Plan de alertas basado en indicadores de compromiso para dispositivos smart switch/plug en IoT

Alexander Ortiz Restrepo

Instituto Tecnológico Metropolitano

Facultad de Ingeniería

Medellín, Colombia

2020

Plan de alertas basado en indicadores de compromiso para dispositivos smart switch/plug en IoT

Alexander Ortiz Restrepo

Tesis o trabajo de investigación presentada(o) como requisito parcial para optar al título de:

Magister en Seguridad Informática

Director (a):

Magister Milton Javier Mateus Hernández

Línea de Investigación:

Seguridad en redes e IoT

Instituto Tecnológico Metropolitano

Facultad de Ingenierías

Medellín, Colombia

2020

(Dedicatoria o lema)

“Hazlo por la anécdota”

Agradecimientos

Este trabajo fue realizado gracias al apoyo incondicional y constante de Héctor Vargas, gracias por su paciencia, ayuda y acompañamiento, estoy infinitamente agradecido.

Gracias a mi asesor de tesis Milton Javier Mateus Hernández por su aporte durante este largo proceso.

Gracias a mi gran amigo Gabriel por siempre estar ahí conmigo.

Resumen

En el mercado actual existen a la venta muchos dispositivos IoT con múltiples fallas de seguridad y poca o cero protección ante ataques cibernéticos [6]. Se investigó sobre los dispositivos IoT vendidos para el hogar y se encontró que los dispositivos bajo la categoría de smart switch son bastante comunes y adquiridos. La investigación se enfocó en encontrar mecanismos de detección y alerta para ataques informáticos en los dispositivos IoT.

Mediante una división clave en el desarrollo de la metodología se logró realizar la totalidad de la investigación. La investigación se dividió en 5 objetivos específicos, buscando proponer e implementar un sistema de alertas para el usuario final ante ataques informáticos en su dispositivo Smart switch. Primero fue lo más básico y era encontrar un dispositivo IoT que fuera común en este tipo de hogares, un dispositivo que fuera básico para el usuario promedio y fácil te utilizar. Con estos criterios se determinó escoger un smart switch/plug inteligente.

Una vez encontrado el dispositivo, se buscó entre las marcas comerciales, cuál sería el más apropiado para la investigación, con unos estudios de consumo se determinó que la marca Sonoff disponía de un smart switch popular, económico y fácil de usar. Lo siguiente fue investigar en el entorno IoT sobre los ataques informáticos que aplicaban a este tipo de dispositivo, conocer cuáles eran comunes y replicables, luego se decidió investigar sobre los ataques de tipo denegación de servicio.

Posteriormente con estos ataques se investigaron y se crearon sus respectivos indicadores de compromiso con los cuales se conoció el alcance y el comportamiento de cada ataque. Una vez conocido este comportamiento malicioso se procedió con realizar una arquitectura de seguridad que me generara alertas en la red cuando se encontrara con uno de estos comportamientos ya conocidos anteriormente. Esta arquitectura fue realizada gracias a un software de detección de intrusos y para esta investigación se eligió snort.

Con la realización de esta investigación, se creó un sistema que le notifica al usuario final sobre un incidente de seguridad informática en su dispositivo IoT mediante una alerta electrónica. El usuario final dispone de una documentación guía, la cual puede seguir para evitar el ataque,

adicionalmente allí dispone de buenas prácticas para el tratamiento de los incidentes de ciber seguridad.

Palabras clave: Indicadores de compromiso (IoC), Internet de las cosas (IOT), Smart switch/plug, Snort, Swatch, Ubuntu, Respuesta incidentes

Abstract

In today's marketplace there are many IoT devices for sale with multiple security flaws and little or no protection against cyber-attacks [6]. In the research it was found that a specific kind of IoT home device was popular, known as a smart switch/plug, this smart switch is quite common and bought by customers. The research was focused on finding detection and alert mechanisms for cyber-attacks on this IoT device.

Through a key division in the development of the methodology, the entire investigation was carried out. The first thing was the most basic and it was to find an IoT device that was common in a modern IoT home, a device that was basic for the average user and easy to use. With these criteria I settled to investigate in a smart switch.

Once the device was found, we searched among the brands which would be the most appropriate for the investigation, with some consumer studies it was determined that the Sonoff brand had a popular, economic and easy to use smart switch. The next thing was to investigate in the IoT environment about the computer attacks that are applied to this type of device, to know which were common and replicable, it was investigated on attacks categorized as denial of service attacks.

Subsequently, with these attacks, the indicators of compromise were investigated and created, with which the scope and behavior of each attack was known. Once this malicious behavior is known, it was processed by implementing a security architecture that generates alerts on the network when it encounters one of these behaviors previously known. This architecture was made thanks to an intrusion detection software and “snort” was chosen for this investigation.

By conducting this investigation, a system was created that notifies the end user of a cyber security incident on their IoT device. The end user is given documentation so he can handle the situation, additionally it has included best practices for this type of incidents.

Keywords: Indicators of compromise, Internet of things, Smart switch/plug, Swatch, Snort, Ubuntu, Incident response

Contenido

Pág.

Lista de figuras

Pág.

Lista de tablas

Pág.

Lista de Abreviaturas

Abreviatura	Término
ARP	Address Resolution Protocol
CVSS	Common Vulnerability Scoring System
DNS	Domain Name Server
DoS	Denial of service
FTP	File transfer protocol
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IoC	Indicator of compromise
IoT	Internet of things
IoT	Internet of things
IP	Internet Protocol
LAN	Local Area Network
LT	Primera ley de la termodinámica
MAC	Media access control
MITM	Man in the middle
OSI	Open Systems Interconnection Model
RAE	Real academia española
RFF	Racimos de fruta fresca
SSH	Secure Shell Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
URL	Uniform Resource Locator

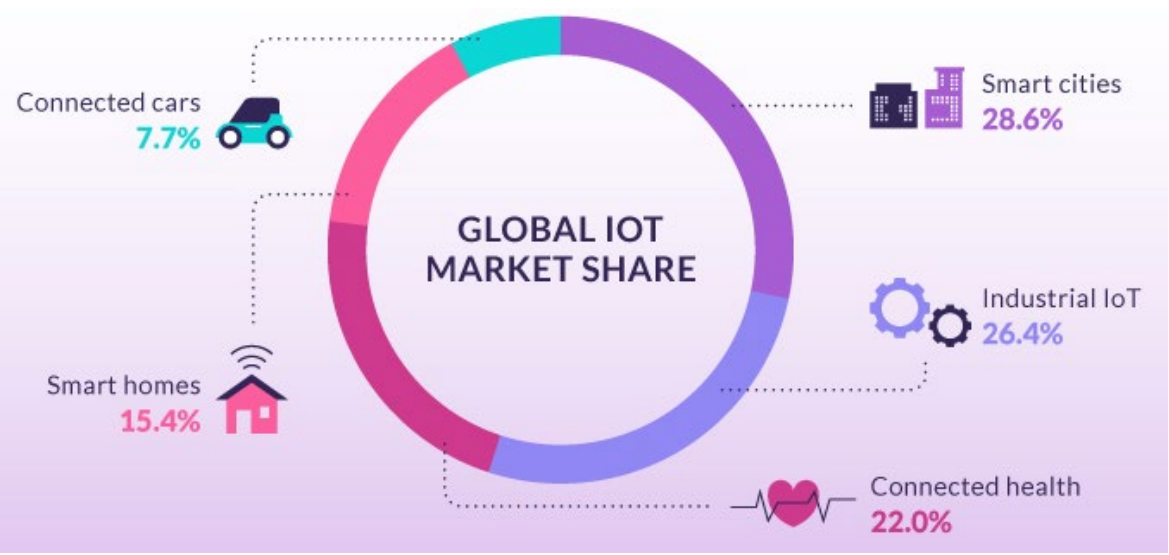
WAN	Wide area network
WEP	Wired Equivalent Privacy
WPA	Wired Protected Access
WPA2	Wired Protected Access 2

Introducción

El internet de las cosas, o IoT (Internet of things) es el método por medio del cual diferentes dispositivos de uso diario, como nuestras lavadoras, vehículos, hogares, sensores, y hasta relojes se conectan al internet. El internet de las cosas se ve en los dispositivos modernos que se conectan hacia el internet para poder recibir y enviar información a este. La razón por la cual conectamos nuestros objetos cotidianos hacia el internet es sencilla, por conveniencia y automatización. Tener la capacidad de encender la lavadora desde la distancia, poder registrar nuestro ritmo cardiaco en una aplicación, revisar el estado del vehículo sin estar dentro de él, poder revisar la seguridad de del hogar sin tener que estar allí y hasta apagar nuestro fogón así no estemos en el hogar [1].

En este sentido, para el 2020 se tienen más de 10 billones de dispositivos IoT conectados hacía el internet [2]. Ahora, los dispositivos que componen el entorno IoT actualmente se clasifican en 5 categorías: Vehículos conectados, ciudades inteligentes, hogares inteligentes, salud conectada, y el sector de la industria inteligente.

Figura 1: Categorías IoT



Fuente: Safeatlast [1]

De acuerdo con la **Figura 1**, el sector donde más se utiliza la tecnología IoT es en las ciudades inteligentes. Por ejemplo, en la ciudad de Medellín se cuenta con tecnología IoT en los semáforos,

los cuales permiten que se determinen zonas de alta congestión para agilizar el tráfico [3]. Las ciudades inteligentes equivalen a 2.3 billones de dispositivos conectados actualmente hacía el internet observado en la figura 1. Allí se encuentran dispositivos para monitorear los lugares de parqueaderos disponibles, semáforos inteligentes, luces viales inteligentes y hasta calles con señalización inteligente según la situación. Para acotar el mercado y el alcance tan amplio que tiene el tema de IoT se concentró en la categoría de los hogares inteligentes. Los dispositivos IoT en los hogares inteligentes componen el 15.4% o 1.3 billones en el mercado global.

Por consiguiente, se generan interrogantes como ¿Qué dispositivos componen un hogar inteligente? Lo más esencial es tener una consola central (asistente inteligente) como por ejemplo un dispositivo “Amazon Alexa” o “Google Assistant”, incluso la marca de Apple también lanzó su asistente para el hogar “Siri” [4]. Estos dispositivos cumplen con funciones muy similares, son dispositivos operados por la voz del usuario donde se le dan instrucciones que debe obedecer. La idea de estas consolas es que el usuario pueda tener acceso mediante la voz a los televisores inteligentes, parlantes inteligentes y hasta a la nevera inteligente. Otros componentes importantes que componen un hogar inteligente son: las luces inteligentes, un sistema de seguridad para alertar contra intrusos, cámaras de vigilancia, aire acondicionado Smart y lo más sencillo de tener es un switch o plug inteligente. Por lo tanto, el dispositivo elegido para la investigación también debería tener compatibilidad con las consolas centrales del hogar.

