas en IEEE 802.11. Universitat Politécnica de Catalunya.

CARDWELL, K. 2014. Building Virtual Pentesting Labs for Advanced Penetration

Testing. Packt.

CARBALLEIRO, G. 2013. Redes Wi-Fi en entornos Windows. *In:* ANDINA, F. (ed.).

CHAOUCHI, H. & LAURENT-MAKNAVICIUS, M. 2007. Wireless and Mobile Network Security. *In:* WILEY (ed.).

CHATZIMISIOS, P., IOSSIFIDES, A. & ALONSO-ZARATE, J. 2014. Past, Present and Future of IEEE 802.11 toward Wireless Gigabit Experience. *IEEE Globecom.* Austin,Texas,US.

CHATZISOFRONIOU, G. 2014. *Wifiphisher* [En línea]. Disponible: https://github.com/sophron/wifiphisher [Accedido 6 de abril del 2015].

CHIU, S. H. 2006. Seguridad en Redes Inalámbricas 802.11.

CLOUDCRACKER. 2012. *Divide and Conquer: Cracking MS-CHAPv2 with a 100% success rate* [En línea]. Disponible: https://www.cloudcracker.com/blog/2012/07/29/cracking-ms-chap-v2/ [Accedido 4 de marzo del 2015].

DALEY, W. M. 1999. *Data encryption standard (DES)* [En línea]. Disponible: http://csrc.nist.gov [Accedido 8 de abril del 2014].

DISTRO, S. 2014. *Security Distributions* [En línea]. Disponible: http://www.securitydistro.com/security-distros [Accedido 20 de febrero del 2015].

ENGEBRETSON, P. 2011. The Basics of Hacking and Penetration Testing. *In:* BROAD, J. (ed.). Elsevier Inc.

FLECK, B. & POTTER, B. 2002. 802.11 Security. O'Reilly.

FLUHRER, S., MANTIN, I. & SHAMIR, A. 2001. Weaknesses in the Key Scheduling Algorithm of RC4.

GAST, M. 2005. Wireless Network The Definitive Guide. 2nd ed.: O'Reilly.

GARCÍA, R. 2011. Arquitectura para el Control de Acceso de la Red inalámbrica local de la Universidad Central "Marta Abreu" de Las Villas. UCLV.

GEIER, J. 2002. *Understanding 802.11 Frame Types* [En línea]. Disponible: http://www.wi-fiplanet.com/tutorials/article.php/1447501 [Accedido 10 de marzo del 2015].

GEIER, J. 2002a. *802. 11 Data Frames Revealed* [En línea]. Disponible: http://www.wi-fiplanet.com/tutorials/article.php/3442991 [Accedido 10 de marzo del 2015].

GIFTS, N. & JONES, J. M. 2008. Python for Unix and Linux System Administration. O'Reilly.

GITHUB. 2014. *Airgraph-ng* [En línea]. Disponible: https://github.com/aircrack-ng/aircrack-g/tree/master/scripts/airgraph-ng [Accedido 2 de febrero del 2015].

GU, Q. & LIU, P. 2007. Denial of service attacks. Handbook of Computer Networks: Distributed Networks, *Network Planning, Control, Management, and New Trends and Applications*, 3**,** 454-468.

HASHCAT. 2015a. *Hashcat* [En línea]. Disponible: http://hashcat.net/wiki/doku.php?id=hashcat [Accedido 6 de abril del 2015].

HASHCAT. 2015b. *Oclhashcat* [En línea]. Disponible: http://hashcat.net/wiki/doku.php?id=oclhashcat [Accedido 6 de abril del 2015].

HASHCAT. 2015c. *Mask Attack* [En línea]. Disponible: http://hashcat.net/wiki/doku.php?id=mask_attack [Accedido 2 de febrero del 2015].

HAT, R. 2005. *Capítulo 20: protocolo SSH* [En línea]. Disponible: http://web.mit.edu [Accedido 15 de Febrero del 2015].

HELLMANN, D. 2011. The Python Standard Library by Example. Addison-Wesley.

HOLT, A. & HUANG, C.-Y. 2010. 802.11 Wireless Networks Security and Analysis. Springer.

IEEE 1997. 802.11-1997 - IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. IEEE.

IEEE 1999. 802.11a-1999 - IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: High Speed Physical Layer in the 5 GHz band. IEEE.

IEEE 1999a. 802.11b-1999 - IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and Metropolitan networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher Speed Physical Layer (PHY) Extension in the 2.4 GHz band. IEEE.

IEEE 2003. 802.11g-2003 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access

Control (MAC) and Physical Layer (PHY) Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band. IEEE.

IEEE 2004. 802.11i-2004 - IEEE Standard for information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 6: Medium Access Control (MAC) Security Enhancements. IEEE.

IEEE 2007. 802.11-2007 - IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE.

IEEE 2008. 802.11k-2008 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 1: Radio Resource Measurement of Wireless LANs. IEEE.

IEEE 2009. 802.11n-2009 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput. IEEE.

IEEE 2009a. 802.11w-2009 - IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Protected Management Frames. IEEE.

IEEE 2010. 802.11p-2010 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments. IEEE.

IEEE 2010a. 802.1X-2010 - IEEE Standard for Local and metropolitan area networks--Port-Based Network Access Control. IEEE.

IEEE 2010b. 802.1X-2010 - Revision of 802.1X-2004. IEEE.

IEEE 2011. 802.11v-2011 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 8: IEEE 802.11 Wireless Network Management. IEEE.

IEEE 2012. 802.11-2012 - IEEE Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE.

IEEE 2012a. 802.11ad-2012 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements--Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band.IEEE

IEEE 2013. 802.11ac-2013 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements--Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz.IEEE

IEEE 2013a. 802.11af-2013 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 5: Television White Spaces (TVWS) Operation.IEEE

IEEE. 2015. *OUI* [En Línea]. Disponible: http://standards-oui.ieee.org/oui.txt [Accedido 3 de marzo del 2015].

IETF, N. W. G. 1995. The ESP DES-CBC Transform.

IETF, N. W. G. 1999. The TLS Protocol Version 1.0.

IETF 2000. Microsoft PPP CHAP Extensions, Version 2(RFC 2759).

IETF, N. W. G. 2005. Security Architecture for the Internet Protocol.

IETF, N. W. G. 2005a. Internet Key Exchange (IKEv2) Protocol.

IETF, N. W. G. 2005a. Using Advanced Encryption Standard (AES) CCM Mode with IPsec Encapsulating Security Payload (ESP).

JOHNS, A. 2015. Mastering Wireless Penetration Testing for Highly Secured Environments. Packt.

KISMET. 2011. *Kismet Readme* [En línea]. Disponible: https://www.kismetwireless.net/documentation.shtml [Accedido 1 de abril del 2015].

KUO, C., WALKER, J. & PERRIG, A. 2007. Low-cost manufacturing, usability, and security: an analysis of bluetooth simple pairing and Wi-Fi protected setup. *Financial Cryptography and Data Security*. Springer.

KUROSE, J. F. & ROSS, K. W. 2013. Computer Networking. A Top-Down Approach. 6ta ed.

K.SANDHU, G., MANN, G. S. & KAUR, R. 2013. Benefit and security issues in wireless technologies: Wi-fi and WiMax. *International Journal of Innovative Research in Computer and Communication Engineering* 1.

LAKHANI, A. 2013. *Breaking WPA2-PSK with Kali Linux* [En línea]. Disponible: http://www.drchaos.com/breaking-wpa2-psk-with-kali/ [Accedido 3 de febrero del 2015].

LAKHANI, A. & MUNIZ, J. 2015. Penetration Testing with Raspberry Pi. Packt.

LARSON. 2014. *Bully attack tool* [En línea]. Disponible: https://github.com/Lrs121/bully [Accedido 6 de abril del 2015].

LIBCAP, T. 2015. *Tcpdump & Libcap* [En línea]. Disponible: http://www.tcpdump.org/ [Accedido 5 de abril del 2015].

LINUX, K. 2015. *Kali Linux Downloads* [En línea]. Disponible: https://www.kali.org/downloads/ [Accedido 2 de febrero del 2015].

LÓPEZ, P. A. 2010. Seguridad informática. In: EDITEX (ed.).

MARTINOVIC, I., ZDARSKY, F. A., BACHOREK, A., JUNG, C. & SCHMITT, J. B. 2007. *Phishing in the wireless: Implementation and analysis*, Springer.

MCCLURE, S., SCAMBRAY, J. & KURTZ, G. 2012. *Hacking Exposed 7: Network security secrets & solutions.* 7ma ed. New York: McGraw Hill.

MERKLER, D. 2015. Wifite [En línea]. Disponible: https://github.com/derv82/wifite/ [Accedido 2 de mayo del 2015].

MOHIT 2015. *Python Penetration Testing Essentials.* Packt.

NICKERSON, C. 2014. *Pentesting Standard* [En línea]. Disponible: http://www.pentest-standard.org [Accedido 26 de abril del 2015].

OPS, B. 2014. *Cracking WPA2 WPA with Hashcat in Kali Linux (BruteForce MASK based attack on Wifi passwords)* [En línea]. Disponible: http://www.blackmoreops.com/2014/03/27/cracking-wpa-wpa2-with-hashcat-kali-linux [Accedido 2 de febrero del 2015].

OSTERHAGE, W. 2011. Wireless Security. CRC Press.

ORZACH, Y. 2013. *Network Analysis using Wireshark Cookbook.* Packt.

PELLEJERO, I., ANDREU, F. y LESTA, A. 2006. *Fundamentos y aplicaciones en redes WLAN,* Barcelona Marcombo.

PERU. 2013. *Rogue Access Point using Kali Linux* [En línea]. Disponible: http://www.gosecure.it/blog/art/376/note/rougue-access-point-using-kali-linux/ [Accedido 10 de febrero del 2015].

PRITCHETT, W. L. & SMET, D. D. 2013. Kali Linux Cookbook. Packt.

PYRIT. 2011. *Downloads* [En línea]. Disponible: http://code.google.com/p/pyrit/downloads/list. [Accedido 3 de mayo del 2015].

RAMACHANDRAN, V. 2011. BackTrack 5 Wireless Penetration Testing. Packt.

GARCÍA, R. R. 2011. *Arquitectura para el Control de Acceso de la Red inalámbrica local de la Universidad Central "Marta Abreu" de Las Villas.* Maestría, UCLV.

REGALADO, D., HARRIS, S., HARPER, A., EAGLE, C., NESS, J., SPASOJEVIC, B., LINN, R. & SIMS, S. 2015. Gray Hat Hacking. McGraw Hill.

RILEY, S. 2005. *Mitigating the Threats of Rogue Machines—802.1X or IPsec?* [En línea]. Disponible: https://technet.microsoft.com/library/cc512611.aspx [Accedido 12 de marzo del 2015].

ROYER, J. M. 2004. Seguridad en la informática de empresa: riesgos, amenazas, prevención y soluciones ENI.

SERVER, W. 2010. *MS-CHAP v2* [En línea]. Disponible: https://technet.microsoft.com/es-es/library/cc731462%28v=ws.10%29.aspx [Accedido 3 de abril del 2015].

SINGH, A. 2013. Instant Kali. Packt.

SONG, Y., YANG, C. & GU, G. Who is peeping at your passwords at Starbucks?—To catch an evil twin access point. Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on, 2010.

SOURCEFORGE. 2013. *Crunch* [En línea]. Disponible: http://sourceforge.net/projects/crunch-wordlist/files/crunch-wordlist/crunch-3.6.tgz/download. [Accedido 15 de marzo del 2015].

STUBBLEFIELD, A., IOANNIDIS, J. & RUBIN, A. D. 2002. Using the Fluhrer, Mantin, and Shamir Attack to Break WEP. *Network and Distributed System Security Symposium*.

SUKHIJA, S. & GUPTA, S. 2012. Wireless Network Security Protocols: A Comparative Study. *International Journal of Emerging Technology and Advanced Engineering*.

SUSHI, W. H. F. 2015. *Asleap* [En línea]. Disponible: http://www.willhackforsushi.com/?page_id=41 [Accedido 6 de abril del 2015].

SUSHI, W. H. F. 2015. *Cowpatty* [En línea]. Disponible: http://www.willhackforsushi.com/?page_id=50 [Accedido 6 de abril del 2015].

SYSTEMS, C. 2014. *802.11 frames: A starter guide to learn wireless sniffer traces* [En línea]. Disponible: https://supportforums.cisco.com/document/52391/80211-frames-starter-guide-learn-wireless-sniffer-traces [Accedido 20 de febrero del 2015].

TANENBAUM, A. 2011. Computer Networks. 5ta ed.

TRENDNET. 2015. *54Mbps Wireless G Access Point* [En línea]. Disponible: http://www.trendnet.com/products/proddetail.asp?status=view&prod=150_TEW-430APB [Accedido 24 de mayo del 2015].

TUSHAR, S. & LAKSHMAN, S. 2013. Linux Shell Scripting Cookbook. Packt.

WEIDMAN, G. 2014. Penetration Testing. *In:* NO STARCH PRESS, I. (ed.). William Pollock.

WIFITE. 2014. *Wifite* [En línea]. Disponible: https://code.google.com/p/wifite/ [Accedido 6 de abril del 2015].

WILLIAM E. SHOTTS, J. 2013. The Linux Command Line.

WIRELESS, L. 2015. *About hostapd* [En línea]. Disponible: https://wireless.wiki.kernel.org/en/users/documentation/hostapd [Accedido 6 de abril del 2015].

WIRESHARK. 2015. *Wireshark* [En línea]. Disponible: http://www.wireshark.org/ [Accedido 4 de abril 2015].

WRIGHT, J. 2003. Weaknesses in LEAP Challenge/Response. *Abusing 802.11: Weaknesses in LEAP Challenge/Response.*

GLOSARIO

AES: Es un esquema de cifrado por bloques adoptado como un estándar de cifrado por el gobierno de los Estados Unidos. El AES fue anunciado por el Instituto Nacional de Estándares y Tecnología (NIST). Es uno de los algoritmos más populares usados en criptografía simétrica, utilizando un tamaño de bloque fijo de 128 bits y un tamaño de llave variable de 128, 192 o 256 bits.