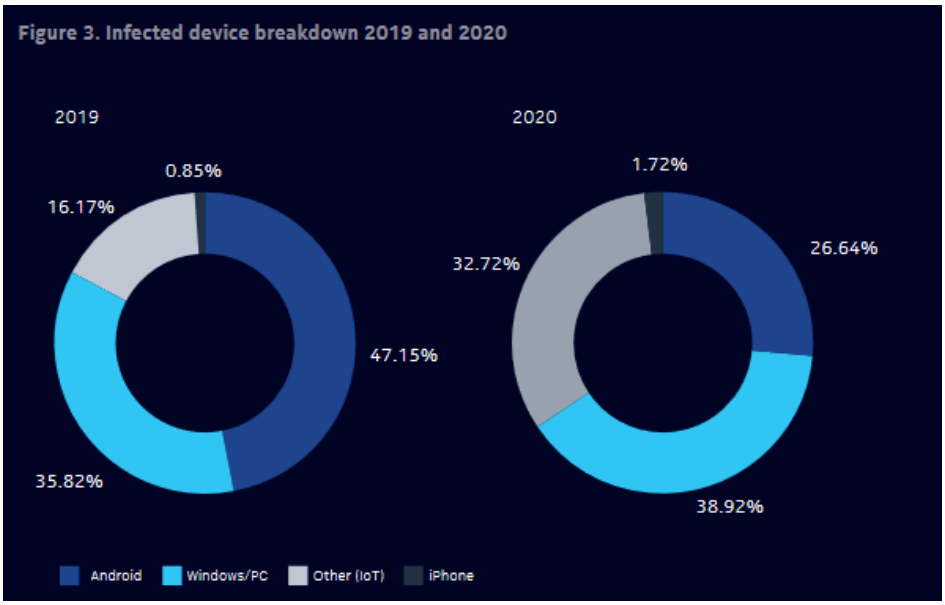
De tal manera, la problemática en IoT radica en que los fabricantes de estos dispositivos no están incorporando buenas prácticas de seguridad en sus dispositivos, estos dispositivos actualmente los venden muy vulnerables. Investigadores de la Universidad Ben-Gurion del Negev en Israel analizaron 16 dispositivos diferentes de IoT y encontraron que incluso los dispositivos críticos como cámaras inteligentes o timbres de puerta no eran difíciles de hackear. "Debido a que estos dispositivos son fabricados en masa, puedes comprar una cámara específica y descubrir sus debilidades, luego usarla contra cualquier otra persona con la misma cámara", dijo el investigador principal, Omer Shwarts [5]. Los dispositivos que Shwartz probó tenían contraseñas para acceder a la configuración, pero no siempre eran seguras. Shwartz dijo que el dispositivo más fácil de hackear tenía "1234" como contraseña, y el dispositivo más difícil de hackear, era un monitor para bebés, solo tardó dos días en vulnerar su seguridad. "Creo que soy la única persona que conoce la

contraseña, por lo que no voy a compartirla", dijo Shwartz, "pero es preocupante que haya sido tan fácil" [5].

Un reporte realizado por Avast sobre los hogares inteligentes revela que dos de cada cinco hogares digitales (40.8%) en todo el mundo contienen al menos un dispositivo que es vulnerable a los ataques cibernéticos. De estos, 69.2% son vulnerables debido a credenciales débiles, y 31.4% debido a vulnerabilidades de software. Con solo tener un dispositivo vulnerable conectado a internet este pone en riesgo todo el hogar inteligente [6]. Una pareja residente de la ciudad de Chicago, Estados Unidos, reportó una intrusión a su hogar inteligente por parte de un hacker. La pareja relató que era una noche tranquila, cuando estaban por dormirse escucharon la voz de un hombre desconocido hablándole por medio del monitor de bebés a su hijo de 7 meses. De inmediato se llevaron al niño de la habitación, pero también notaron que su termómetro inteligente había sido alterado toda vez la temperatura de la habitación fue modificada de 72 grados hasta los 90 grados. Luego la voz a través de los parlantes le decía a la pareja que dejara de moverse y que no miraran a la cámara de seguridad. Allí fue cuando se dieron cuenta que también su cámara de vigilancia fue vulnerada [7].

De hecho, un estudio publicado por Palo alto en el 2020, llegaron a la conclusión de que el 98% del tráfico en los dispositivos IoT no está encriptado y que el 57% de los dispositivos IoT en el mercado son vulnerables a ataques de severidad media o alta [8]. Un reporte de amenazas realizado por la compañía Nokia encontró que en el 2020 los ataques informáticos por malware a dispositivos IoT se duplicaron de un 16.17% en 2019 a 32.72% (Figura **2**) en 2020 [9]. Algunos factores que contribuyen a este crecimiento fueron por ejemplo la pandemia del virus Covid-19, el incremento de dispositivos IoT en el mercado y la facilidad de infectar y vulnerar los dispositivos IoT.

Figura 2: Dispositivos infectados 2019 - 2020



Fuente: Nokia [9]

Ahora bien, es evidente que la tecnología IoT presenta mucha vulnerabilidad con los atacantes cibernéticos. Este proyecto de investigación aporta un mecanismo de defensa para el usuario final que adquiere un dispositivo smart switch en su hogar inteligente. Esta investigación no se pudo realizar en todas las marcas de los dispositivos smart switch por que se sale del alcance para una investigación de esta magnitud, por lo tanto, solo se trabajó con una marca en específico. Para la realización de esta investigación se cuenta con el siguiente objetivo general y objetivos específicos:

Objetivo General

Crear un plan de alertas en dispositivos Smart switch/plug, basado en indicadores de compromiso (IoC), que, a través de un manejo de incidentes de seguridad, identifiquen y notifiquen al usuario los ataques informáticos en la red IOT.

Objetivos Específicos

- Identificar los ataques informáticos más relevantes que se pueden presentar en las redes IoT asociados a los Smart switch/plug.
- Clasificar los indicadores de compromiso que se generan en los dispositivos Smart switch/plug después de un ataque informático.

- Diseñar la arquitectura que monitoree en tiempo real al dispositivo Smart switch/plug en la red IoT.
- Realizar un plan de manejo de incidentes de seguridad para el usuario final una vez identificado los ataques informáticos.
- Validar el plan de alertas implementado a partir de ataques controlados evidenciados en alertas por correo electrónico.

Esta investigación presenta la forma cómo se realizan los ataques encontrados que vulneran la disponibilidad del dispositivo IoT, se muestra cómo se le hace su respectiva detección de acuerdo con el indicador de compromiso hallado y finalmente una documentación guía que ayudara al usuario final ante este tipo de situaciones. Esta investigación tiene los apartados de su resumen e introducción, su respectivo marco teórico donde se explican los conceptos y las tecnologías aplicadas, el estado del arte para comprender lo que se ha hecho en este campo, seguido de la metodología con los resultados mostrados para evidenciar el desarrollo de los objetivos y por último las conclusiones y trabajo a futuro.

1. Marco Teórico y Estado del Arte

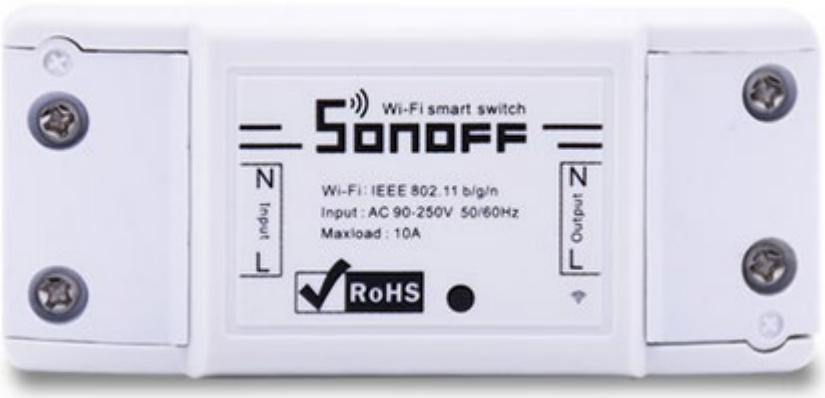
En este capítulo de la investigación se explica la tecnología utilizada por el smart switch elegido, la teoría clave para el entendimiento de los dispositivos y los ataques, los conceptos necesarios para entender la arquitectura implementada y la solución dada, los protocolos utilizados, los diferentes ataques informáticos replicados y luego se expone el estado del arte.

1.1 Marco teórico

1.1.1 Sonoff Smart switch

El Sonoff smart switch brinda la capacidad de convertir dispositivos ordinarios en dispositivos inteligentes, lo que le permite controlarlos a través de la aplicación móvil desde cualquier lugar. Al conectar el Smart switch con una bombilla, por ejemplo, se puede encender y apagar desde el celular. También trabaja con plataformas activadas por voz, incluidas Amazon Alexa y Google Assistant, esta característica lo hace más apetecido por el público de los sistemas inteligentes en el hogar o el entorno IoT.

Figura 3: Dispositivo Sonoff smart switch basic



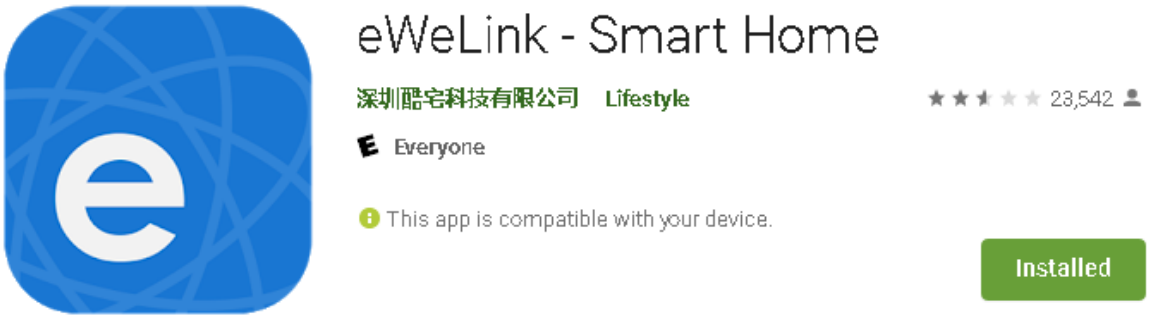
Fuente: Sonoff [10]

El proveedor del dispositivo (Figura 3), Sonoff, muestra la tecnología que utiliza y sus capacidades:

- Modelo: Sonoff basic
- Standard inalámbrico: Wi-Fi 2.4GHz 802.11 b/g/n
- Color: blanco
- Material: fuego retardante Acrilonitrilo butadieno estireno
- Seguridad: WEP/WPA-PSK/WPA2-PSK
- Fuente de poder: 90 – 250V AC
- Máxima corriente: 2200 W
- Temperatura para operar: 0°C – 40°C
- Wifi chip ESP8266

Con estos datos se esclarece el funcionamiento del dispositivo, la tecnología que tiene incorporada y los protocolos que maneja, a continuación, se profundiza un poco en los protocolos y tecnologías importantes. La aplicación que utiliza el Sonoff smart switch se llama “eWeLink” (Figura 4), se encuentra disponible para dispositivos Android y Apple. Desde esta aplicación es por donde se enciende o apaga el smart switch.

Figura 4: Aplicación móvil eWelink



Fuente: eWelink [11]

Una aplicación, innumerables dispositivos. eWeLink es la plataforma de aplicaciones que admite múltiples marcas de dispositivos inteligentes, incluido Sonoff. Permite conexiones entre hardware inteligente diversificado e integra consolas inteligentes populares como Amazon Alexa, Google Home. eWeLink sirve como el mejor centro de control del hogar. Algunas características de la

aplicación es que permite el control remoto, programación, temporizador, temporizador de bucle, avance lento, escena inteligente, uso compartido, agrupación, modo LAN [11].

1.1.2 El estándar 802.11 b/g/n

802.11 representa la designación IEEE para las redes inalámbricas. Existen varias especificaciones de redes inalámbricas bajo el nombre de 802.11. Algunos de los estándares más comunes en el mercado son la 802.11, 802.11a, 802.11b, 802.11g y 802.11n. Todos estos estándares utilizan el protocolo Ethernet y el método de acceso CSMA / CA.

Figura 5: Wifi chip ESP8266EX

Fuente: Searchnetworking [12]

El chip utilizado en el smart switch (Figura 5)es el que ofrece la tecnología de conexión a la red inalámbrica, en este caso el chip ESP8266EX utiliza el estándar 802.11 b/g/n.

Los canales de radiofrecuencia (RF) son una parte importante de la comunicación inalámbrica. Un canal es la banda de RF utilizada para la comunicación inalámbrica. Cada estándar inalámbrico IEEE especifica los canales que se pueden usar. El estándar 802.11a especifica rangos de frecuencia de radio entre 5.15 y 5.875GHz. En contraste, los estándares 802.11by 802.11g operan en el rango de

2.4 a 2.497GHz [13]. Como lo estableció el fabricante, el dispositivo smart switch opera en la frecuencia de 2.4GHZ y por lo tanto maneja el estándar de 802.11 b/g/n.

El método de acceso CSMA / CA (Carrier Sense Multiple Access with Collision Avoidance) o, en español, acceso múltiple con escucha de portadora y evasión de colisiones es un protocolo para transmisión de envió de datos en redes 802.11. A diferencia de CSMA / CD (Acceso múltiple con detección de portador / Detección de colisión) que se ocupa de las transmisiones después de que se haya producido una colisión, CSMA / CA actúa para evitar colisiones antes de que sucedan.

En CSMA / CA, tan pronto como un nodo recibe un paquete que se enviará, verifica para asegurarse de que el canal esté libre (ningún otro nodo está transmitiendo en ese momento). Si el canal está libre, se envía el paquete. Si el canal no está libre, el nodo espera un período de tiempo elegido al azar y luego vuelve a comprobar si el canal está libre. Este período de tiempo se denomina factor de retroceso y un contador de retroceso lo cuenta regresivamente. Si el canal está libre cuando el contador de retroceso llega a cero, el nodo transmite el paquete. Si el canal no está libre cuando el contador de retroceso llega a cero, el factor de retroceso se establece nuevamente y el proceso se repite [14].

Los estándares inalámbricos 802.11 pueden diferir en términos de velocidad, rangos de transmisión y frecuencia utilizada, pero en términos de implementación son similares. Todos los estándares pueden usar una infraestructura o un diseño de red inalámbrica, incluso admiten una arquitectura ad hoc (redes temporales sin tener la necesidad de un enrutador), y cada uno puede usar los mismos protocolos de seguridad.

- **EEE 802.11:** En realidad, hubo dos variaciones en el estándar inalámbrico 802.11 inicial. Ambos ofrecieron velocidades de transmisión de 1 o 2Mbps y la misma RF de 2.4GHz. La diferencia entre los dos estaba en cómo viajaban los datos a través de los medios de RF. Uno usaba FHSS y el otro usaba DSSS. Los estándares 802.11 originales son demasiado lentos para las necesidades modernas de redes y ahora ya no se implementan.
- **IEEE 802.11b:** el estándar 802.11b proporciona una velocidad de transmisión máxima de 11 Mbps. Sin embargo, los dispositivos están diseñados para ser compatibles con los

estándares 802.11 anteriores que proporcionaban velocidades de 1, 2 y 5.5Mbps. 802.11b usa un rango de RF de 2.4GHz y es compatible con 802.11g.

- **IEEE 802.11g**: 802.11g es un estándar inalámbrico popular hoy en día. 802.11g ofrece transmisión inalámbrica en distancias de más de 45 metros y velocidades de hasta 54Mbps en comparación con los 11Mbps del estándar 802.11b, el rango y las velocidades pueden ser afectadas por obstáculos, las métricas son basadas en un ambiente sin obstáculos. Al igual que 802.11b, 802.11g opera en el rango de 2.4GHz y, por lo tanto, es compatible con él.
- **IEEE 802.11n:** El objetivo del estándar 802.11n es aumentar significativamente el rendimiento en el rango de frecuencia de 2.4GHz y 5GHz. El objetivo básico del estándar era alcanzar velocidades de 100 Mbps, pero dadas las condiciones adecuadas, se estima que las velocidades 802.11n podrían alcanzar la asombrosa cifra de 600 Mbps. En la operación práctica, las velocidades 802.11n serán mucho más lentas. La distancia de cobertura se amplía un poco más 53 metros, el rango y las velocidades pueden ser afectadas por obstáculos, las métricas son basadas en un ambiente sin obstáculos.

El estándar que utiliza el dispositivo será determinado por el enrutador inalámbrico toda vez él es quien ofrece la señal inalámbrica a donde se conectara este [13].

1.1.3 Protocolos de Seguridad WEP y WPA/WPA2-PSK

Cuando se trata de la seguridad inalámbrica en WiFi, en realidad solo hay un par de opciones que tiene para el público común, especialmente si se está configurando una red inalámbrica doméstica. Los tres grandes protocolos de seguridad actuales son WEP, WPA y WPA2. Los dos grandes algoritmos que se utilizan con estos protocolos son TKIP y AES. Básicamente, los protocolos de seguridad inalámbrica surgieron a finales de los 90 y han ido evolucionando desde entonces. Afortunadamente, solo se aceptaron unos pocos protocolos y, por lo tanto, es mucho más fácil de entender [15].

WEP

WEP o Wired Equivalent Privacy (privacidad cableada equivalente) se lanzó en 1997 junto con el estándar 802.11 para redes inalámbricas. Se suponía que debía proporcionar una confidencialidad equivalente a la de las redes cableadas (de ahí el nombre).

WEP comenzó con un cifrado de 64 bits y eventualmente llegó hasta el cifrado de 256 bits, pero la implementación más popular en los enrutadores fue el cifrado de 128 bits. Desafortunadamente, muy poco después de la introducción de WEP, los investigadores de seguridad encontraron varias vulnerabilidades que les permitieron descifrar una clave WEP en pocos minutos.

Incluso con actualizaciones y correcciones, el protocolo WEP seguía siendo vulnerable y fácil de penetrar. En respuesta a estos problemas, WiFi Alliance introdujo WPA o WiFi Protected Access (acceso protegido wifi), que se adoptó en 2003 [15].

WPA

WPA en realidad estaba destinado a ser solo un remedio intermedio hasta que pudieran finalizar WPA2, que se introdujo en 2004 y ahora es el estándar utilizado actualmente. WPA utilizó TKIP o Protocolo de integridad de clave temporal como una forma de garantizar la integridad del mensaje. Esto era diferente de WEP, que usaba CRC o Cyclic Redundancy Check. TKIP fue mucho más fuerte que CRC.

Desafortunadamente, para mantener las cosas compatibles, WiFi Alliance tomó prestados algunos aspectos de WEP, lo que terminó haciendo que WPA con TKIP también fuera inseguro. WPA incluyó una nueva característica llamada WPS (WiFi Protected Setup), que se suponía que facilitaría a los usuarios la conexión de dispositivos al enrutador inalámbrico. Sin embargo, terminó teniendo vulnerabilidades que permitieron a los investigadores de seguridad descifrar una clave WPA en un corto período de tiempo también [15].

WPA2

WPA2 estuvo disponible a partir de 2004 y fue requerido oficialmente en 2006. El mayor cambio entre WPA y WPA2 fue el uso del algoritmo de cifrado AES con CCMP en lugar de TKIP.

En WPA, AES era opcional, pero en WPA2, se debe elegir entre encriptación con AES o utilizar la TKIP. En términos de seguridad, AES es mucho más seguro que TKIP. Se han encontrado algunos problemas en WPA2, en la actualidad ya se encontraron vulnerabilidades y problemas de seguridad con este protocolo, pero es el más seguro que se encuentra en el mercado disponible para los hogares domésticos mientras el protocolo WPA3 se comercializa más y se expande.

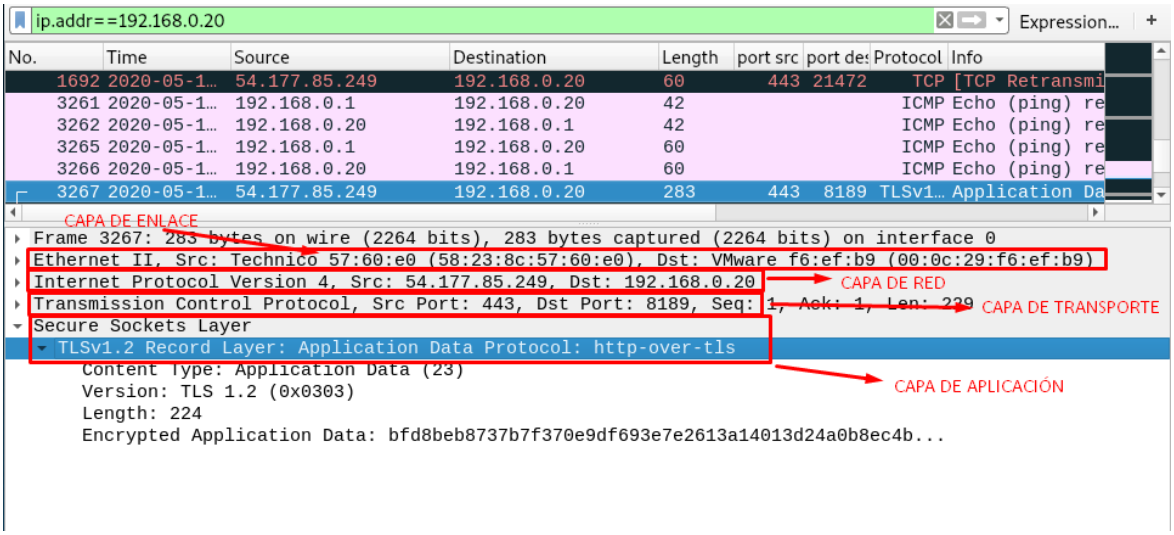
WPA utiliza una clave de 64 bits o de 128 bits, la más común es la de 64 bits para los enrutadores domésticos. PSK o Clave Pre-Compartida está diseñado para redes en hogares y oficinas pequeñas donde cada usuario tiene la misma frase contraseña[15].

Una vez más, el protocolo de seguridad a utilizar por el dispositivo es dada por el protocolo configurado en el enrutador, esto ya varía de acuerdo con el administrador del dispositivo y la configuración que este le dio.

1.1.4 Modelo de comunicación del smart switch Sonoff

Con analizar el comportamiento del tráfico de red que navega hacia y afuera del dispositivo se logró establecer el modelo de comunicación y fue evidente que utiliza el modelo de conexión por capas de TCP-IP. Con la utilización de un software para capturar el tráfico en la red, wireshark (Figura 6), se pudo analizar el modelo de comunicación del dispositivo:

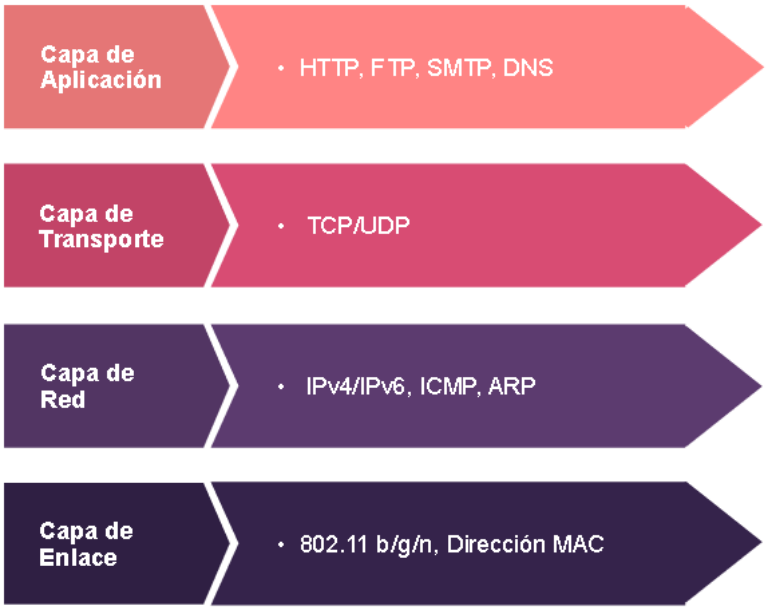
Figura 6 : Capas del modelo TCP/IP wireshark



Fuente: Autor

Las capas evidenciadas son cuatro (Figura 7), la capa de enlace, la capa de red, la capa de transporte y la capa de aplicación. A continuación, se muestran algunos ejemplos de lo que ocurre en cada una de las capas.