ANSI: El Instituto Nacional Estadounidense de Estándares (del inglés *American National Standards Institute,* ANSI) es una organización sin fines de lucro que supervisa el desarrollo de estándares para productos, servicios, procesos y sistemas en los Estados Unidos.

ARP: El Protocolo de resolución de direcciones o *Address Resolution Protocol,* es un protocolo de resolución de direcciones, perteneciente a la capa de enlace de datos, responsable de encontrar la dirección *hardware* que corresponde a una determinada dirección IP.

***Bash* (del inglés *bourne again shell*):** Es un programa informático cuya función consiste en interpretar órdenes. Está basado en la *shell* de *Unix* y es compatible con POSIX. Fue escrito para el proyecto GNU y es el intérprete de comandos por defecto en la mayoría de las distribuciones de GNU con *Linux*.

BSS: Acrónimo de conjunto de servicio básico, es el grupo de estaciones que se intercomunican entre ellas. Se definen dos tipos, independientes, cuando las estaciones, se intercomunican directamente e infraestructura, cuando se comunican todas a través de un punto de acceso.

BSSID: El *Basic Service Set Identifier* por sus siglas en inglés, de una red de área local inalámbrica, es el nombre que identifica de manera única a todos los paquetes de una red. A diferencia del *Service Set Identifier* (SSID), que puede ser usado en múltiples BSS, el BSSID sólo puede hacerlo en una.

CBC-MAC: En criptografía, un código de cifrado en bloque enlazado de autenticación de mensaje (del inglés *Cipher Block Chaining Message Authentication Code*) es una técnica para la construcción de códigos de mensajes de autenticación desde un cifrado en bloque.

***Cracking*:** Es un proceso informático que consiste en descifrar contraseñas, desarrollado por los "*crackers*" o personas que irrumpen sin permiso en la computadora de otras personas.

CRC: La verificación por redundancia cíclica es un código de detección de errores usado frecuentemente en redes digitales y en dispositivos de almacenamiento para detectar cambios accidentales en los datos.

DHCP: Protocolo de Configuración Dinámica de Host o *Dynamic Host Configuration Protocol* es un protocolo de red que permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente. Este protocolo se publicó en octubre de 1993, y su implementación actual está en la RFC 2131.

***Diameter*:** es un protocolo de red para la autenticación de los usuarios que se conectan remotamente a internet a través de la conexión por línea conmutada o RTC, también provee servicios de autorización y auditoría para aplicaciones tales como acceso de red o movilidad IP. El concepto básico del protocolo *Diameter*, cuyo desarrollo se ha basado en el protocolo *Radius*, es proporcionar un protocolo base que pueda ser extendido para proporcionar servicios de autenticación, autorización y auditoría.

DSSS: Espectro ensanchado por secuencia directa (en inglés *Direct Sequence Spread Spectrum*), también conocido en comunicaciones móviles como DS-CDMA (Acceso Múltiple por División de Código en Secuencia Directa), es uno de los métodos de codificación de canal (previa a la modulación) en espectro ensanchado para transmisión de señales digitales sobre ondas radiofónicas que más se utilizan.

GPG (del inglés *GNU Privacy Guard*): Es un reemplazo de *software* libre de la suite PGP perteneciente a Symantec. GnuPG o GPC cumple con la RFC 4880, la cual constituye el estándar actual de la IETF de las especificaciones OpenPGP.

Hackear: Término adaptado del inglés "*hacking*" se refiere a la acción de explorar y buscar las limitantes de un código o de una máquina. Según el "Glosario del Argot Hacker" o "*Jargon File*", cuyo creador fue Eric S. Raymond, el término "hackear" también significa acción de irrumpir o entrar de manera forzada a un sistema de cómputo o a una red. Este último significado para muchos hace alusión al término "craquear". Sin embargo, en *The New Hacker's Dictionary2* se plantea como un significado igualmente válido para "hackear".

Handshake: Es el proceso automatizado de negociación que establece de forma dinámica los parámetros de un canal de comunicaciones establecido entre dos entidades antes de que comience la comunicación normal por el canal.

***Hash*:** A las funciones *hash* se les llama funciones picadillo, funciones resumen o funciones *digest*. Una función *hash* es una función computada mediante un algoritmo, que tiene como entrada un conjunto de elementos, como cadenas, y los convierte (mapea) en un rango de salida finito, normalmente cadenas de longitud fija.

HTML: Lenguaje de Marcas de Hipertexto (del inglés *HyperText Markup Language*), es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, entre otros.

IDS: Un sistema de detección de intrusiones (del inglés *Intrusion Detection System*) es un programa de detección de accesos no autorizados a un computador o a una red.

IEEE: Instituto de Ingeniería Eléctrica y Electrónica es una asociación mundial de técnicos e ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas.

ISM (*Industrial, Scientific and Medical*): son bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética en áreas industrial, científica y médica. En la actualidad estas bandas han sido popularizadas por su uso en comunicaciones WLAN como Wi-Fi o WPAN como *Bluetooth*.

***Libpcap*:** Es una implementación que pertenece a la interfaz de aplicación de programación pcap, para sistemas basados en Unix, la cual se utiliza para la captura de paquetes.

MAC: En las redes de computadoras, la dirección MAC (en inglés de *media access control*; en español "control de acceso al medio") es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de forma única a una tarjeta o dispositivo de red.

Modo RFMON o modo monitor: Permite el monitoreo de una red inalámbrica con una computadora que posea una interfaz inalámbrica. A diferencia del modo promiscuo, que también es usado en el *sniffing* de paquetes, no necesita la asociación con un punto de acceso o una red *ad hoc*.

NIC: En inglés, se denomina *Network Interface Card* o *Network interface controller* (NIC), es el periférico que actúa de interfaz de conexión entre aparatos o dispositivos, y también posibilita compartir recursos (discos duros, impresoras, etcétera) entre dos o más computadoras, es decir, en una red de computadoras.

OFMD: Multiplexación por División de Frecuencias Ortogonales (en inglés *Orthogonal Frequency Division Multiplexing*) es una multiplexación que consiste en enviar un conjunto de ondas portadoras de diferentes frecuencias, donde cada una transporta información, la cual es modulada en QAM o en PSK.

OUI (identificador único de organización): Es un número de 24 bits comprado a la Autoridad de Registro del Instituto de Ingeniería Eléctrica y Electrónica (IEEE) que identifica a cada empresa u organización (llamados asignados) a nivel mundial y reserva un bloque en cada posible identificador derivado (como las direcciones MAC, direcciones de grupos, identificadores para el Protocolo de acceso a subredes, etc.) para el uso exclusivo del asignado.

OSI: Modelo de interconexión de sistemas abiertos (ISO/IEC 7498-1), más conocido como "modelo OSI" (en inglés, *Open System Interconnection*), es el modelo de red descriptivo, que fue creado en el año 1980 por la Organización Internacional de Normalización. Es un marco de referencia para la definición de arquitecturas en la interconexión de los sistemas de comunicaciones.

PKI: Infraestructura de clave pública (en inglés, *Public Key Infrastructure*) es una combinación de *hardware* y *software*, políticas y procedimientos de seguridad que permiten la ejecución con garantías de operaciones criptográficas como el cifrado, la firma digital o el no repudio de transacciones electrónicas.

PMK: Es una llave criptográfica utilizada en la derivación de claves de bajo nivel. Redes basadas en tecnologías como UMTS y WiMAX usan PMK como parte de sus procedimientos relativos de seguridad.

PPTP: Protocolo de tunelización punto a punto (en inglés, *Point to Point Tunneling Protocol*), es un protocolo de comunicaciones desarrollado por *Microsoft, U.S. Robotics, Ascend Communications, 3Com/Primary Access, ECI Telematics* conocidas colectivamente como PPTP *Forum*, para implementar redes privadas virtuales o VPN.

***Python*:** Es un lenguaje de programación interpretado cuya filosofía se inclina hacia a una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

***Radius*:** Del inglés *Remote Authentication Dial-In User Service,* es un protocolo de autenticación y autorización para aplicaciones de acceso a la red o movilidad IP. Utiliza el puerto 1812 UDP para establecer sus conexiones.

RC4: Dentro de la criptografía RC4 o ARC4 es el sistema de cifrado de flujo más utilizado y se usa en algunos de los protocolos más populares como TLS/SSL (para proteger el tráfico de internet) y WEP (para añadir seguridad en las redes inalámbricas). RC4 fue excluido de los estándares de alta seguridad por los criptógrafos y algunos modos de usar el algoritmo de criptografía RC4 lo han llevado a ser un sistema de criptografía muy inseguro, incluyendo su uso WEP.

***Roaming*:** Se refiere al servicio continuado de datos cuando se sobrepasan los límites del área de cobertura del operador telefónico u otros sistemas inalámbricos

Sniffing: Escuchas, este ataque tiene como objetivo final monitorizar la red para capturar información sensible como un paso previo a ataques posteriores.

***Spoofing*:** Ataque de engaño, de falsificación, consiste en emplear un terminal cliente al que se han asociado validadores estáticos de una red para suplantar la identidad de algún miembro de la comunicación.

SSID: *Service Set Identifier* es un nombre incluido en todos los paquetes de una red inalámbrica para identificarlos como parte de esa red. El código consiste en un máximo de 32 caracteres, que la mayoría de las veces son alfanuméricos (aunque el estándar no lo especifica, así que puede consistir en cualquier carácter). Todos los dispositivos inalámbricos que intentan comunicarse entre sí deben compartir el mismo SSID. Existen algunas variantes principales del SSID. Las redes *ad-hoc*, que consisten en máquinas cliente sin un punto de acceso, utilizan el BSSID (del inglés *Basic Service Set Identifier*); mientras que en las redes en infraestructura que incorporan un punto de acceso se utiliza el ESSID (del inglés *Extended*

Service Set Identifier). Es posible referirse a cada uno de estos tipos, como SSID en términos generales.

TCP/IP: Familia de protocolos de internet es un conjunto de protocolos de red en los que se basa internet y que permiten la transmisión de datos entre computadoras.

TLS: Protocolo que garantiza la privacidad y la integridad de los datos entre aplicaciones cliente/servidor que se comunican a través de internet.

VLAN (*Virtual Local Area Network*): Es un método de creación de redes lógicas e independientes dentro de una misma red física. Varias VLAN pueden coexistir en un único conmutador físico o en una única red física.

XOR: Comúnmente conocido como OR-Exclusivo. Operación de bits que en ocasiones algunos virus utilizan como algoritmo de cifrado sencillo para ocultar el contenido o engañar a algunos antivirus o desensambladores.

ANEXOS

Anexo A. Herramientas para las pruebas de penetración en redes inalámbricas

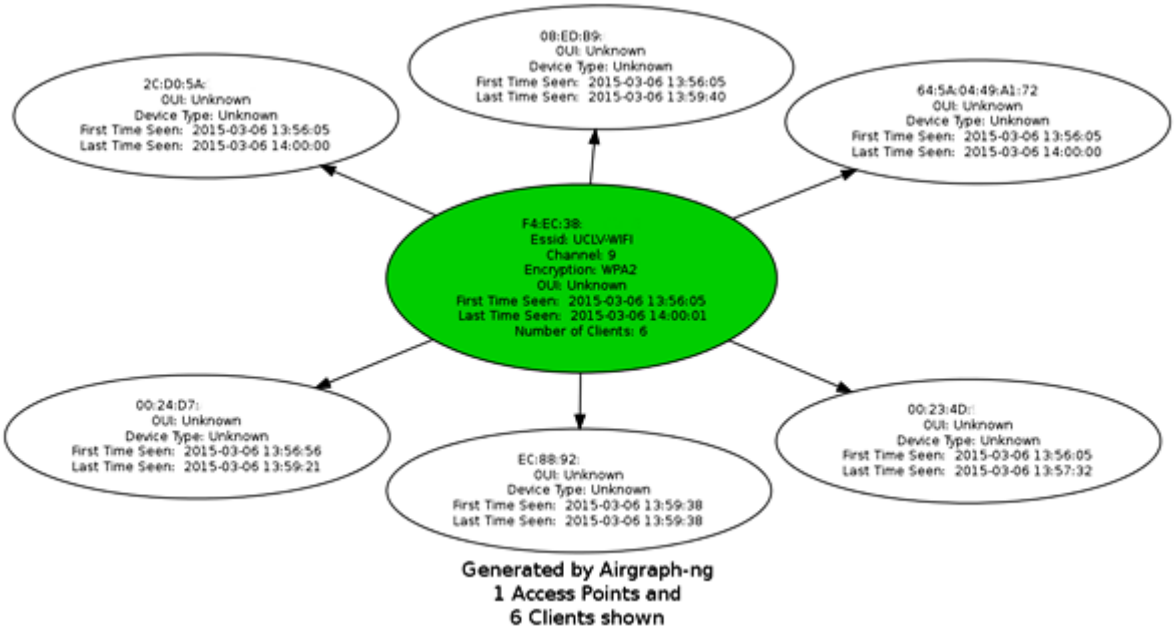

Figura A.1. Resultado de la herramienta de descubrimiento *airgraph-ng*.

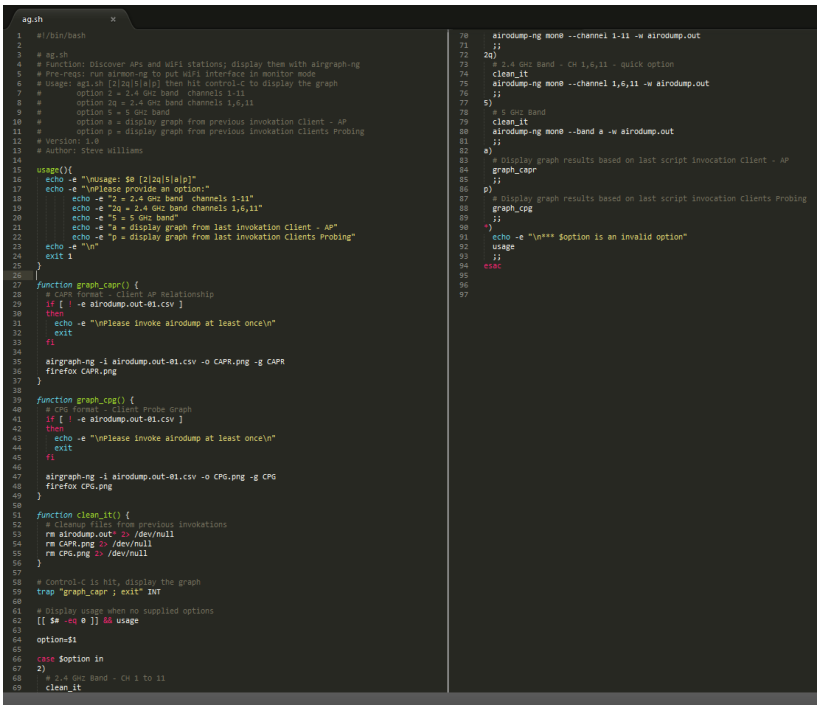

Figura A.2. Script para la facilitación del uso de *airgraph-ng*.