Figura 7: Capas del modelo TCP/IP



Fuente: Autor

- **Capa de aplicación**: Es la capa ubicada en la parte más superior de la pila de protocolos y es donde la aplicación recibe los comandos enviados e interactúa con el usuario. En la captura de tráfico con wireshark se observa como la aplicación envía datos encriptados mediante el puerto 443 correspondiente a TLS.
- **Capa de transporte**: La capa de transporte establece la conexión entre las aplicaciones que se ejecutan en diferentes hosts. Utiliza TCP para conexiones confiables y UDP para conexiones rápidas. Realiza un seguimiento de los procesos que se ejecutan en las aplicaciones que se encuentran arriba al asignarles números de puerto y utiliza la capa de red para acceder a la red TCP / IP. En esta capa es donde se define el puerto que deberá seguir el paquete.
- **Capa de red:** La capa de red es responsable de crear los paquetes que se mueven a través de la red. Utiliza direcciones IP para identificar el origen y el destino del paquete.
- **Capa de enlace:** La capa de enlace es la combinación de la capa de la capa de enlace de datos y la capa física del modelo OSI tradicional. Es responsable de crear las tramas que se mueven a través de la red. Estas tramas encapsulan los paquetes y usan direcciones MAC para identificar el origen y el destino. En esta capa también viene embebida los estándares de conexión y los protocolos de conexión que irían en la capa física [16].

1.1.5 Indicadores de compromiso

Es importante aclarar el concepto de Indicador de compromiso (IOC), toda vez es parte fundamental para la investigación presente. Los indicadores de compromiso (IOC) son datos forenses dejados atrás, como los datos que se encuentran en los logs de los sistemas, que identifican actividades potencialmente maliciosas en un sistema o red. Estos IoC ayudan a los profesionales de tecnología con la detección de robo de información, infecciones de malware, u otra actividad de amenaza. Al monitorear los indicadores de compromiso, las organizaciones pueden detectar ataques y actuar rápidamente para evitar que ocurran violaciones o limitar los daños al detener los ataques en etapas tempranas. Los indicadores de compromiso actúan como migajas de pan para detectar actividad maliciosa al principio de la secuencia de ataque. Estas actividades inusuales son las banderas rojas que indican un ataque potencial o en curso que podría llevar a un daño, robo de datos o un compromiso de los sistemas [17].

En la búsqueda para detectar brechas de datos más rápidamente, los indicadores de compromiso pueden actuar como rutas de acceso importantes para los profesionales de seguridad que vigilan sus entornos de TI. Según los expertos, algunos indicadores clave de compromiso para monitorear son [18]:

- Tráfico de red inusual.
- Archivos desconocidos, aplicaciones y procesos en el sistema.
- Actividad sospechosa en administrador o cuentas privilegiadas.
- Actividades irregulares, como el tráfico en los países inusuales.
- Inicios de sesión, acceso y otras actividades de red que indiquen ataques de sondeo o de fuerza bruta.
- Picos anómalos de solicitudes y volumen de lectura en archivos.
- Tráfico de red por puertos inusualmente utilizados.
- Configuraciones de archivos, servidores de nombres de dominio (DNS) y registros alterados, cambios en la configuración del sistema.
- Grandes cantidades de archivos comprimidos y datos inexplicablemente encontrados en ubicaciones inusuales
- Llaves de registro alteradas

Sobre el tráfico de red inusual es el indicador por el cual nos basamos para la recolección de la evidencia del ataque y así poder implementar el sistema de detección y alerta.

1.1.6 Ataques de Red

Se definió por enfocar la investigación en ataques que interrumpan la disponibilidad del servicio del smart switch, toda vez el dispositivo no guarda información confidencial de nadie y como no tiene almacenamiento entonces tampoco nos impacta en la integridad de los datos.

Los ataques establecidos impactan al dispositivo en la capa de enlace y en la capa de red. En la capa de enlace tenemos unos ataques con la suplantación de la dirección MAC y en la capa de red

tenemos ataques con la suplantación de la dirección IP, estos ataques se conocen como Arp spoofing o Arp poisoning (envenenamiento de la ARP). Otro de los ataques replicados es un MITM (man in the middle) o en español hombre en el medio, ataques por denegación de servicio de tipo flooding que afectan al dispositivo en su capa de red y ataques de des-autenticación al dispositivo de la red conectada.

Protocolo de resolución de direcciones (ARP)

El Protocolo de resolución de direcciones (ARP) consiste en un procedimiento para asignar una dirección de protocolo de Internet dinámica (dirección IP) a una dirección de máquina física permanente en una red de área local (LAN). La dirección física de la máquina también se conoce como Control de acceso al medio o dirección MAC.

El trabajo del ARP es esencialmente traducir direcciones de 32 bits (IP) a direcciones de 48 bits (MAC) y viceversa. Esto es necesario porque en la versión 4 de IP (IPv4), el nivel más común de protocolo de Internet (IP) en uso hoy en día, una dirección IP tiene 32 bits de largo, pero las direcciones MAC tienen 48 bits.

ARP funciona entre las capas de red 2 y 3 del modelo de interconexión de sistemas abiertos (modelo OSI). En este modelo evidenciado, el modelo TCP/IP, la dirección MAC existe en la capa 1 referenciada anteriormente, la capa de enlace de datos, mientras que la dirección IP existe en la capa 2, la capa de red [19].

ARP Poisoning

El envenenamiento ARP (también conocido como ARP Spoofing) es un tipo de ataque informático llevado a cabo a través de una red de área local (LAN) que implica el envío de paquetes ARP maliciosos a una puerta de enlace predeterminada en una LAN para cambiar los emparejamientos con su tabla de dirección IP y las direcciones MAC que tiene registradas. La dirección MAC o media Access control (control de acceso de medio) es un código que identifica a la tarjeta de red de un dispositivo, está compuesta por números hexadecimal distribuidos entre 6 pares de dígitos, por

ejemplo: 00:0d:81:b1:c0:8b. A continuación, se muestra un ejemplo de una tabla ARP encontrada en un enrutador cisco,

Tabla 1: Tabla ARP router cisco

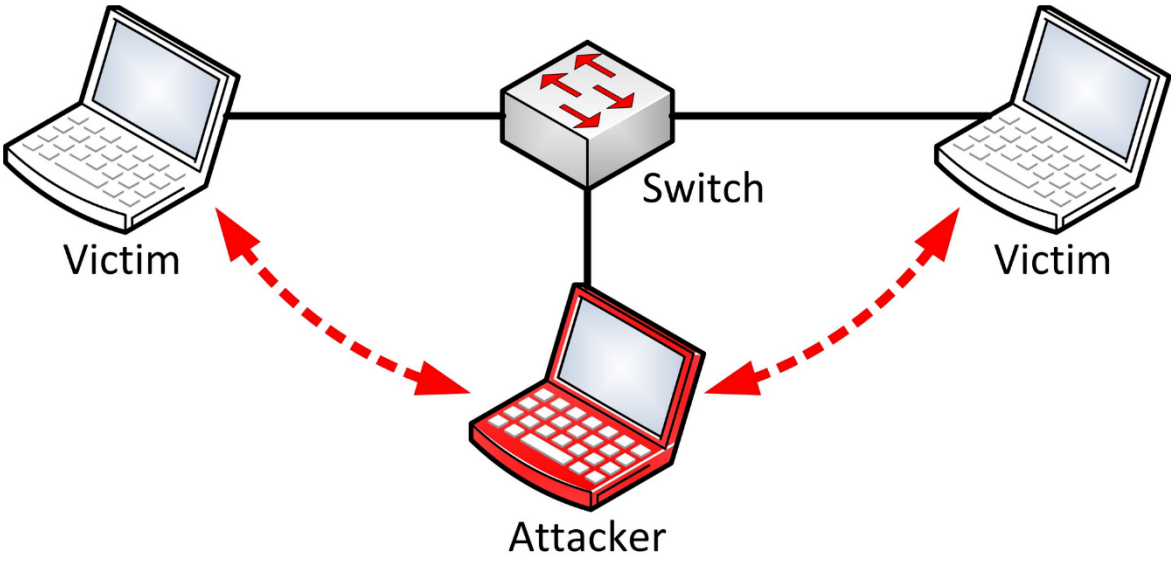
```
R1#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.0.1               -   0060.5C32.7E01  ARPA   GigabitEthernet0/0
Internet  10.0.0.10              6   000C.85CA.AD73  ARPA   GigabitEthernet0/0
Internet  172.16.0.1             -   0060.5C32.7E02  ARPA   GigabitEthernet0/1
Internet  172.16.0.2            10   0001.63DB.1802  ARPA   GigabitEthernet0/1
```

Fuente: Bestestredteam [20]

Como se puede observar en la Tabla 1, el dispositivo guarda en la tabla las direcciones IP y las asocia con las direcciones físicas MAC de los dispositivos de la red local LAN. Básicamente el ataque lo que busca es modificar esta tabla y engañar al enrutador con datos falsos de dirección MAC y dirección IP. El protocolo ARP traduce las direcciones IP en direcciones MAC. Debido a que el protocolo ARP fue diseñado exclusivamente para la eficiencia y no para la seguridad, los ataques de envenenamiento ARP son extremadamente fáciles de llevar a cabo siempre que el atacante tenga el control de una máquina dentro de la LAN de destino o esté directamente conectada a ella.

Por consiguiente el ataque en sí consiste en que un atacante envía un mensaje de respuesta ARP falsa a la puerta de enlace de red predeterminada, informándole que su dirección MAC debe estar asociada con la dirección IP de su objetivo (y viceversa, por lo que el MAC de su objetivo es ahora asociado con la dirección IP del atacante). Una vez que la puerta de enlace predeterminada ha recibido este mensaje y transmite sus cambios a todos los demás dispositivos en la red, todo el tráfico del destino a cualquier otro dispositivo en la red viaja a través de la computadora del atacante, permitiendo que el atacante lo inspeccione o modifique antes de reenviarlo a su verdadero destino. Debido a que los ataques de ARP Poisoning ocurren en un nivel tan bajo, los usuarios objetivo de ARP Poisoning rara vez se dan cuenta de que su tráfico está siendo inspeccionado o modificado. Además de los ataques Man-in-the-Middle (MITM), el envenenamiento ARP se puede usar para causar una condición de denegación de servicio en una LAN simplemente interceptando o soltando y no reenviando los paquetes del objetivo [21].

Figura 8: ARP Poisoning



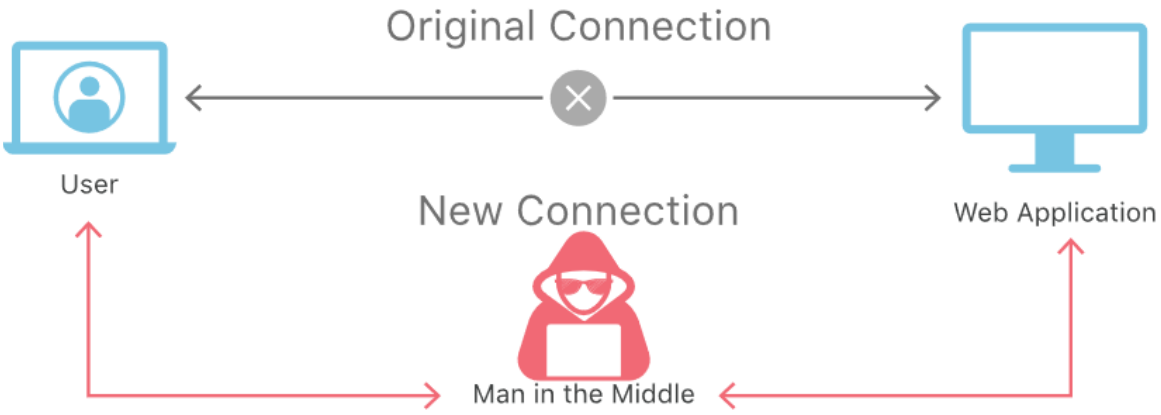
Fuente: Paloalto [22]

Al realizar el ataque (Figura 8), el atacante puede ver todo el tráfico saliente de la víctima y decidir qué hacer con él.

MITM (Man in the middle u hombre en el medio)

En un ataque de hombre en el medio, el atacante se coloca entre dos dispositivos (a menudo un navegador web y un servidor web) e interceptan o modifican las comunicaciones entre los dos. Los atacantes pueden recopilar información y suplantar a cualquiera de los dos agentes. Además de los sitios web, estos ataques pueden dirigirse a comunicaciones por correo electrónico, búsquedas de DNS y redes WiFi-públicas. Los objetivos típicos de los ataques man-in-the-middle incluyen todo tipo de empresas, el comercio por correo electrónico, las redes wifi públicas, bancos y lugares donde pueda haber personas que el atacante considere “importantes” [23].

Figura 9: Ataque MITM



Fuente: Angela Wood[23]

Se puede pensar en un ataque de hombre en el medio (Figura 9) como un trabajador postal deshonesto que se sienta en una oficina de correos e intercepta cartas escritas entre dos personas. Este empleado postal puede leer mensajes privados e incluso editar el contenido de esas cartas antes de pasarlas a sus destinatarios. El ataque de hombre en el medio tiene muchos fines, como, por ejemplo: encontrar usuarios y contraseñas de una víctima, robar información financiera, robar correos, pero para la investigación presente se utilizó este ataque para denegar el servicio de la víctima. El ataque de hombre en el medio también funciona con la modificación de la tabla ARP de las víctimas.

Ataque de denegación de servicio (DOS) por flooding

Un ataque de denegación de servicio (DoS) es un ataque destinado a apagar/inhabilitar o dejar sin servicio a una máquina o red, haciéndola inaccesible para sus usuarios clientes. Los ataques DoS logran esto inundando el objetivo con tráfico o enviándole información que desencadena un bloqueo. En ambos casos, el ataque DoS priva a los usuarios legítimos (es decir, empleados, miembros o titulares del servicio) del servicio o recurso que esperaban.

Las víctimas de ataques DoS a menudo se dirigen a servidores web de organizaciones de alto perfil, como bancos, empresas comerciales y de medios, u organizaciones gubernamentales y comerciales. Aunque los ataques DoS generalmente no resultan en el robo o la pérdida de

información significativa u otros activos, pueden costarle a la víctima una gran cantidad de tiempo y dinero para manejar.

Hay dos métodos generales de ataques DoS: servicios de inundación o servicios de bloqueo. Los ataques de inundación se producen cuando el sistema recibe demasiado tráfico para que el servidor se almacene en el búfer, lo que hace que se desaceleren y finalmente se detengan. Este ataque por inundación es el que se conoce como “flooding”. Los ataques de inundación populares incluyen [24]:

- **Ataques de desbordamiento de búfer:** El ataque DoS más común. El concepto es enviar más tráfico a una dirección de red de lo que los programadores han construido para manejar el sistema. Incluye los ataques enumerados a continuación, además de otros que están diseñados para explotar errores específicos de ciertas aplicaciones o redes.
- **Inundación ICMP:** Aprovecha los dispositivos de red mal configurados mediante el envío de paquetes falsificados que hacen ping a cada computadora en la red de destino, en lugar de solo una máquina específica. La red se activa para amplificar el tráfico. Este ataque también se conoce como el ataque de los pitufos o ping de la muerte.
- **Inundación SYN:** Envía una solicitud para conectarse a un servidor, pero nunca completa el protocolo de conexión si no que crea interminables peticiones de conexión. Continúa hasta que todos los puertos abiertos estén saturados de solicitudes y ninguno esté disponible para que los usuarios legítimos se conecten. Este tipo de inundación fue la realizada en la investigación.

Ataque por desautenticación o deauthentication attack

La desautenticación es una técnica utilizada por hackers y profesionales de ciberseguridad para alterar o deshabilitar las conexiones inalámbricas en una red. El objetivo final del ataque es eliminar a los usuarios de una red y evitar conexiones adicionales, dejando el punto de acceso inutilizable.

Un ataque de desautenticación puede describirse como un ataque DDoS (Denegación de servicio distribuida). Las nuevas redes inalámbricas y los dispositivos habilitados para Internet han cambiado el panorama de los ataques de desautorización. Se han desarrollado más métodos y aplicaciones de desautorización, pero el principio básico sigue siendo el mismo.

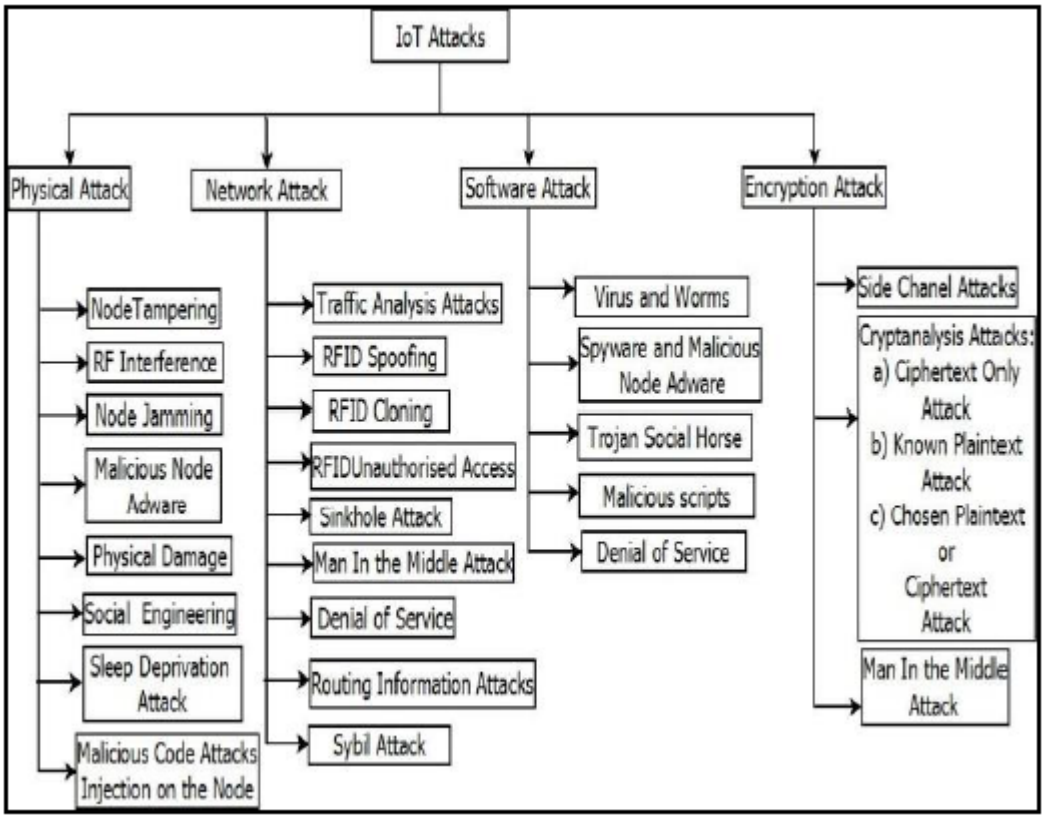
El ataque comienza con la identificación de las direcciones MAC de los usuarios conectados en las redes inalámbricas. El protocolo Wi-Fi está diseñado para manejar tramas de autenticación, estas son instrucciones formales para desconectar un dispositivo o estación de la red. El atacante falsifica su dirección MAC con las direcciones MAC del objetivo y envía un marco de desautorización al punto de acceso. El punto de acceso lee el marco de desautenticación y desconecta al usuario objetivo. Esto puede ser instantáneo y automatizado. Por ejemplo, un atacante podría desconectar a todos los usuarios de una red cercana y evitar que se conecten nuevos usuarios. Alternativamente, un atacante podría configurar su software de desautorización para deshabilitar todas las redes dentro del alcance del dispositivo [25].

1.2 Estado del arte

Con la investigación realizada se evidenció que existe muy poco material sobre los dispositivos smart switch/plug y de sus indicadores de compromiso. Tras buscar en muchas fuentes académicas y sitios confiables no se pudo encontrar otro proyecto que propusiera un sistema de alertas en dispositivos IoT en el hogar y mucho menos en smart switch/plug. Se concluye que esta investigación tiene un factor innovador en el aspecto de preocuparse por el usuario final y querer avisarle sobre el estado de su dispositivo inteligente. Un objetivo del actual trabajo consiste en probar ataques informáticos en los dispositivos Smart switch/plug, por lo tanto, lo primero en investigar fue identificar ¿Qué ataques existen para los dispositivos IoT en la actualidad? Toda vez información específica en los smart switch en IoT no es común de encontrar.

Existen varias categorías de ataques informáticos en los dispositivos IoT [26], Entre ellas se pueden encontrar ataques físicos, los que se explotan dentro del proceso de acceso, y los ataques de tipo lógico a nivel de red, software y de cifrado. Estos últimos tienen la posibilidad de ejecutarse o bien de forma local o remota, creando vectores de ataques más simples de ejecutar. Así mismo, los dispositivos IoT presentan una serie de ataques que pueden ser explotados acorde a las categorías (27 ataques encontrados), lo que supone una amplia gama de riesgos informáticos que los atacantes pueden tener.

Figura 10: Clasificación de ataques IoT



Fuente: Security attacks in IoT: A survey [26]

Esta investigación detalla los diferentes métodos de ataques informáticos hacía los dispositivos IoT (Figura 10), pero en un nivel muy general. Con esta investigación en desarrollo se planea analizar los ataques que sean específicamente compatibles con el dispositivo smart switch/plug para IoT.

En la investigación titulada “A Denial of Service Attack Method for an IoT System” cuatro investigadores chinos proponen diferentes métodos de ataque para causar una denegación de servicio a un sistema IoT. Los investigadores lanzan tres diferentes ataques de denegación de servicio mediante la utilización del sistema operativo Kali Linux. Mediante mediciones gráficas los investigadores calculan el tiempo de respuesta antes del ataque al dispositivo IoT y durante el ataque para evidenciar la denegación de servicio, los tres ataques utilizados fueron los siguientes [27]:

- Ataque DoS mediante Hping3: En este método el atacante envía 10000 paquetes de 1200 bytes cada uno, con un comando de cambio de IP aleatorio y Windows size de 64 “hping3–c 10000 –d 1200 –S–w 64 –p 6633 –flood –rand-source targetIP”.
- Ataque DoS mediante Hping3 con SYN: En este método se utilizan varias banderas en la modificación de los paquetes TCP, se utilizan SYN, PUSH y URG en cada uno de los paquetes enviados
  “hping3 –S–P–U–flood –V–rand-source targetIP”.
- Ataque DoS con nping: En este último método se envían 900000 peticiones al dispositivo víctima, se envían peticiones TCP y se configura así “nping –tcp-connect –rate=90000 –c 9000000 –q targetIP”.

Evaluando la efectividad se concluyó que el método más efectivo de ataque fue el primero, toda vez causó una pérdida del 90% de los paquetes enviados y recibidos por la víctima. El segundo método fue el menos efectivo con 19% de perdida y el tercer método tuvo 28% de perdida en la víctima [27].

Por otro lado, en una investigación enfocada a la denegación de servicio en dispositivos IoT “A Tool for Denial of Service Attack Testing in IoT” en la cual se utilizan ataques diferentes a los vistos en la investigación anterior. En esta investigación los autores crearon una herramienta de ataque utilizando el lenguaje de programación de C y C ++, los ataques utilizados fueron: *Sleep deprivation Attack*, *MQTT flooding Attack*, y *Radio frequency interference* y fueron lanzados contra un sensor IoT [28].

El ataque de *Sleep deprivation* tiene como objetivo interferir en los sensores del dispositivo IoT por la razón de que en muchas ocasiones los sensores son alimentados por baterías, entonces existe una posibilidad de agotar estas baterías para que los sensores dejen de funcionar. Con la herramienta que se desarrolló se pudo simular que el atacante es un servidor que tiene permiso para activar los sensores. Cuando se lanzó el ataque, se enviarán grandes cantidades de requisitos a los sensores causando que la vida útil de los sensores se reducirá después del ataque. El ataque de interferencia por radio frecuencia se logra en una red IoT de sensores con dispositivos interconectados por radiofrecuencia, grandes cantidades de mensajes de radio pueden bloquear fácilmente la red porque todos los nodos recibirán el mensaje una vez sea atacada. La herramienta diseñada logra enviar un mensaje de radio con un tamaño y una frecuencia personalizada que logra interferir en los dispositivos de la red.