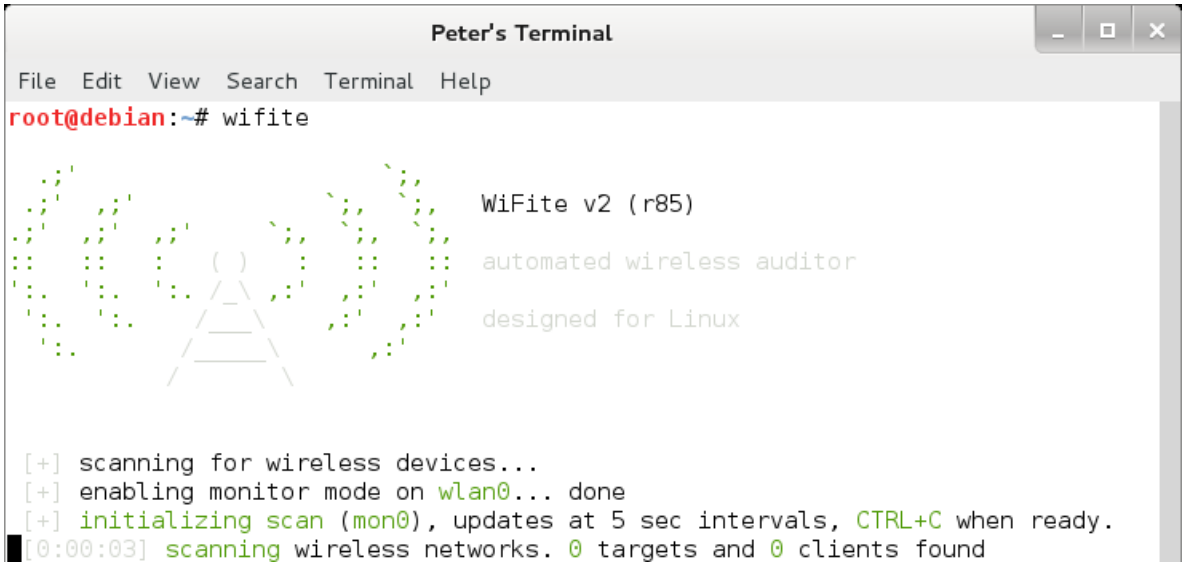

Figura A.3. Herramienta integradora *wifite*.

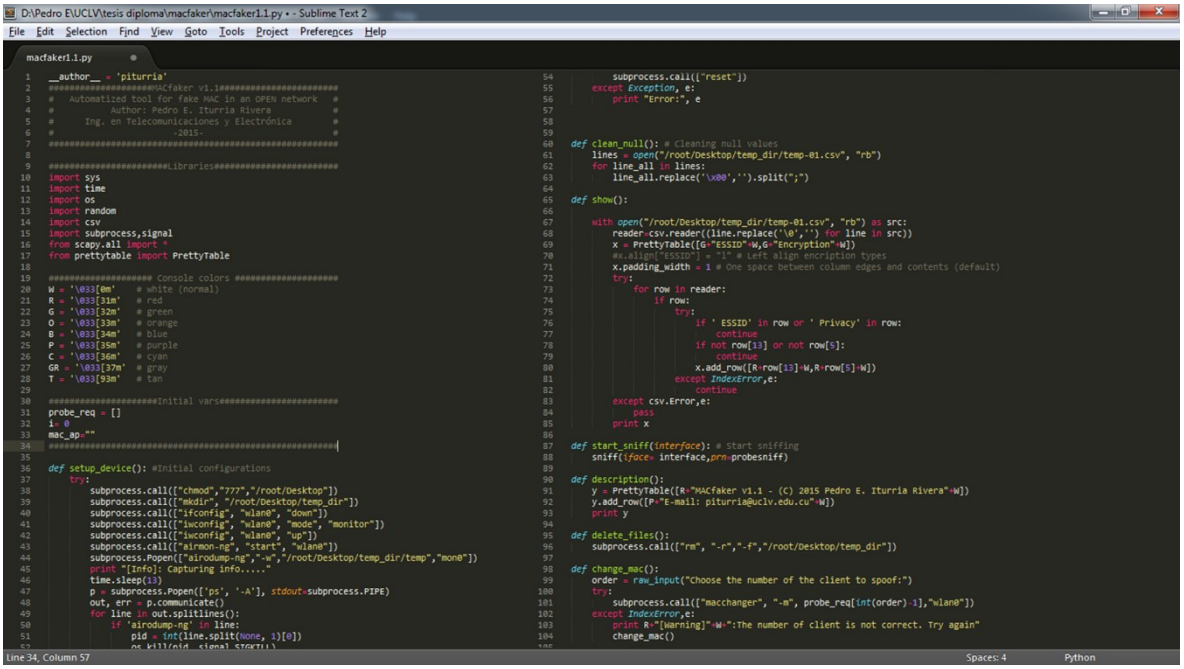

Figura A.4. Fragmento de código de *MACfaker v1.1*.

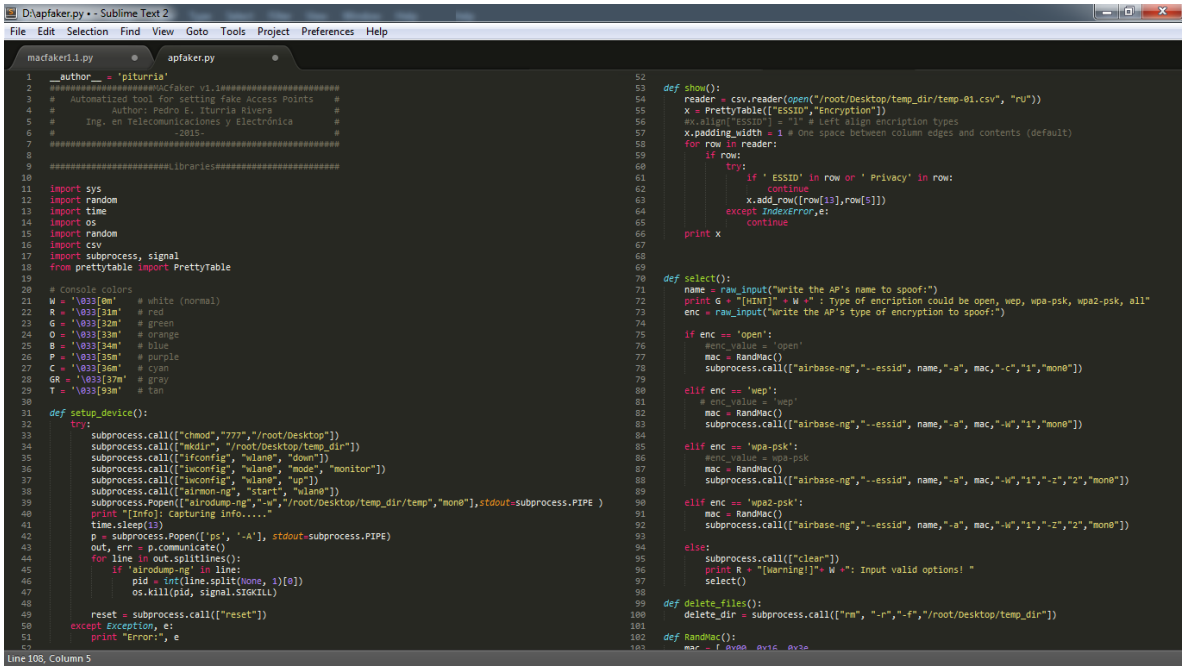

Figura A.5. Fragmento de código de *APfaker v1.1*.

Anexo B. Pruebas de penetración en redes Wi-Fi

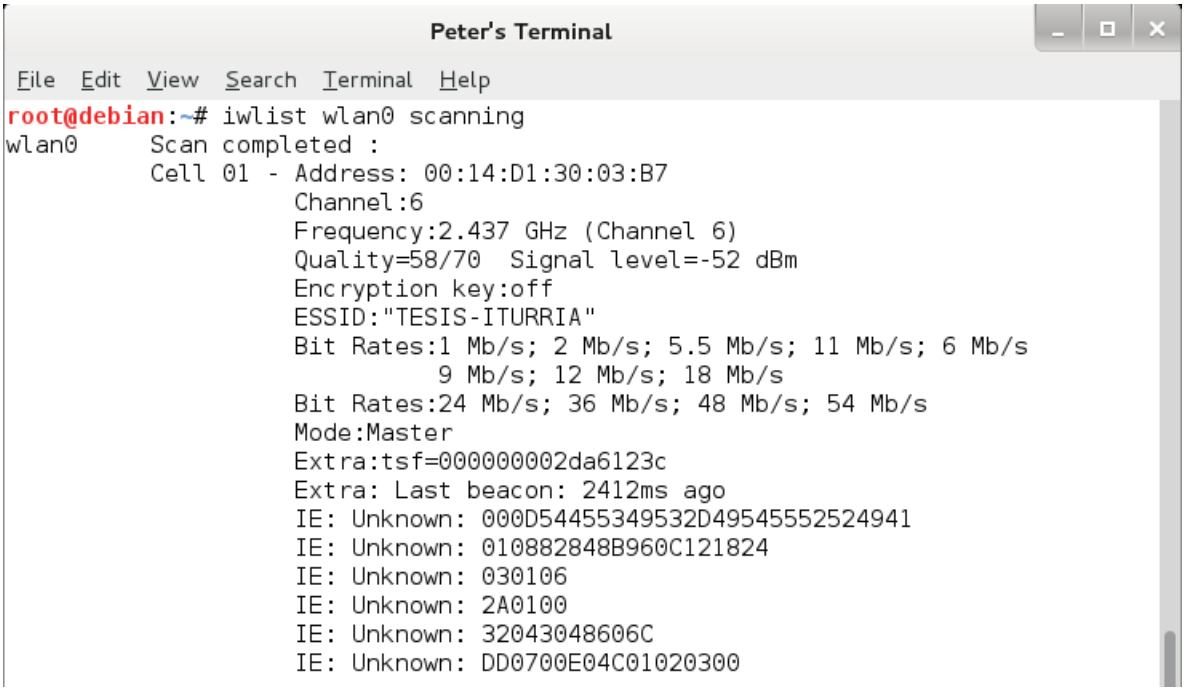

Figura B.1. Detección de redes inalámbricas.

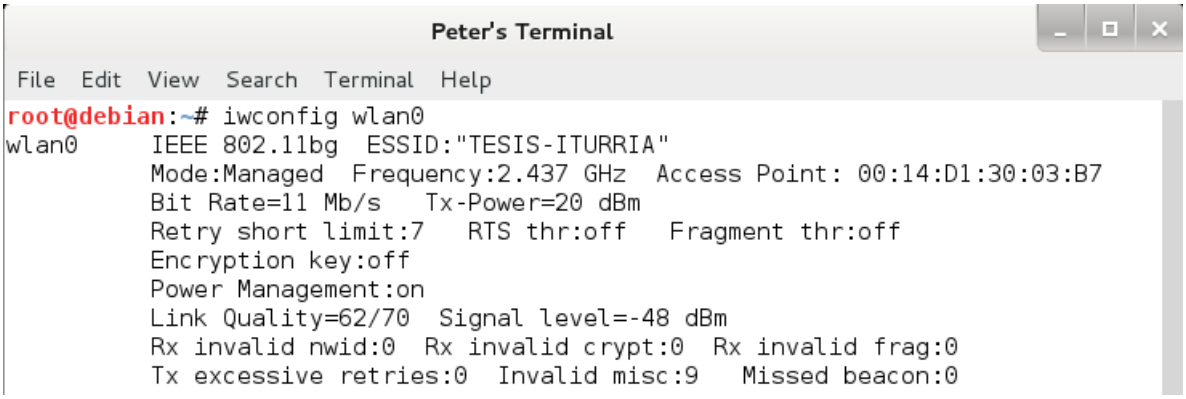

Figura B.2. Comprobación de conexión.

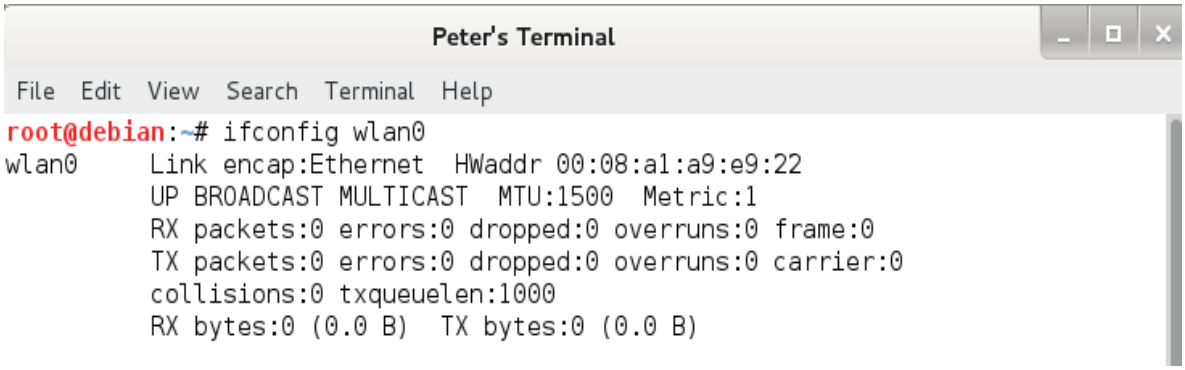

Figura B.3. Comprobación del funcionamiento de la tarjeta inalámbrica.