Por último, el *MQTT flooding Attack*, es un ataque dirigido al protocolo MQTT y apunta a consumir el recurso de un bróker (servicio de comunicación a internet) para que no funcione y poder administrar los recursos de publicación y suscripción de comunicación. La herramienta está diseñada para crear números personalizados de atacantes. Cada atacante seguirá publicando un mensaje de comunicación con una carga útil personalizada (*payload*) y todos publican al mismo tiempo [28]. La desventaja de esta investigación es que los autores no publicaron la herramienta realizada, pero ellos dejaron los resultados de los ataques y con esta información se puede investigar en otras fuentes sobre como lanzar estos tipos de ataques y probarlos en el smart switch/plug.

Toda vez el Sonoff smart switch utiliza un chip para la conexión wifi llamado ESP8266, se investigó sobre posibles ataques hacía este. En el portal de GitHub se reportó sobre una vulnerabilidad descubierta hacia este chip el cual lo vuelve blanco de un ataque llamado “Krack Attack” [29]. Lo que implica este ataque es que el ciber delincuente puede “interceptar y leer el tráfico que envía un dispositivo e incluso manipular este tráfico. Es decir, que además de violar completamente la privacidad, puede realizar otros ataques, como insertar códigos maliciosos, manipular los DNS o utilizar otras técnicas combinadas con ataques de Man In The Middle [30]. Toda vez el Sonoff utiliza el protocolo de Wifi “WPA2-PSK” esto lo deja vulnerable para todos los ataques existentes en este, como por ejemplo: eavesdropping (sniffing), ataque de fuerza bruta con diccionario, envenenamiento de ARP, krack Attack, crackeo de contraseñas.

El siguiente paso fue buscar alguna investigación orientada a los indicadores de compromiso (IoC) los cuales son los que nos van a decir si el sistema corre peligro o no. La investigación que más se acercaba a estos indicadores es de hace varios años, se llama “An Analytics Framework to Detect Compromised IoT Devices using Mobility” [31]. El autor propone un framework analítico donde se especifican unos indicadores de red los cuales son aplicados en los dispositivos IoT. El autor también propone la utilización de un protocolo llamado “CoAP” el cual es más liviano que el http, para el envío del tráfico. El tráfico se envía a un servidor de análisis y el servidor informa si ha detectado actividad inusual. Este servidor de análisis detecta si hay cambios en el comportamiento de envió de paquetes del dispositivo IoT, luego analiza los indicadores de compromiso ya subidos a este y alerta al usuario sobre el sistema comprometido.

Este es un documento del 2013, se debería profundizar más sobre el protocolo CoAP del cual el autor habla, lo negativo es que todo es a nivel teórico, es un framework para aplicar, lo cual genera muchas preguntas por resolver sobre todo acerca de cómo ejecutar y cargar los indicadores de compromiso y sobre ¿qué tecnología tendría que ser el servidor de análisis? Esta investigación nos genera varias ideas para la implementación de la arquitectura actual, con esta arquitectura se crearán los indicadores de compromiso y se explicara a detalle sobre ¿cómo se hicieron estos indicadores?

Otra investigación encontrada que trabaja con los IoC la cual también es un framework, es una investigación que si tiene un desarrollo y una metodología de implementación, se llama “Dridex: Analysis of the traffic and automatic generation of IOCs” [32]. Básicamente traduce a “Análisis de tráfico y automatización y generación de Indicadores de compromiso”.

Este *framework* habla de recibir un tráfico entrante (el virus dridex) y lo que entrega son los indicadores de compromiso de la red. Un indicador de compromiso es una información que puede usarse para identificar un sistema potencialmente comprometido. Podría incluir sospechosas direcciones IP, nombres de dominio, direcciones de correo electrónico, hashes de archivos, etc. Los indicadores de compromiso son analizados utilizando el software *STIX*, es un lenguaje que permite que la información sobre amenazas sea fácilmente almacenada, analizada y compartida de manera consistente. Aparte de *Stix*, también nos revela otras herramientas para la creación de indicadores de compromiso como: *openIOC, CybOX, Cyber threat intelligence (CTI), IOC_Creator, IOCAware*.

En esta investigación los autores crearon 7 tipos de indicadores de compromiso, todos orientados a la red, y son: Indicador de tipo ICMP, TCP, UDP, HTTP, DNS, FTP, SSH. Toda vez el virus que ellos investigaban afectaba servicios y protocolos específicos que se debían vigilar con los indicadores creados. La desventaja de esta investigación es que toda fue orientada hacia el virus “Dridex”, por lo tanto, los mismos métodos de creación de indicadores tendrían que ser creados para IoC, en lo positivo esta investigación nos aporta mucho para el desarrollo de la metodología de este trabajo actual toda vez habla específicamente de software para utilizar.

Por último, se encontró una investigación enfocada en los indicadores de compromiso. Este documento propone un método de cuantificar la cantidad de indicadores de compromiso cuando se tienen muchos ataques. El método propuesto clasifica el indicador de compromiso con un algoritmo para saber cuál tiene más prioridad que el otro, la investigación se llama “Method of Quantification of Cyber Threat based on Indicator of Compromise” [33]

Este documento es claro en que identifica los puntos que se tienen en cuenta para hacer los indicadores de compromiso:

- Verificar el origen del indicador de compromiso
- Verificar la hora del indicador de compromiso
- Verificar las bases de datos de *blacklist*
- Verificar el cambio constante en los servicios del *dns*
- Verificar las direcciones URL
- Verificar el código malicioso detectado
- Verificar si el mismo dominio es un distribuidor constante de amenazas
- Verificar la alteración histórica en la web [33]

Con estos indicadores el algoritmo de este articulo clasifica el indicador según el ítem que tenga. A pesar de la descripción, el artículo no es específico en decir qué software se utilizaron y tampoco cómo.

En consecuencia al proceso de atención de incidentes, la empresa consultora [34] establece varios procesos para el ciber riesgo y para la Automatización de Procesos Robóticos (RPA), dentro de las propuestas indica la necesidad de contar con un proceso de monitoreo y gestión de alertas para la

supervisión de eventos de seguridad, lo que constituye un marco de gestión más allá de los elementos tradicionales, lo que fortalece la estructura de seguridad que puede ser aplicada en IoT u otra tecnología.

Adicional a esto, se encontró una investigación enfocada en el análisis de indicadores de compromiso para los ransomware con la utilización de técnicas de aprendizaje de máquina. En esta investigación los autores recolectaron un total de 848 muestras de ransomware y los emularon en un entorno controlado donde se afectaba un computador de estudio, se llama “Analysing Indicator of Compromises for Ransomware: Leveraging IOCs with Machine Learning Techniques” [35] .

Mediante la implementación de Cuckoo sandbox (un ambiente virtual de pruebas seguras y vigiladas) los investigadores de este trabajo pudieron encontrar 45 tipos de indicadores de compromiso relacionados con los ransomware. De estos 45 tipos de indicadores de compromiso se clasificaron 11 de ellos como los más críticos, algunos ejemplos de la clasificación se indican a continuación:

- Indicadores de compromiso en la red: ¿El ransomware hace peticiones http? ¿Hace peticiones al DNS?
- Indicadores de compromiso en el sistema: ¿Cambiaron las opciones de arranque del sistema? ¿Se deshabilitan las herramientas para la recuperación ante fallas? ¿El firewall se desactivo? ¿El antivirus sigue activo?
- Indicadores de compromiso estáticos: Revisar si algún software se actualizó, revisar descargar recientes, revisas las firmas en los procesos activos, revisar si se importaron librerías nuevas en el sistema.
- Indicadores de compromiso de comportamiento: Intento de detección de ambiente virtualizado, actividad recurrente en el sistema, directorio de temporales, edición y creación de nuevas llaves de registro, creación de archivos ocultos, eliminación de archivos originales en el disco, hace procesos de encriptación a los archivos del sistema operativo.

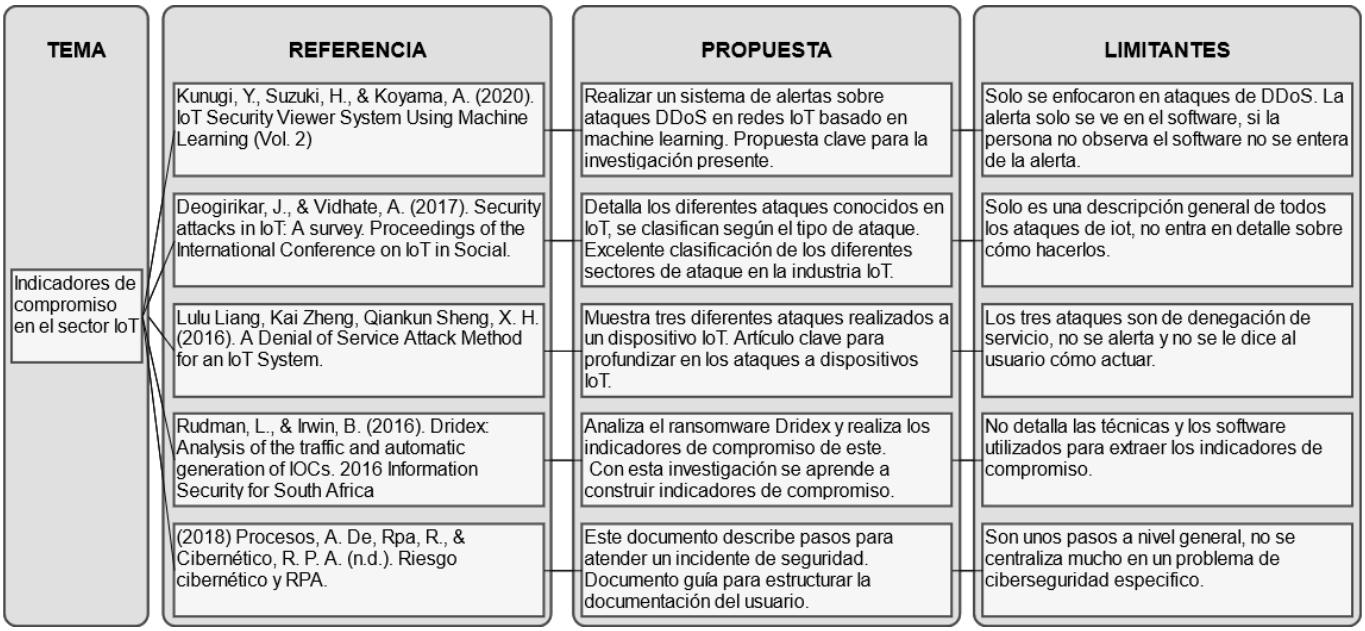
Estos comportamientos se evidencian con el malware ransomware identificados por los investigadores. Con esta metodología de clasificación del indicador de compromiso se entiende mejor cómo el sistema se ve afectado y qué tipo de consecuencias se sufren al ser infectado [35].

Esta última investigación clasifica muy bien los daños que puede causar los virus en los sistemas actuales a través de diversos indicadores de compromiso. Pero no nos dice si son compatibles con IoT, para el trabajo actual se deben analizar estos indicadores y verificar si aplican o no para el smart switch/plug.

Esta investigación no solo ayudará a mejorar la seguridad en los dispositivos smart switch/plug, sino también a otros dispositivos IoT que utilicen el chip de conexión a wifi ESP8266.