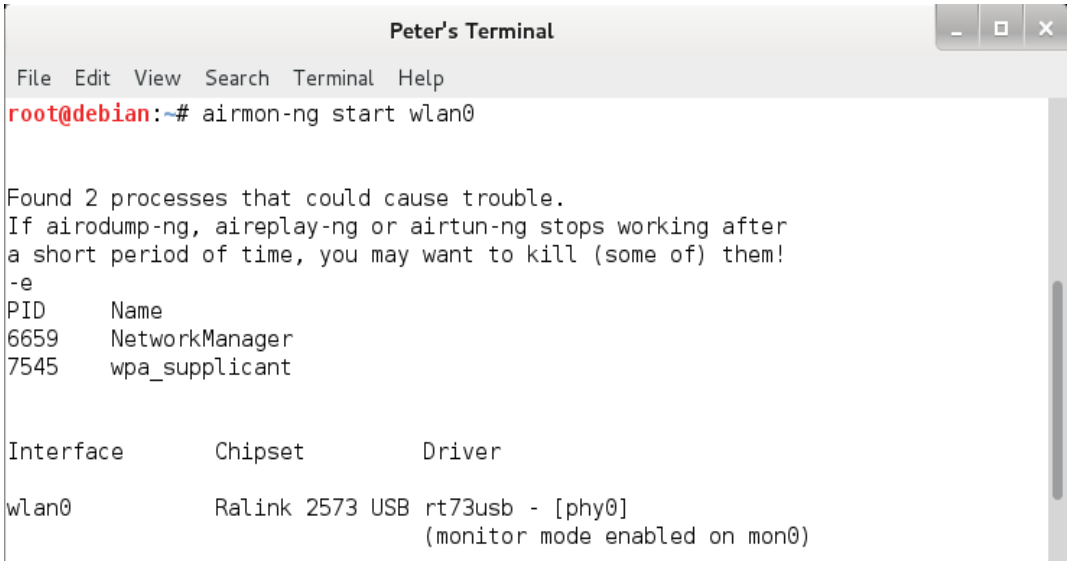

Figura B.4. Creación del modo monitor de la tarjeta inalámbrica.

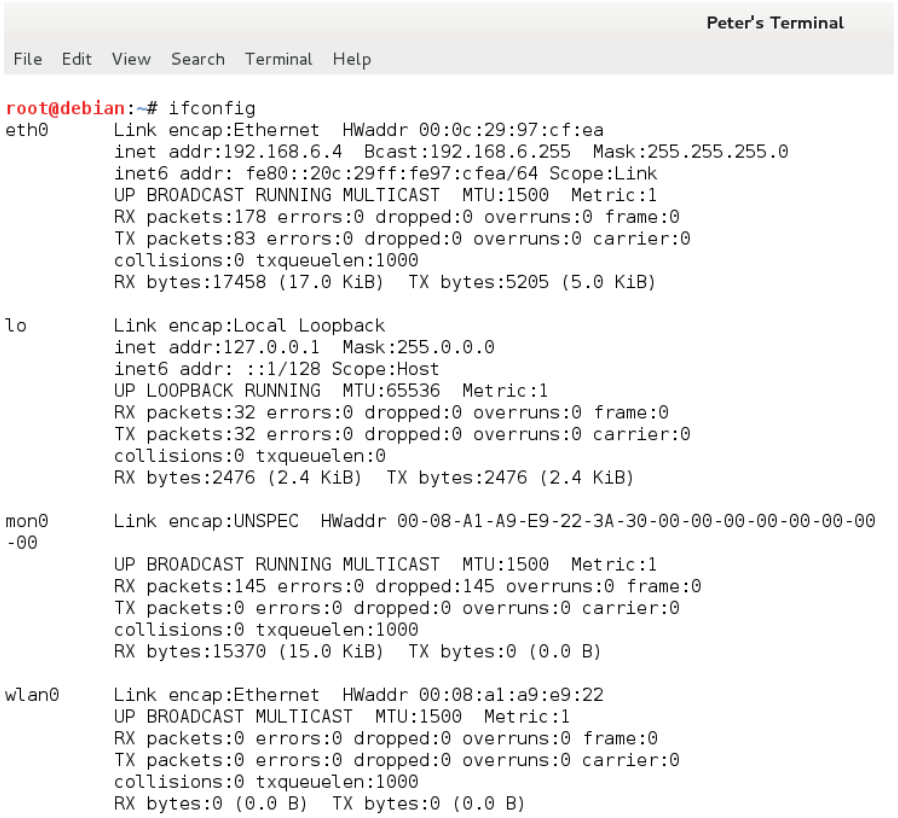

Figura B.5. Comprobación de la creación del modo monitor en la tarjeta inalámbrica.

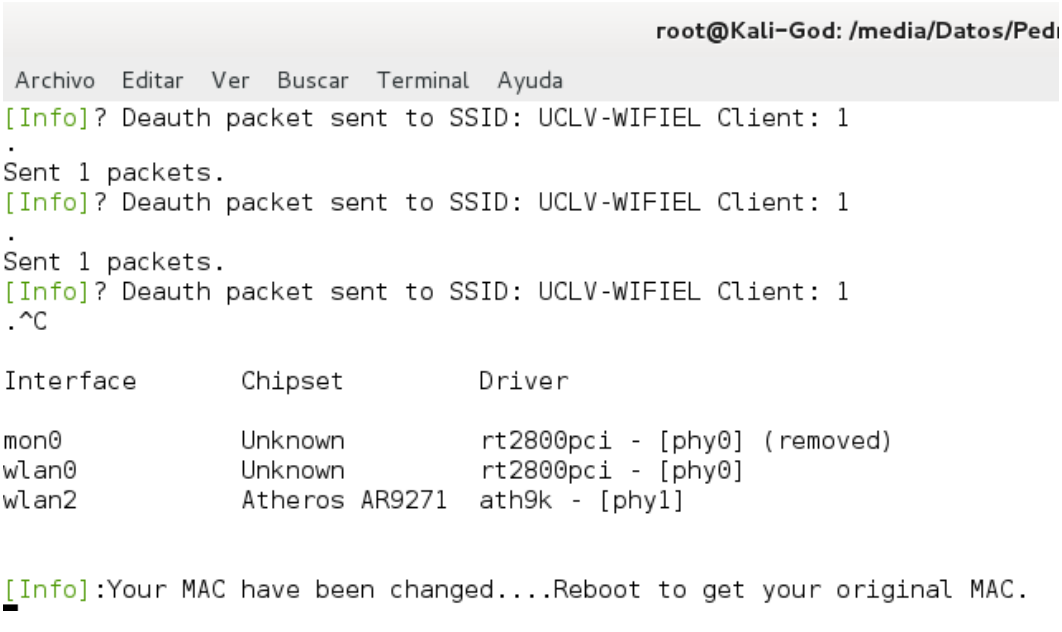

Figura B.6. Ataque de deautenticación y cambio de dirección MAC.

Figura B.7. Configuración del punto de acceso con encriptación WEP.

```
                                                    Peter's Terminal
File  Edit  View  Search  Terminal  Help

CH  3 ][ Elapsed: 8 s ][ 2014-09-29 16:43

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:14:D1:30:01:B0  -45        5        0    0   1  54   WEP  WEP         TESIS-ITURRIA

BSSID              STATION            PWR   Rate    Lost    Frames  Probe
```

Figura B.8. Salida del comando *airodump-ng*.

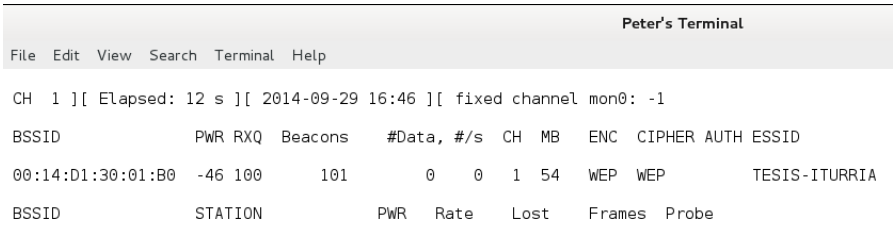

Figura B.9. Captura de paquetes con *airodump-ng*.

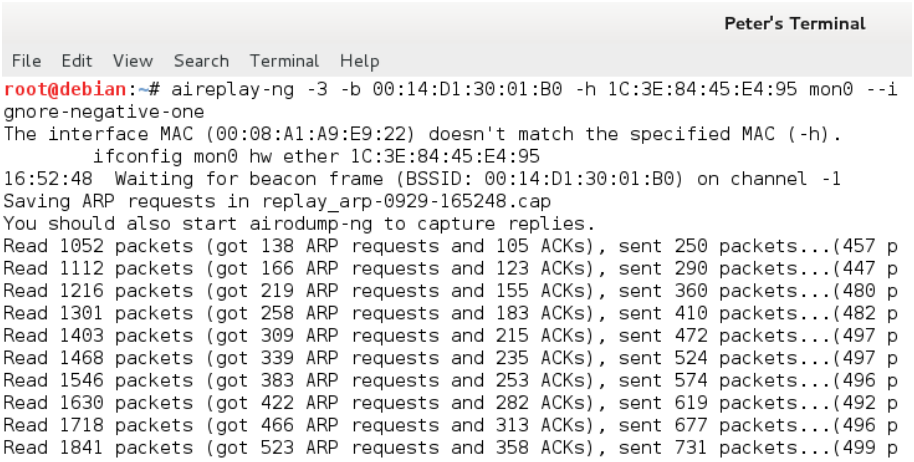

Figura B.10. Generación de tráfico con *aireplay-ng*

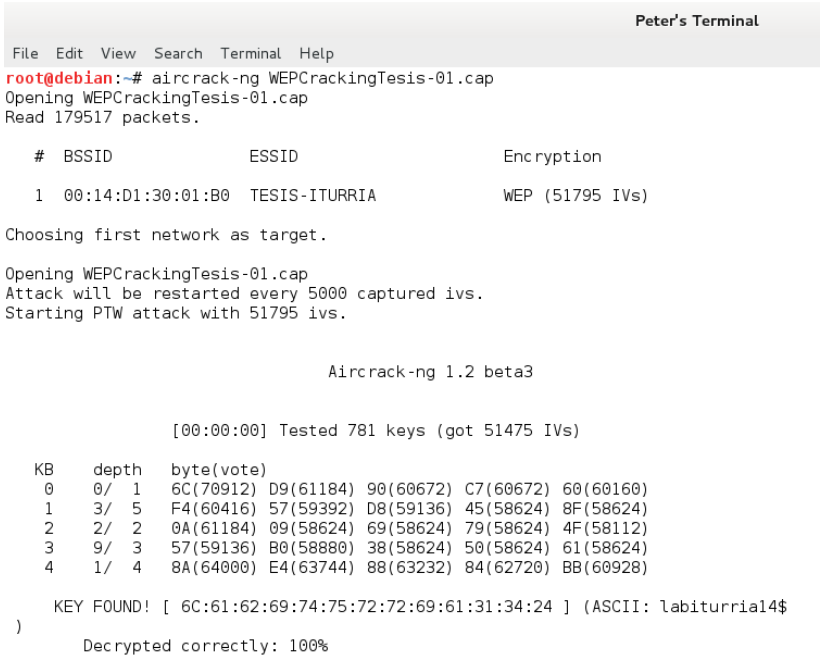

Figura B.11. Craqueo de encriptación WEP con *aircrack-ng*.

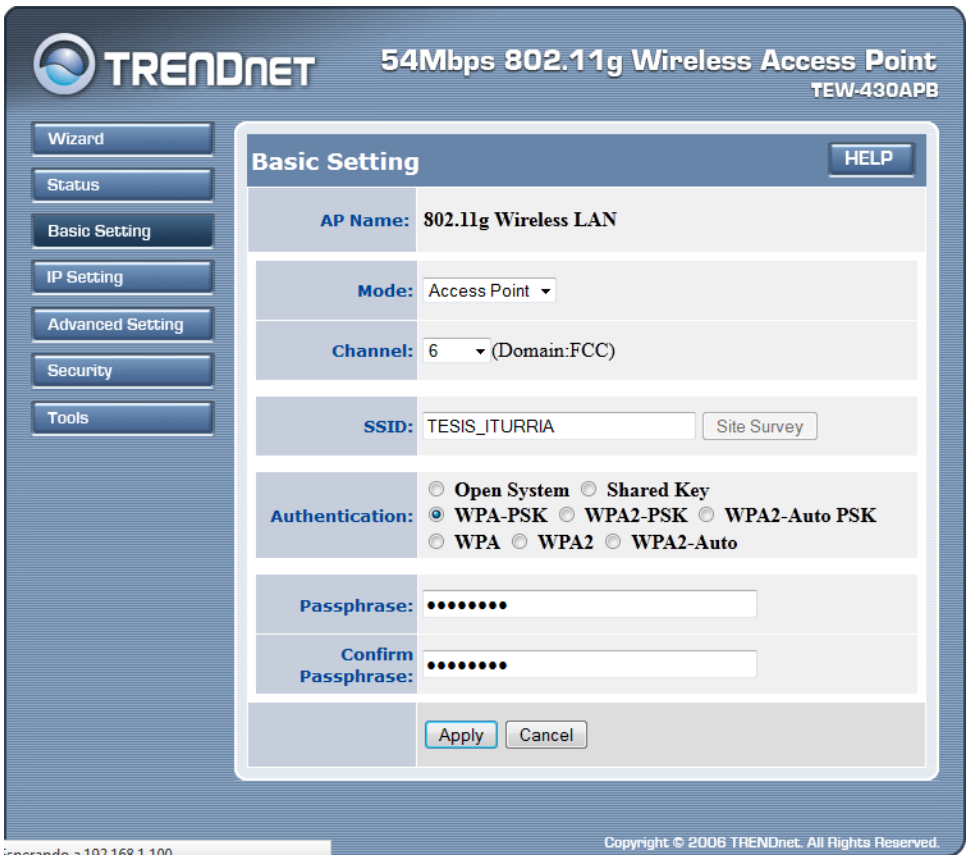

Figura B.12. Configuración del punto de acceso con autenticación WPA-PSK.

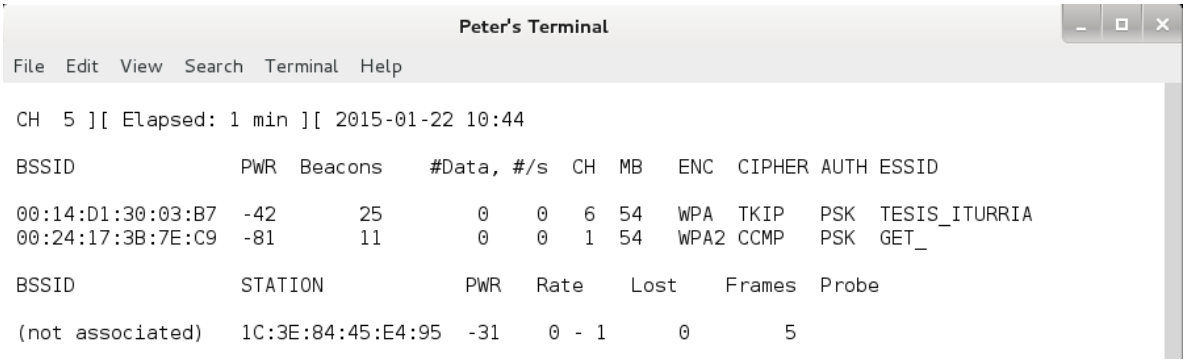

Figura B.13. Salida del comando *airmon-ng*.

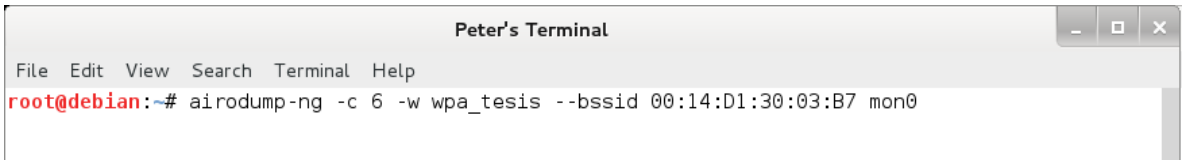

Figura B.14. Comando de ejecución de *airodump-ng*.

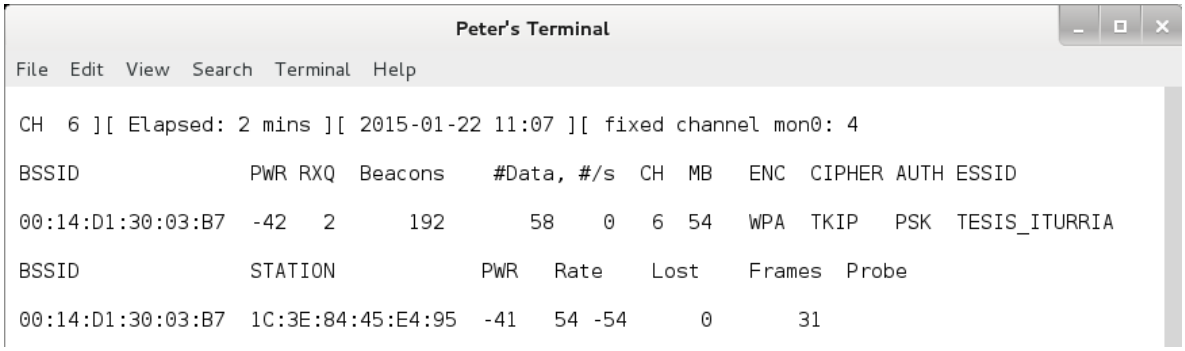

Figura B.15. Salida del comando de *airodump-ng*.

```
root@debian:~# aireplay-ng --deauth 10 -a 00:14:D1:30:03:B7 -c 1C:3E:84:45:E4:95 mon0 --ig
nore-negative-one
11:34:01  Waiting for beacon frame (BSSID: 00:14:D1:30:03:B7) on channel -1
11:34:02  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [16|15 ACKs]
11:34:02  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 0| 0 ACKs]
11:34:03  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 7|21 ACKs]
11:34:03  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 1| 1 ACKs]
11:34:04  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 3| 4 ACKs]
11:34:05  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 0| 1 ACKs]
11:34:05  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 7| 0 ACKs]
11:34:06  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 0| 1 ACKs]
11:34:07  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 0| 0 ACKs]
11:34:07  Sending 64 directed DeAuth. STMAC: [1C:3E:84:45:E4:95] [ 0| 0 ACKs]
```

Figura B.16. Deautenticación activa de clientes con *aireplay-ng*.

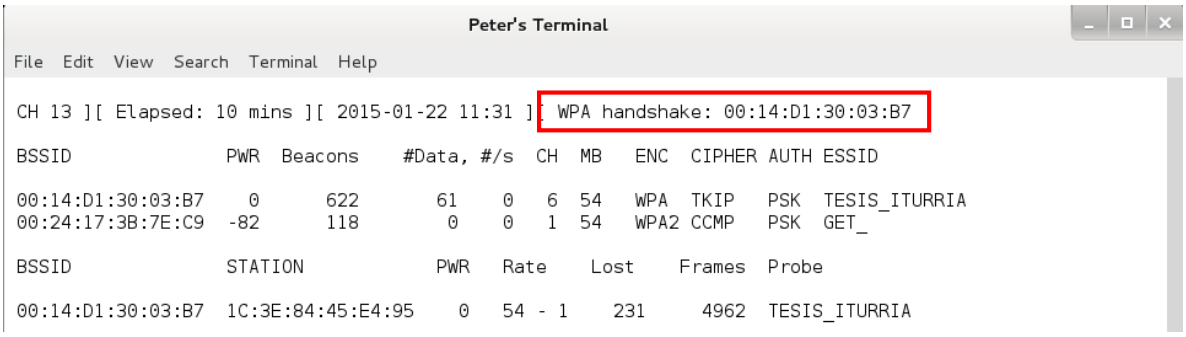

Figura B.17. Captura del *four-way handshake* con *airodump-ng*.

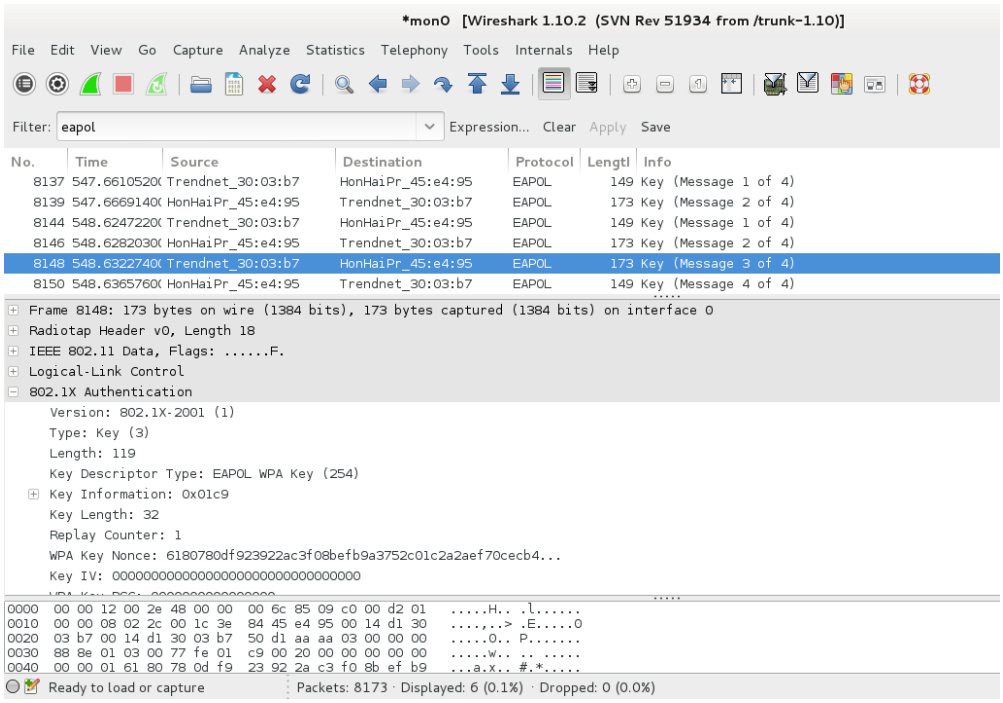

Figura B.18. Análisis del protocolo EAPoL en *Wireshark.*

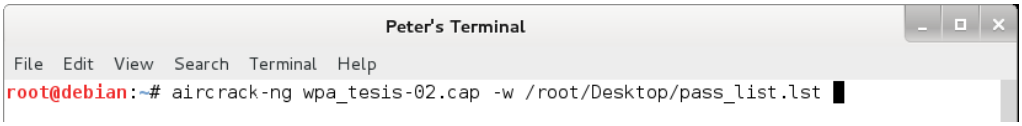

Figura B.196. Comando utilizado para la obtención de la clave WPA con *aircrack-ng.*

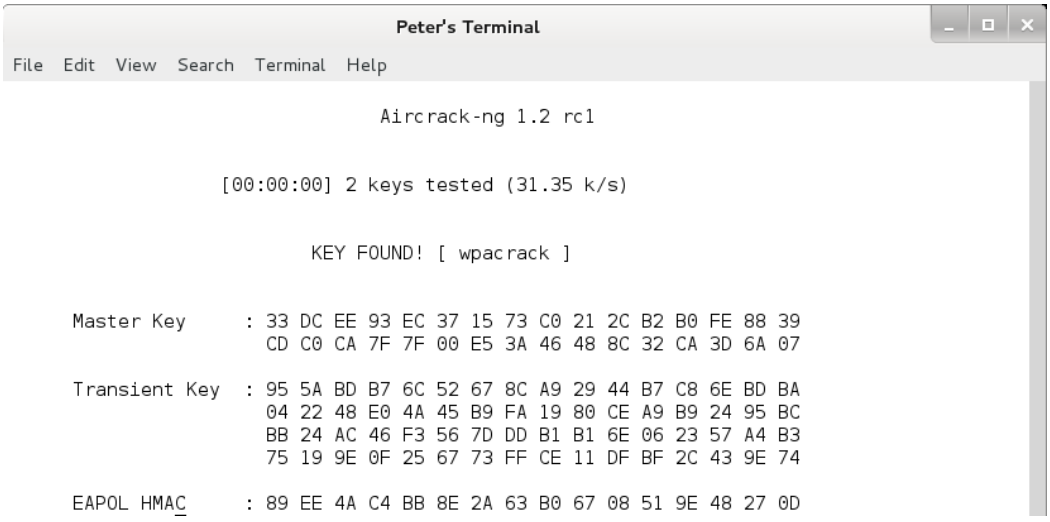

Figura B.20. Ataque exitoso con a*ircrack-ng.*

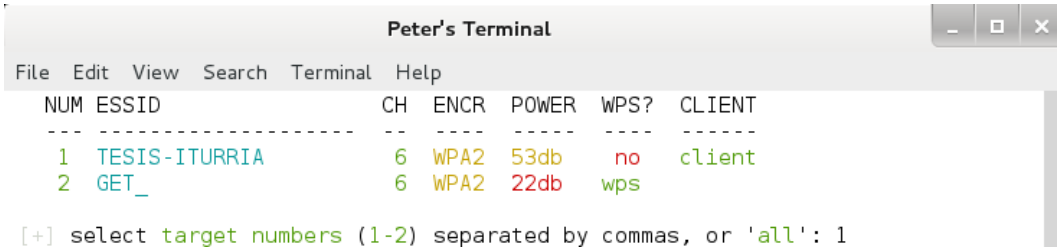

Figura B.21. Resultado del escaneo de redes con *wifite*.

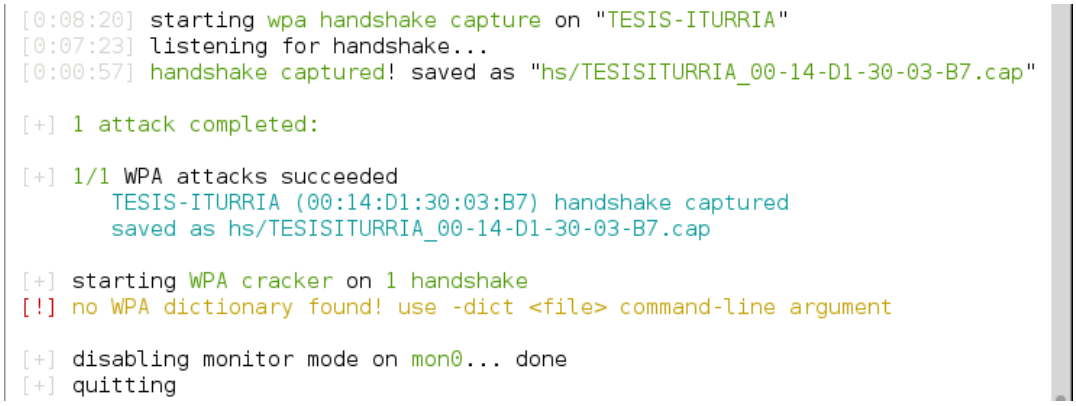

Figura B.22. Obtención del *four-way handshake* mediante *wifite*.

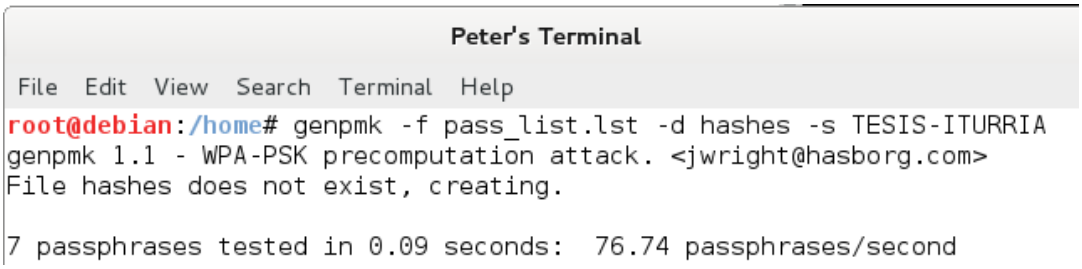

Figura B.23. Creación de la *rainbow table* con el comando **genpmk**.