Una manera de entender más claramente los artículos expuestos se muestra en un gráfico ilustrativo para comprender mejor los pros y los contras de las investigaciones (Figura 11):

Figura 11: Cuadro comparativo investigaciones expuestas



Fuente: Autor

Con la recolección del material se pudo observar varios ataques informáticos, la clasificación de ellos, la generación de indicadores de compromiso, sistemas de defensa, por último, nos debemos preguntar sobre nuestro objetivo de estudio:

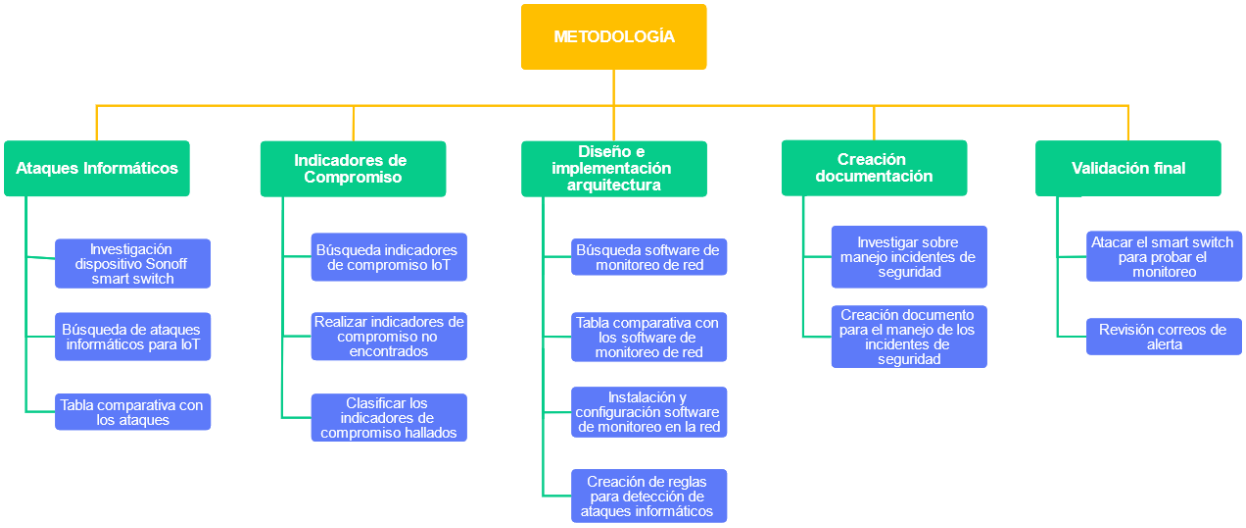
¿Cómo puede el usuario final enterarse de qué sus dispositivos Smart switch/plug se encuentran en peligro de ser vulnerados por un atacante cibernético?

2. Metodología y Resultados

El éxito de la propuesta radica toda en la ejecución de sus objetivos. Primero se deben identificar los ataques informáticos que apliquen a nuestro dispositivo, segundo es realizar los indicadores de compromiso o citarlos si ya existen, tercera será diseñar e implementar la arquitectura de alertas, nuestro cuarto objetivo es crear una documentación sobre el manejo de incidentes para el usuario final, y por último se hará la validación final de todo el modelo planteado.

Para lograr los objetivos propuestos en esta tesis de maestría, la metodología se dividió en 5 fases (Figura 12)

Figura 12: Metodología propuesta para cumplir con los objetivos planteados.



Fuente: Autor

Cada fase corresponde a uno de los objetivos específicos de la investigación, al cumplir las cinco fases se completa el objetivo general de la investigación. En este capítulo de la investigación se explica sobre cómo se realizó cada fase y se muestra el resultado obtenido.

2.1 Fase 1

La primera fase de la investigación tiene varios objetivos con el fin de contribuir a la realización del objetivo principal del trabajo presente. En esta fase se eligió el dispositivo IoT utilizado para hacer los ataques informáticos y posteriormente lanzar las alertas del ataque hacía este, adicional, en esta misma fase se eligieron los ataques informáticos que se utilizaron para la realización de la investigación.

2.1.1 Selección del dispositivo IoT

Estudiando el mercado de los smart switch/plug se encontró que uno de los dispositivos más asequibles y económicos de este mercado es el Sonoff smart switch basic, se puede encontrar entre 6 a 9 USD. Este switch (Figura 14) es configurable con el dispositivo consola de Amazon Alexa el cual permite una administración total de un hogar inteligente por medio de la voz. Sin embargo, también es configurable sin esta consola, el switch se encuentra a la venta en la mayoría de las tiendas virtuales como, por ejemplo: amazon, ebay, aliexpress, y mercadolibre.

Haciendo una búsqueda de mercado se encuentra con una marca de switches inteligentes colombiana, encontrada en la página https://www.smart-life.com.co. Esta empresa aparece en los registros de mercado desde el 2020, es una empresa nueva y no se cuenta con suficientes datos confiables como para comprar y utilizar sus productos, su producto es un tomacorriente wifi inteligente. Otro producto encontrado en el mercado es un plug de Amazon, el inconveniente es que se debe pedir importado y es mucho más costoso. El sonoff Smart switch ya se encuentra en el mercado colombiano, es muy popular en las búsquedas de Google y el proveedor lleva varios años en el mercado global (**Figura 13**: Comparación Smart switchesFigura 13) .

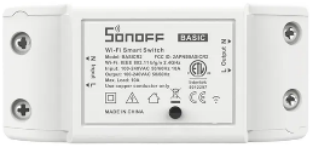
Figura 13: Comparación Smart switches

Amazon smart plug

- Cuesta 166% + que el Sonoff
- Cumplen con las mismas funciones
- Se debe importar

Toma Corriente Wifi inteligente

- No es una marca popular
- Empresa Colombiana nueva en el mercado, empezaron en el 2020
- Cuesta 28% más que el Sonoff



Sonoff Smart Switch

- Primero en búsquedas de Google relacionadas a Smart Switch IoT en Colombia
- Se encuentra en la mayoría de plataformas de ventas online
- El más económico del mercado

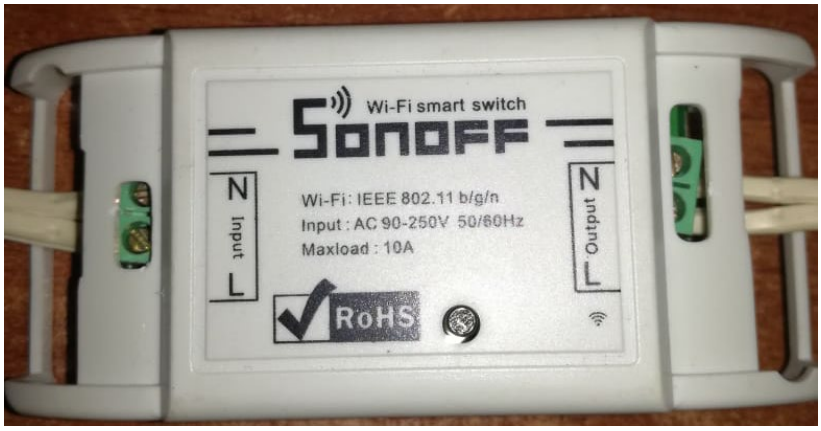
Fuente: Autor

Figura 14: Dispositivo Sonoff smart switch basic

Fuente: Amazon.com [36]

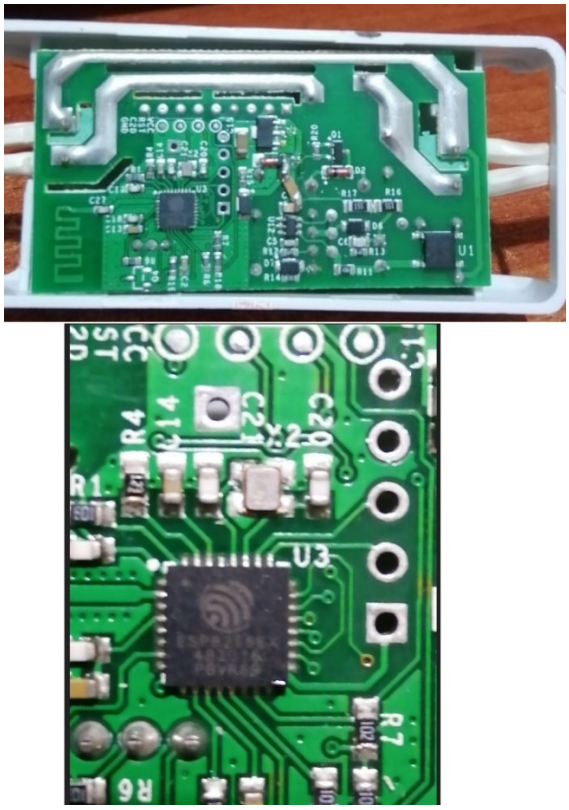
Al verificar los precios y lo fácil que es instalar un dispositivo de estos se procedió a adquirir uno por la web de Amazon. Una vez se obtuvo el dispositivo, se continua con destapar la cobertura del dispositivo para verificar que tipo de chip utilizaba, se encontró lo siguiente (Figura 15):

Figura 15: Dispositivo Sonoff smart switch basic físico



Fuente: Autor

Figura 16: Chip ESP8266



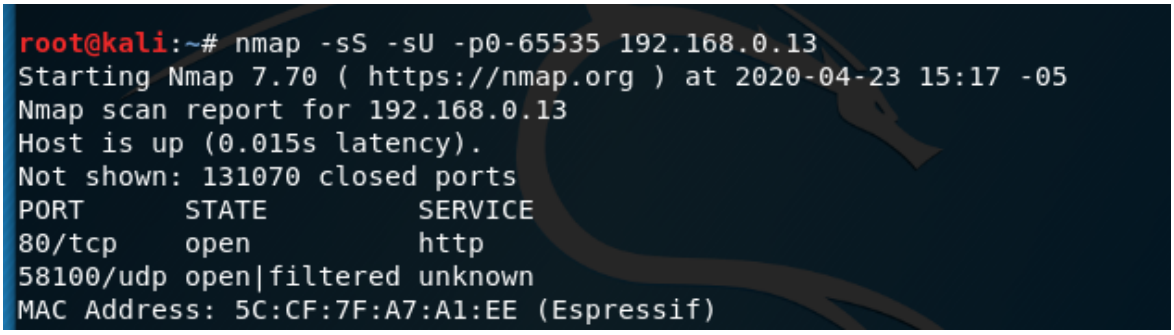
Fuente: Autor

La tecnología que utiliza el Sonoff smart switch para conectarse a la red IoT es un chip (Figura 16) de la familia ESP 8266, este chip es muy vendido para los dispositivos IoT toda vez es muy liviano, pequeño y presta los servicios de conectividad wifi con el protocolo Wi-Fi 2.4GHz 802.11 b/g/n [10].

La empresa fabricante de estos chips, Espressif, reportó en el 2018 que ya había vendido más de 100 millones de ejemplares, gracias a sus precios bajos y tecnología de punta [37].

Antes de empezar con los ataques hacia el dispositivo primero se identificó los puertos lógicos de este para verificar cuál de ellos podía ser el puerto de ataque. Gracias a la herramienta de “nmap” se escanearon todos los puertos TCP del dispositivo y se encontró lo siguiente [38]:

Figura 17: Escaneo de puertos con nmap



Fuente: Autor

Nmap ("Network Mapper") es una utilidad gratuita y de código abierto utilizada para el descubrimiento de redes y la auditoría de seguridad. Muchos administradores de sistemas y redes también lo encuentran útil para tareas como el inventario de la red, actualización de servicios y la supervisión del tiempo de actividad del host o del servicio. Nmap utiliza paquetes IP sin procesar de formas novedosas para determinar qué hosts están disponibles en la red, qué servicios (nombre y versión de la aplicación) ofrecen, qué sistemas operativos (y versiones de SO) están ejecutando, qué tipo de paquetes de filtros / firewalls están en uso, y docenas de otras características.

Con el comando **"nmap -sS -sU -p0-65535 192.168.0.13"(**Figura 17**),** se le indica a nmap que haga un escaneo del dispositivo smart switch cuya IP en este caso es la 192.168.0.13. El parámetro "-sS" indica que se utilizaran paquetes de tipo SYN durante el escaneo, "-sU" indica el escaneo hacia los puertos UDP y el "-p0-65535" indica la revisión de todos los puertos lógicos del dispositivo. Gracias a nmap se encontró que el dispositivo tiene 2 puertos abiertos, el puerto 80 y el 58100, a través del puerto 80 se enfocaran nuestros ataques de red. También se aprendió que la dirección física del dispositivo o dirección MAC es la 5c:c7:7f:a7:a1:ee.