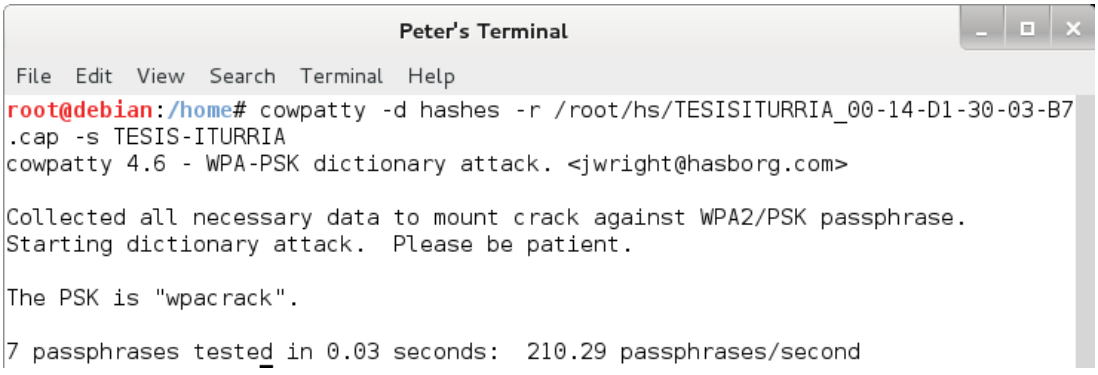

Figura B.24. Craqueo exitoso con *coWPAtty*.

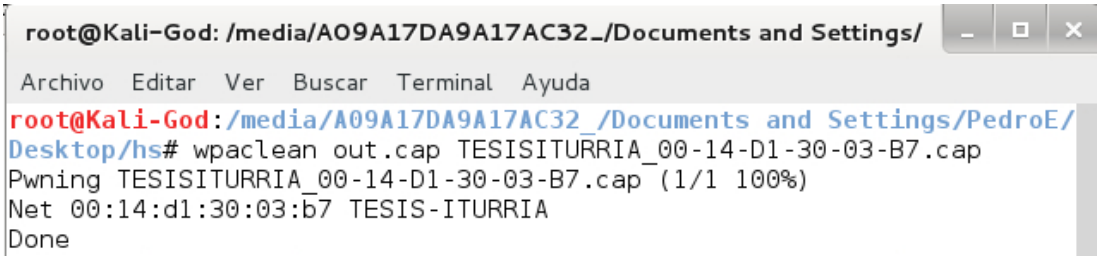

Figura B.25. Limpieza del fichero **.cap** con *wpaclean*.

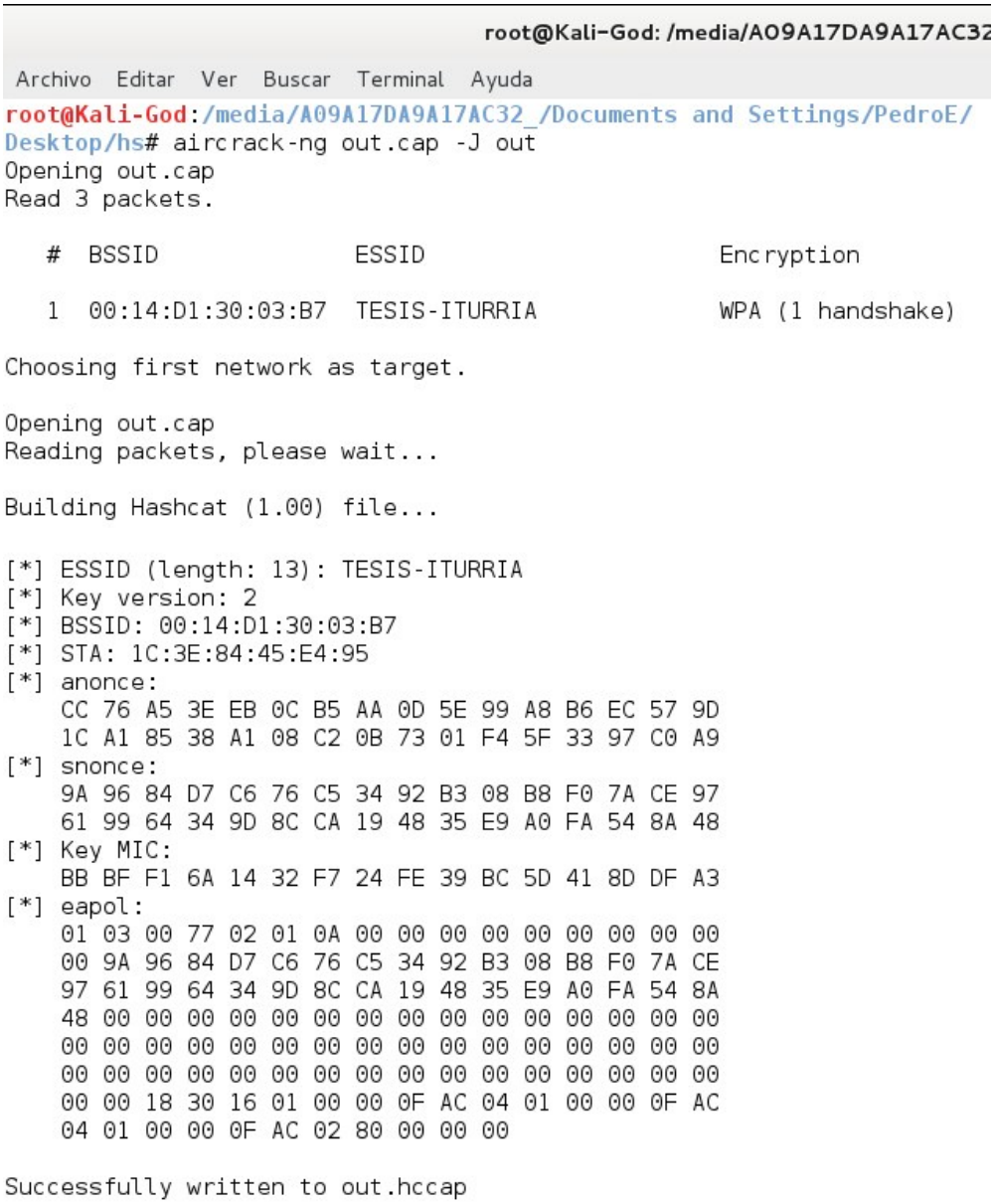

Figura B.26. Conversión del archivo **.cap** al formato **.hccap** con *aircrack-ng*.

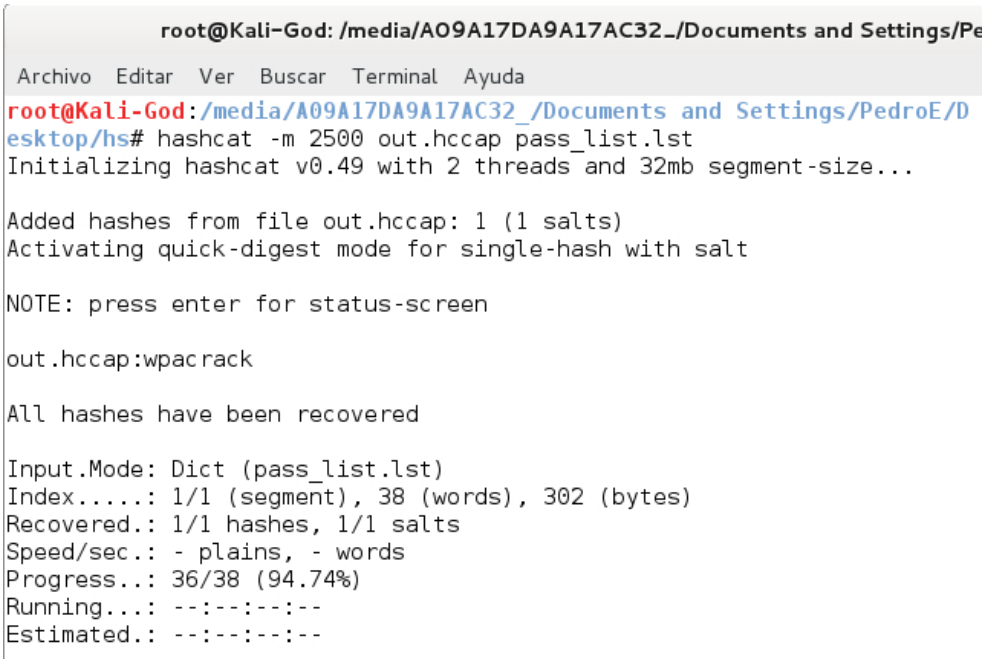

Figura B.27. Ataque de fuerza bruta con *hashcat* y lista de palabras.

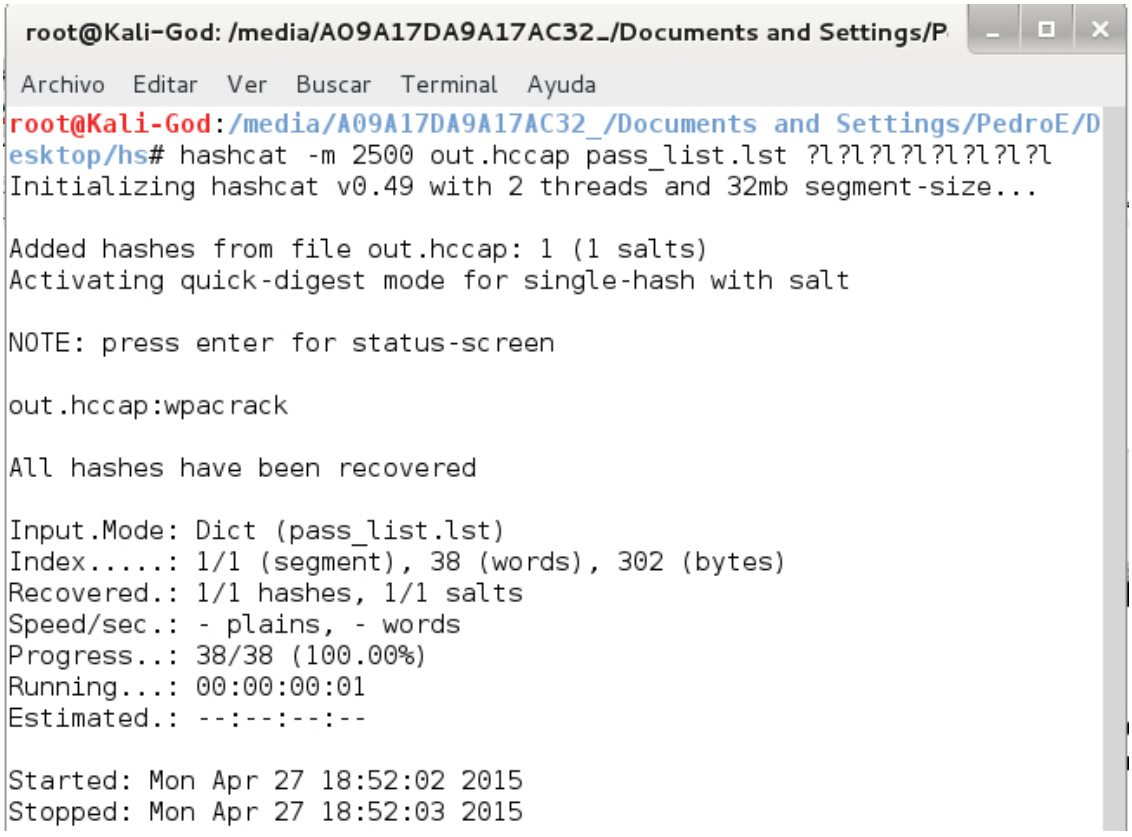

Figura B.28. Ataque de fuerza bruta con *hashcat* y máscara de palabras.

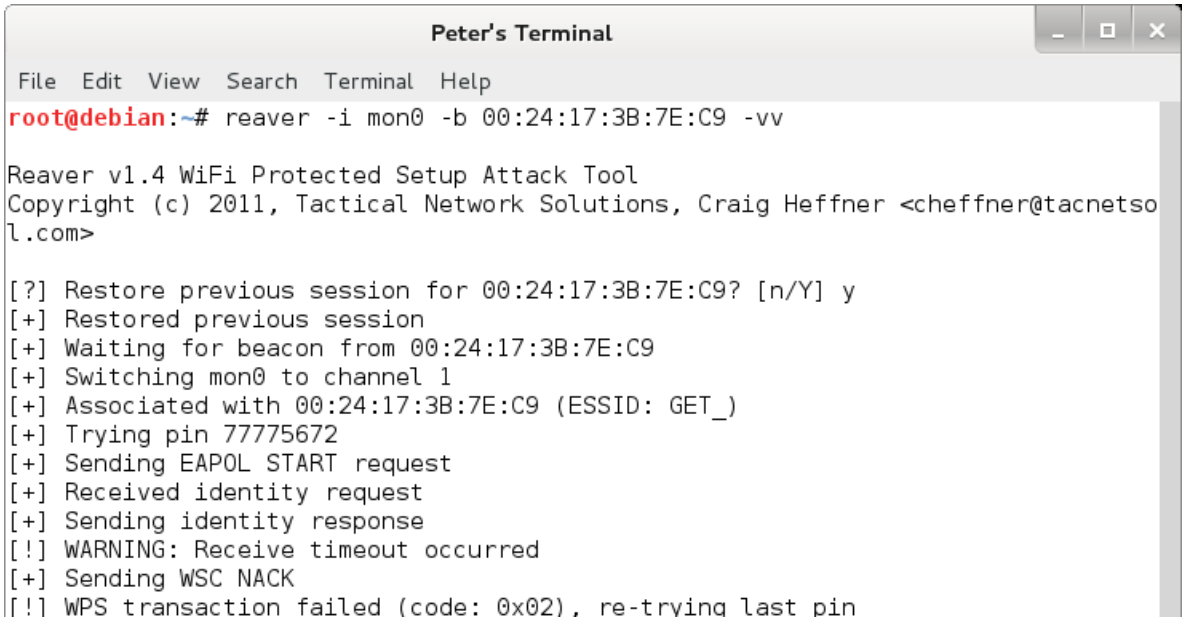

Figura B.29. Ataque a WPS con *reaver*.

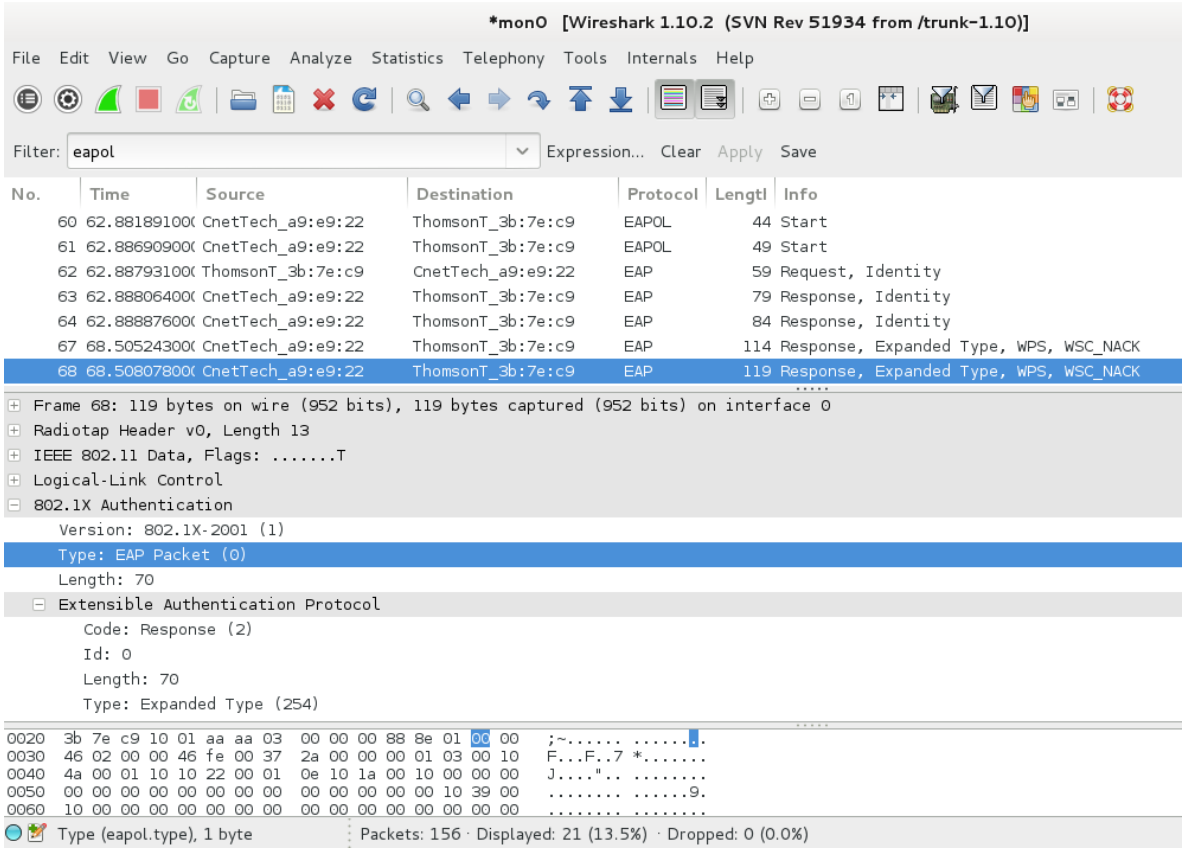

Figura B.30. Análisis del ataque WPS con *Wireshark*.

Figura B.31. Interfaz gráfica de *Fern Wifi Cracker*.

Figura B.32. Panel de ataque de *Fern Wifi Cracker*.

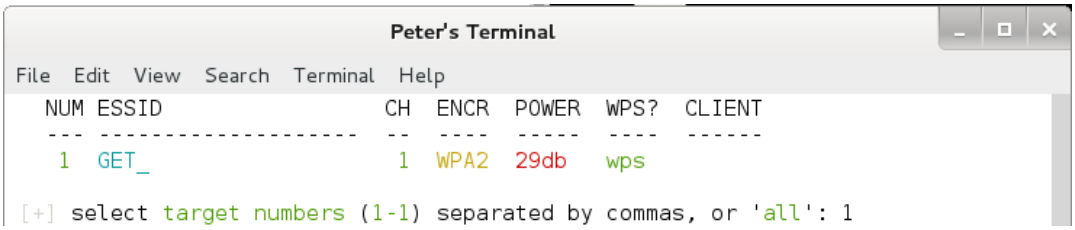

Figura B.33. Ataque a WPS con *wifite*.

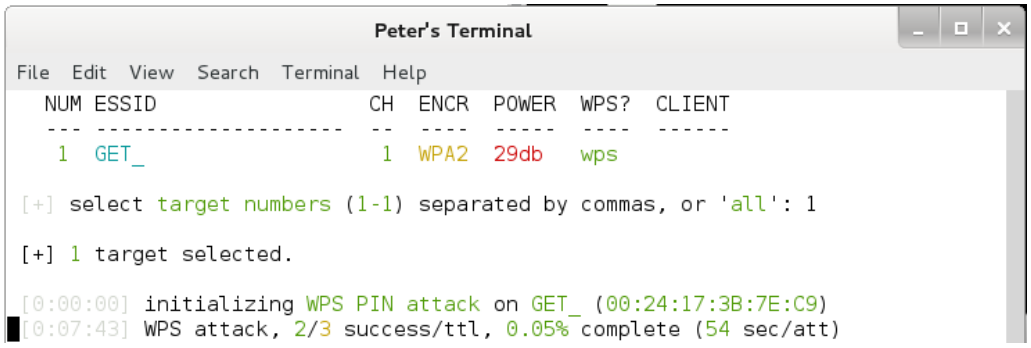

Figura B.34. Ataque a WPS en progreso con *wifite*.

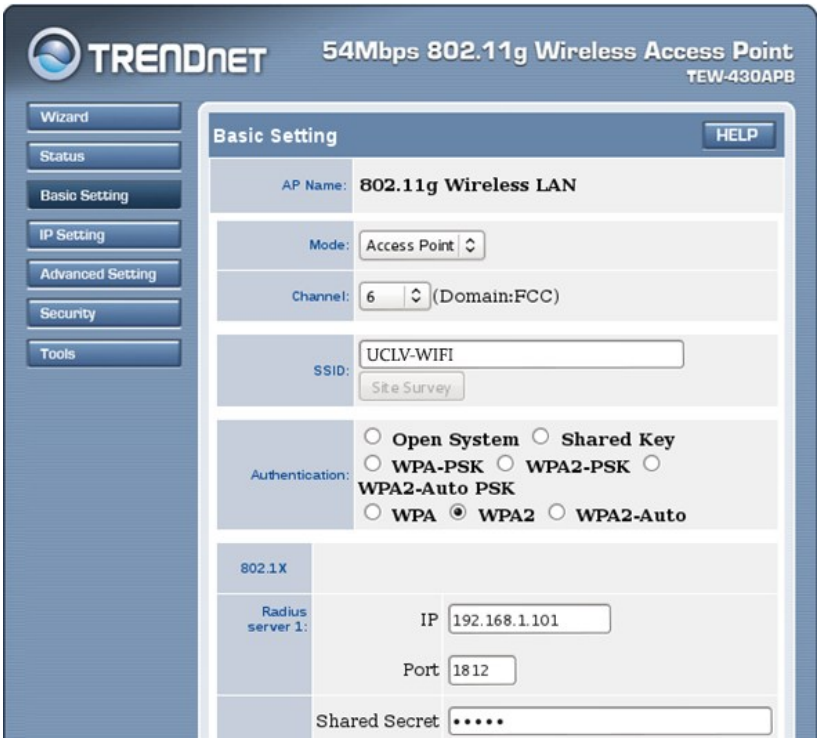

Figura B.375. Interfaz web del AP *Trendnet* TEW-430APB. Configuración 802.1X.

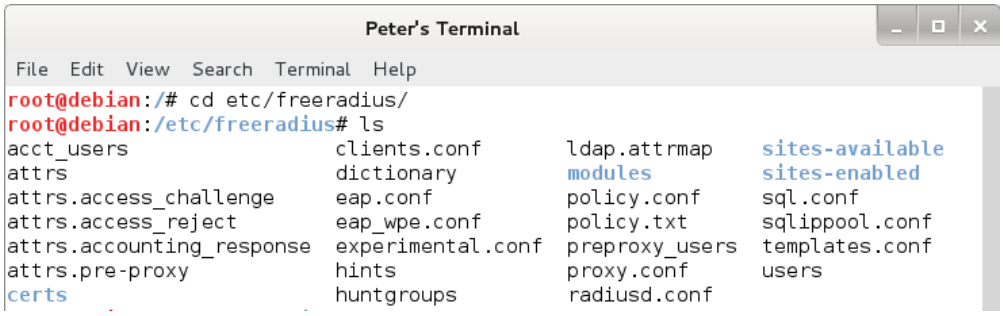

Figura B.36. Directorio raíz de *Freeradius*.

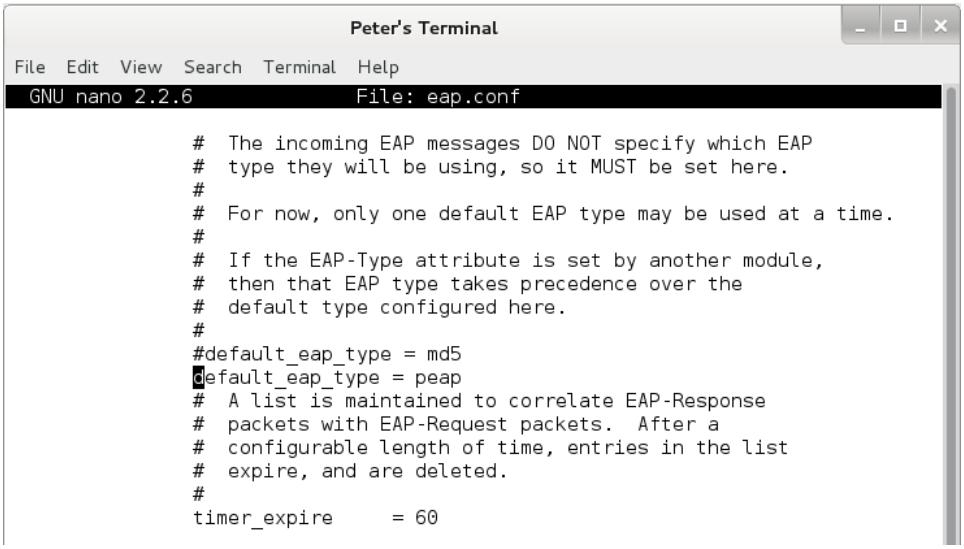

Figura B.37. Archivo de configuracíón **eap.conf**.

Figura B.38. Archivo de configuración **clients.conf**.

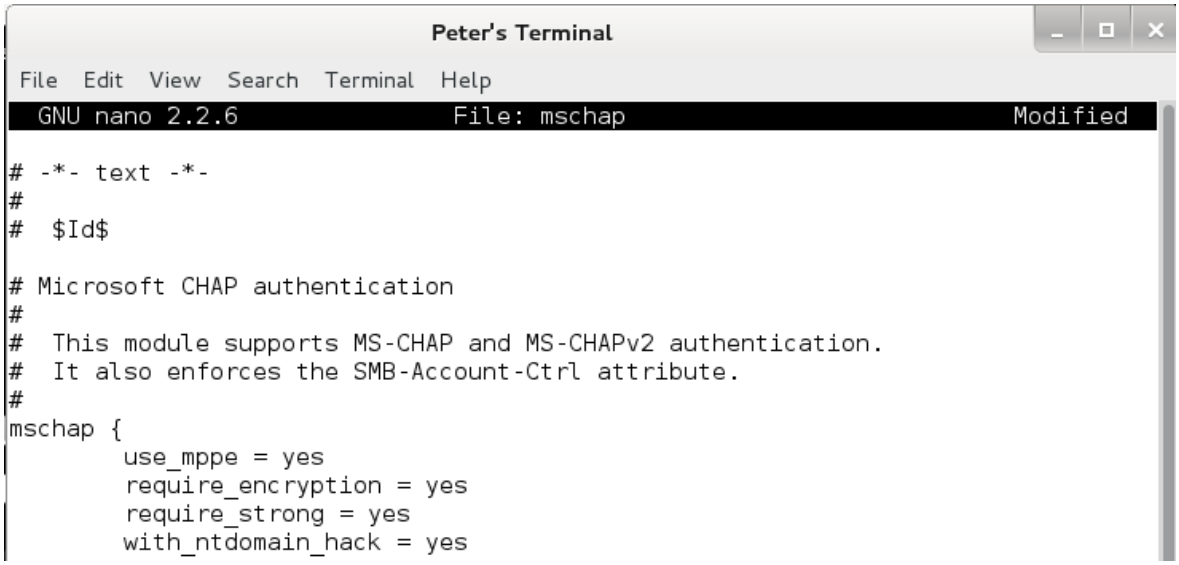

Figura B.39. Archivo de configuración del módulo mschap.

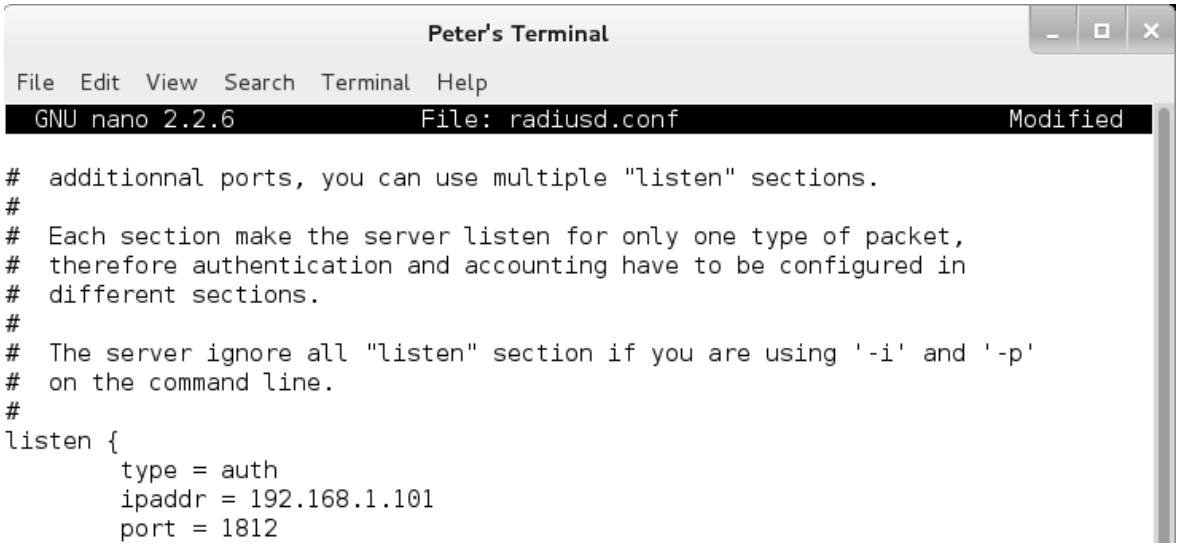

Figura B.40. Archivo de configuración **radiusd.conf**.

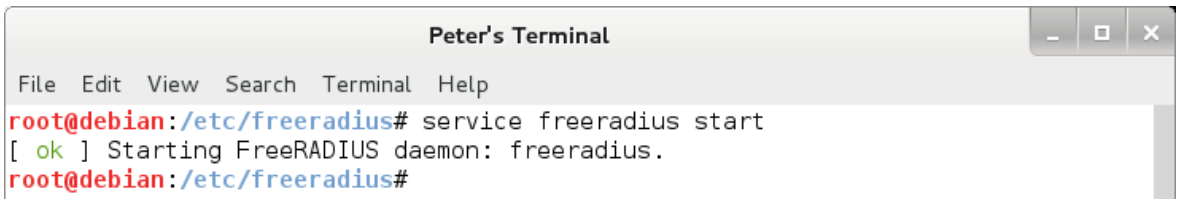

Figura B.41. Comando de encendido del servicio de servidor *Radius*.

```
root@debian:/etc/freeradius# radtest peter peter 192.168.1.101 1812 tesis
Sending Access-Request of id 31 to 192.168.1.101 port 1812
        User-Name = "peter"
        User-Password = "peter"
        NAS-IP-Address = 127.0.1.1
        NAS-Port = 1812
        Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Accept packet from host 192.168.1.101 port 1812, id=31, length=
20
```

Figura B.42. Prueba del correcto funcionamiento del servidor *Radius*.

Figura B.43. Redes inalámbricas identificadas por el usuario.

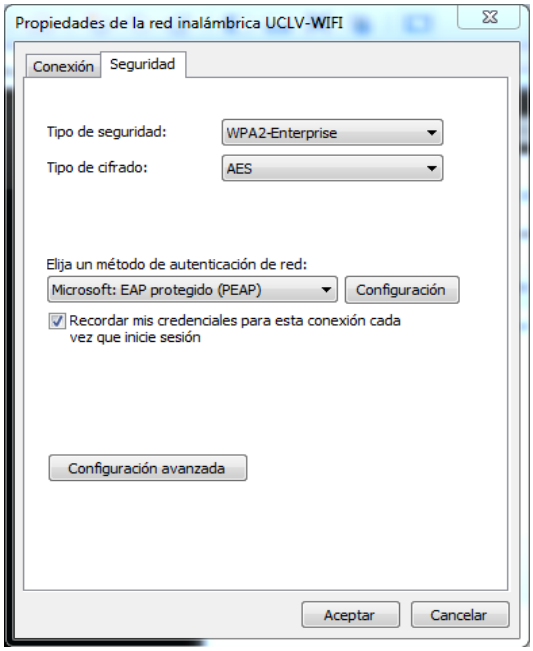

Figura B.44. Propiedades de la red inalámbrica UCLV-WIFI en la PC víctima.

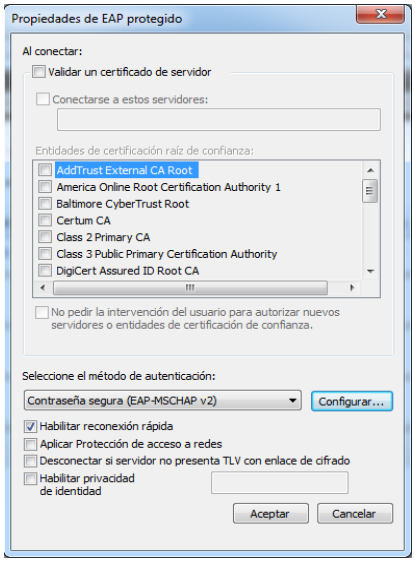

Figura B.45. Propiedades de EAP protegido.

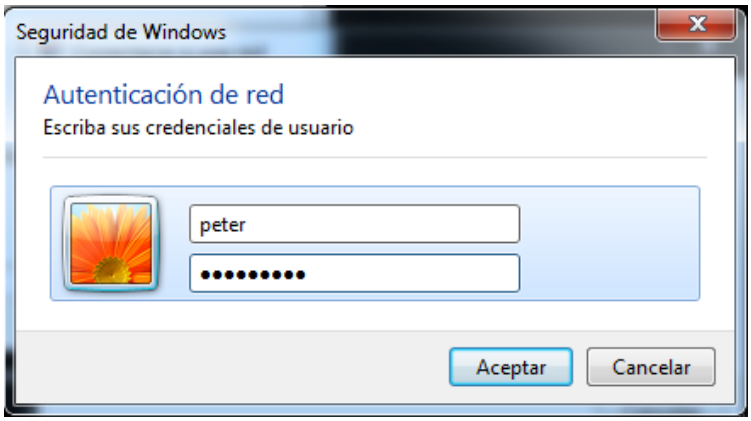

Figura B.46. Ventana de autenticación del usuario en *Windows 7 Ultimate*.

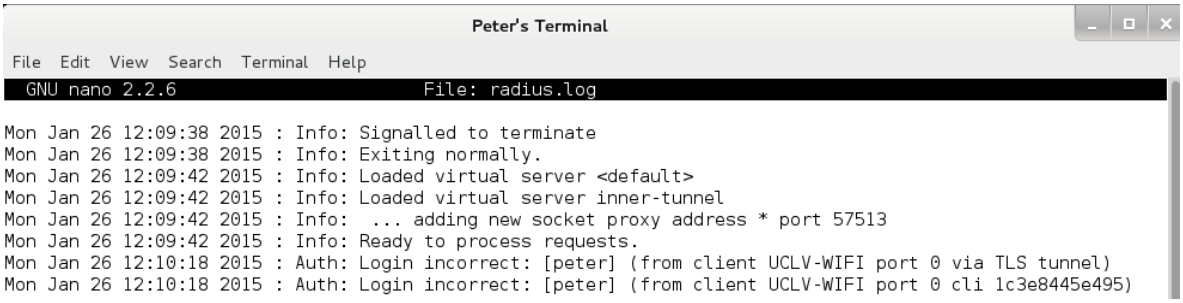

Figura B.47. Archivo de registros (*logs*) **radius.log**.

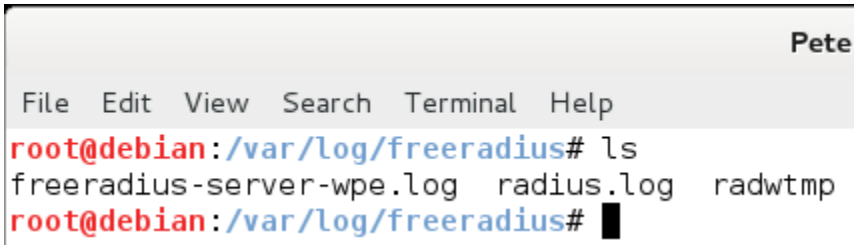

Figura B.48. Directorio de los archivos de registros del servidor *Radius*.

```
root@debian:/var/log/freeradius# tail -f freeradius-server-wpe.log -n 7

mschap: Mon Jan 26 12:10:18 2015

        username: peter
        challenge: 39:d0:ac:57:40:11:a1:ce
        response: 50:ad:df:95:fc:39:2d:06:38:3f:43:a1:12:5a:39:24:aa:30:00:ac:98:65:89:9d
```

Figura B.49. Resultado de la autenticación en el archivo de registros **freeradius-server-wpe.log.**

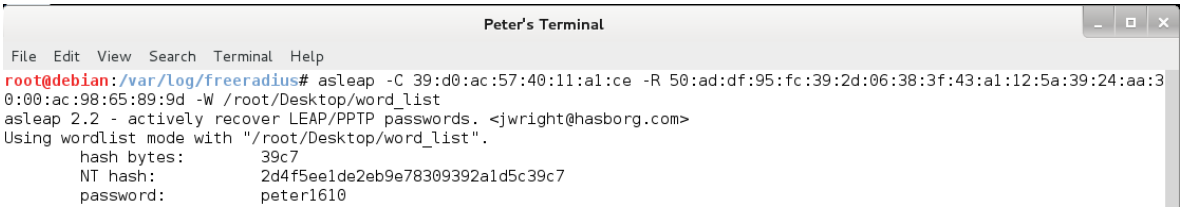

Figura B.50. Obtención de contraseñas con la herramienta *asleap*.

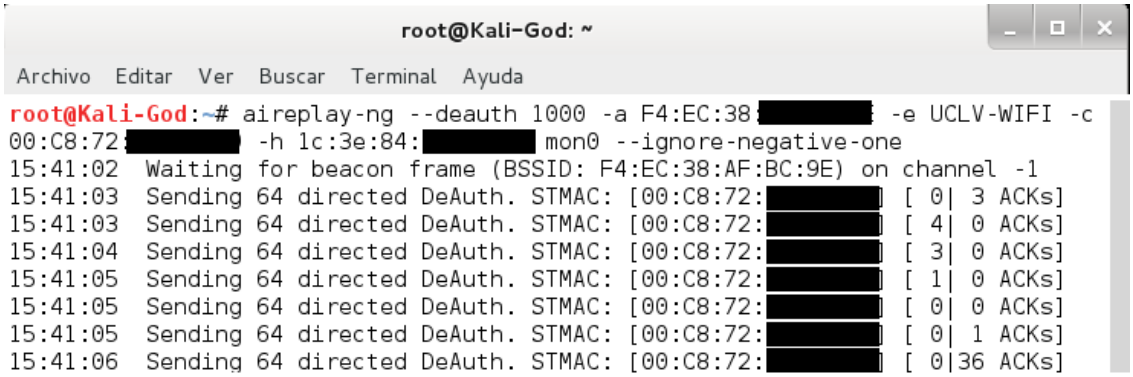

Figura B.51. Ataque de denegación de servicio con *aireplay-ng*.

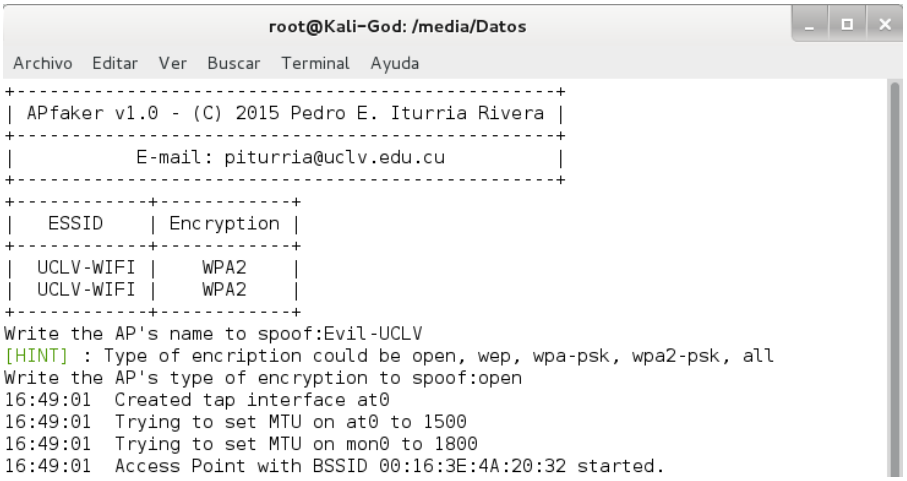

Figura B.52. Creación de un falso AP con *APfaker* v1.0.

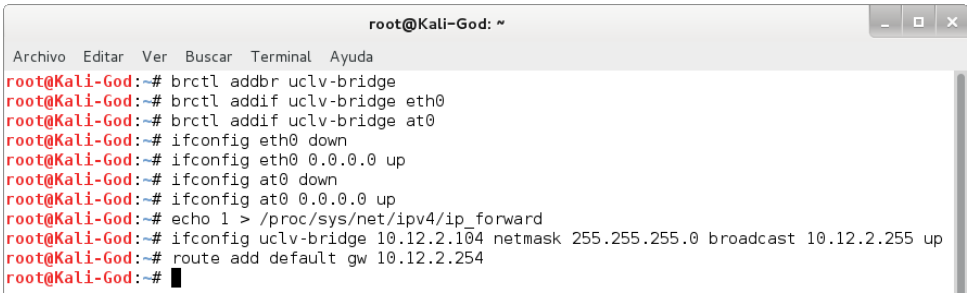

Figura B.53. Creación del puente entre la interfaz **at0** y **eth0**.

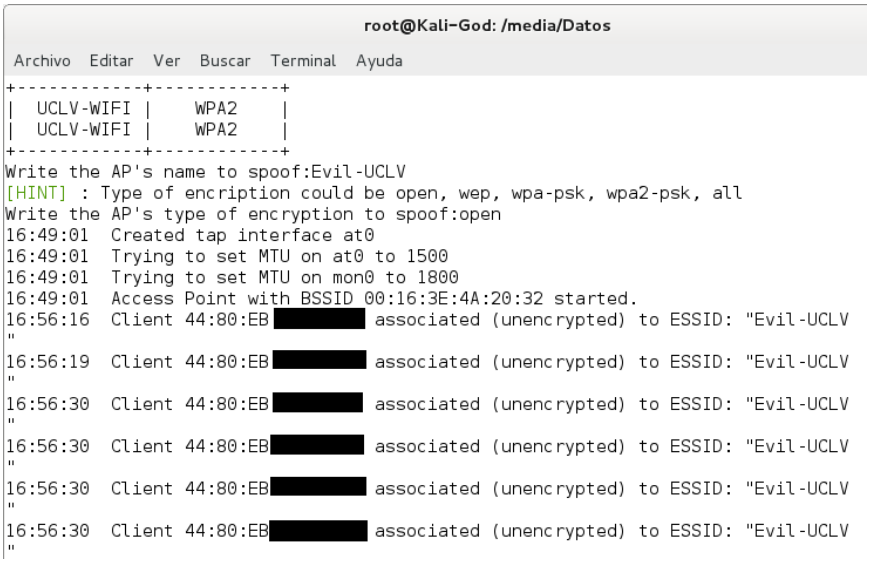

Figura B.54. Asociación del cliente al AP falso en el ataque MITM.