2.1.2 Identificación de ataques informáticos asociados a los Smart switch

En el planteamiento de objetivos se definió el primero de ellos como la identificación de ataques más *relevantes* presentados en las redes IoT que puedan afectar a los smart switch. Pero ¿Qué significa relevante para un ataque dirigido a un smart switch?

Según la RAE, "relevante" significa *sobresaliente, destacado* [39]. De acuerdo con la norma ISO 27001- 2013 se definen tres pilares fundamentales para la seguridad de la información:

Integridad: propiedad que asegura que la información no es alterada sin autorización en su trasporte, procesamiento o almacenamiento.

Disponibilidad: propiedad que asegura que los activos de la información están disponibles para personal autorizado en su uso y demanda.

Confidencialidad: propiedad que asegura que la información es accedida solamente por personal o entidad autorizada.

El dispositivo smart switch/plug tiene tecnología muy liviana que no almacena datos de los usuarios, la información que contiene es únicamente código fuente para operar y debe estar siempre conectado a la red inalámbrica para funcionar.

Por consiguiente, debemos verificar cuál de los pilares es "relevante" para el dispositivo. Los smart switch actuales no manejan información significante de la cual nos debamos preocupar, estos dispositivos no contienen información del usuario, por lo tanto, el pilar de la integridad no es relevante en este caso. No es necesario velar por la Confidencialidad en estos dispositivos puesto que los usuarios nunca van a acceder a la información que haya en este simplemente porque no tiene. Por último, la disponibilidad si es muy crítica en estos dispositivos toda vez ellos deben estar en funcionamiento las 24 horas por siete días a la semana. El usuario puede necesitar su dispositivo IoT en cualquier momento del día y si este dispositivo falla en su disponibilidad entonces se tiene un problema critico en su funcionamiento.

Teniendo en cuenta lo anterior, para nuestra investigación se va a considerar como relevantes a todos los ataques informáticos que se presenten en las redes iot asociados con los smart switch/plug que afecten la disponibilidad del dispositivo.

En la investigación [26] se categorizan los ataques de IoT en cuatro tipos:

- Ataques Físicos
- Ataques de red
- Ataques de software
- Ataques de encriptación

Habiendo definido como pilar principal la disponibilidad en el dispositivo, se extrajeron de este reporte los ataques que afectan dicha disponibilidad en el servicio al dispositivo de investigación. En la categoría de los ataques de red se encontró que el ataque de hombre en el medio, el ataque de denegación de servicio y el node jamming también conocido como deauthentication attack son los que afectan la disponibilidad del servicio.

El smart switch al ser un dispositivo que trabaja con el protocolo 802.11n presenta una dirección MAC y dirección IP como se definen en el modelo OSI en su segunda y tercera capa. Toda vez es un dispositivo ligero y sin mayor seguridad/funciones no contiene configuración por parte del usuario, pero si contiene conectividad a nivel de capa dos. Investigando los ataques más comunes en capa dos encontramos el ataque por Mac spoofing el cual nos puede dejar sin disponibilidad e igual con el IP spoofing [40].

Tabla 2: Ataques y contramedidas en el modelo OSI

Attacks	**Characteristics**	**Countermeasures**
Physical layer attacks		
Eavesdropping attack [14]	Confidential data packets interception.	Cryptographic techniques.
Jamming attack [15]	Legitimate data transmission interruption.	Spread spectrum techniques such as FHSS, DSSS and THSS.
MAC layer attacks		
MAC spoofing [16]	MAC addresses falsification.	Use of ARP packets.

MITM attack [17]	Communicating nodes impersonation.	Use of Virtual Private Networks (VPNs). [25], [26].
Network injection [18]	Preventing networking devices operation.	Reprogramming of network devices.
Network layer attacks		
IP hijacking [19]	Legitimate users IP address impersonation.	Firewalls [27], [30].
IP spoofing [20]	IP address falsification.	Firewalls [28].
Smurf attack [21]	Sending overwhelming number of ICMP requests.	Routers and individual users are configured not to constantly respond to ICMP requests.
Transport layer attacks		
TCP flood [22]	Sending overwhelming number of ping requests.	Increasing the TCP backlog and reducing the SYN timer.
UDP flood [23]	Sending overwhelming number of UDP packets.	Reducing the UDP packets response rate.
Application layer attacks		
Malware attack [24]	Disrupt or intercept the legitimate confidential data.	Firewalls and anti viruses.
SQL injection [29]	Gaining unauthorized access to several websites.	Firewalls and anti viruses.
SMTP attack [31], [32]	Email spoofing and password sniffing.	Firewalls and anti viruses.

Fuente: Security vulnerabilities, attacks and countermeasures in wireless sensor networks at various layers of OSI reference model: A Survey [40]

Los ataques informáticos (Tabla 2) que aplican al dispositivo que son más relevantes es el mac spoofing y el IP spoofing toda vez los otros no aplican o son directamente un DOS (denegación de servicio), en vista de que estos dos ataques se ven combinados en un ARP spoofing o también envenenamiento del ARP significa que se trabajará con spoofing del ARP. El ataque de eavesdropping si puede funcionar contra un smart switch pero no es relevante para la investigación toda vez no afecta la disponibilidad del dispositivo, el jamming Attack ya se eligió anteriormente. En los ataques de la capa MAC se eligió al mac spoofing, el MITM ya estaba elegido, una network injection es un ejemplo claro de DOS. En la capa de red se registra el IP Spoofing toda vez este afecta la disponibilidad del dispositivo, el ataque smurf es otra muestra de DOS, en la capa de transporte todos son ejemplos claros de ataques DOS. Finalmente, en ataques de la capa de

aplicación el malware Attack no afecta toda vez el dispositivo puesto que no tiene software corriendo, SQL injection no aplica y el ataque SMTP tampoco toda vez no se trabajó con correos del dispositivo. Los ataques DOS tienen muchas variantes y es por eso es que no se realizaron todas las variantes de DOS existentes, solo los necesarios para demostrar la funcionalidad del ataque.

2.1.3 Tabla comparativa con los ataques investigados

Una vez establecidos los ataques informáticos que se utilizaron para la investigación se procedió a crear una lista con la información importante de cada uno, como, por ejemplo: su nombre, tipo de ataque, fuente de consulta, la descripción de este y su efecto. En la siguiente Tabla 3 se describen los ataques informáticos más relevantes en los dispositivos Smart switch.

Tabla 3: Ataques informáticos relevantes a los smart switch/plug

Nombre	**Tipo**	**Fuente**	**Descripción**	**Efecto**
Deauthentication attack	Físico	J. Deogirikar and A. Vidhate, "Security attacks in IoT: A survey,"	Es un tipo de ataque que apunta a la comunicación entre el enrutador y el dispositivo	Deshabilitar la comunicación wifi con el dispositivo
Dos Attack by flooding	Red	J. Deogirikar and A. Vidhate, "Security attacks in IoT: A survey,"	Un ataque de denegación de servicio (DoS) es un ataque destinado a apagar o deshabilitar un dispositivo, haciéndolo inaccesible para sus usuarios previstos.	Dejar el dispositivo inaccesible para el usuario

Man in the middle	Red	G. N. Nayak and S. G. Samaddar, “Different Flavours of Man-In-The-Middle Attack , Consequences and Feasible Solutions"	Es un ataque que intercepta la comunicación de terceros por ejemplo un cliente y su servidor	Puede impedir el flujo correcto de tráfico hacia el dispositivo
Arp Poisoning	Red	Data, Mahendra “The Defense Against ARP Spoofing Attack Using Semi-Static ARP Cache Table”	Falsificar las peticiones ARP para engañar a los dispositivos en la red con la identidad errónea de un equipo	Impedir que el tráfico de red llegue al dispositvo smart switch

Fuente: Autor

Con esta lista de ataques elaborada se finalizó el objetivo 1, con base a estos ataques identificados se hicieron los indicadores de compromiso. Se concluye que los ataques que afectan la disponibilidad del dispositivo IoT son los de red, toda vez los ataques físicos implicarían que un sujeto manipule físicamente el dispositivo. Como estos ataques de red son los más comunes, radica la importancia de un sistema de detección en la red local.

2.2 Fase 2

En esta sección se realizó el segundo objetivo específico correspondiente a clasificar los indicadores de compromiso. Se hizo una búsqueda para encontrar indicadores de compromiso existentes que pudieran aplicar a la investigación, pero no se encontraron, por consiguiente se construyeron los indicadores de compromiso, también se clasificaron los indicadores de acuerdo al nivel de impacto del ataque.

2.2.1 Búsqueda indicadores de compromiso asociados a los ataques encontrados

En la investigación realizada no se encontraron indicadores de compromiso disponibles para los ataques identificados en la fase 1 en el dispositivo de interés. Se buscaron en muchos repositorios tales como www.iocbucket.com, www.threatminer.org, www.fireeye.com, otx.alienvault.com, www.virustotal.com, owasp.org. El principal factor común es que estos sitios ofrecen indicadores de compromiso, pero principalmente para malware, pues los malware son el software malicioso que más evidencia dejan en los equipos y los sistemas operativos. Hay cientos de indicadores de compromisos para los sistemas operativos, indicadores para malware, indicadores para ransomware, indicadores para sitios fraudulentos, hasta indicadores para páginas web vulnerables.

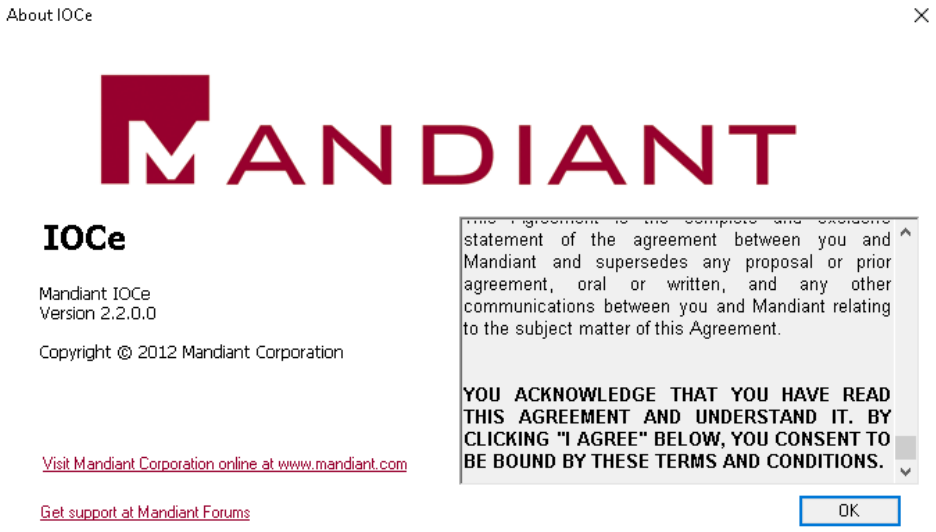
Sin embargo, encontrar indicadores de compromiso para los ataques que se presentan en la red IoT no fue posible, se encontraron herramientas que ayudan a crear los indicadores, pero no funcionaron para el ejercicio presente. Una de las herramientas más reconocidas para la creación y utilización de indicadores de compromiso se conoce como YARA.

YARA es una herramienta destinada (pero no limitada) a ayudar a los investigadores de malware a identificar y clasificar muestras de malware. Con YARA se puede crear descripciones de familias de malware (trojanos, rootkits, gusanos, entre otros) en base a patrones textuales o binarios, la herramienta se puede encontrar en https://virustotal.github.io/yara/. YARA es una herramienta utilizada por empresas como McAfee, ESET, Alien vault, Virustotal según lo muestran en su página web, pero como se puede observar la herramienta no es para documentar ataques de red en tiempo real. Funciona para realizar indicadores de malware, pero para los ataques en la red no es de mucha utilidad.

2.2.2 Realizar los indicadores de compromiso faltantes

Como la investigación no arrojó un resultado positivo para los ataques planteados, se procedió a crear indicadores para cada uno de los ataques planteados en la fase 1. Primero se instaló la herramienta gratuita ofrecida por www.fireeye.com para la creación de indicadores de compromiso llamada Mandiant IOCe (Figura 18).

Figura 18 : Software Mandiant IOCe



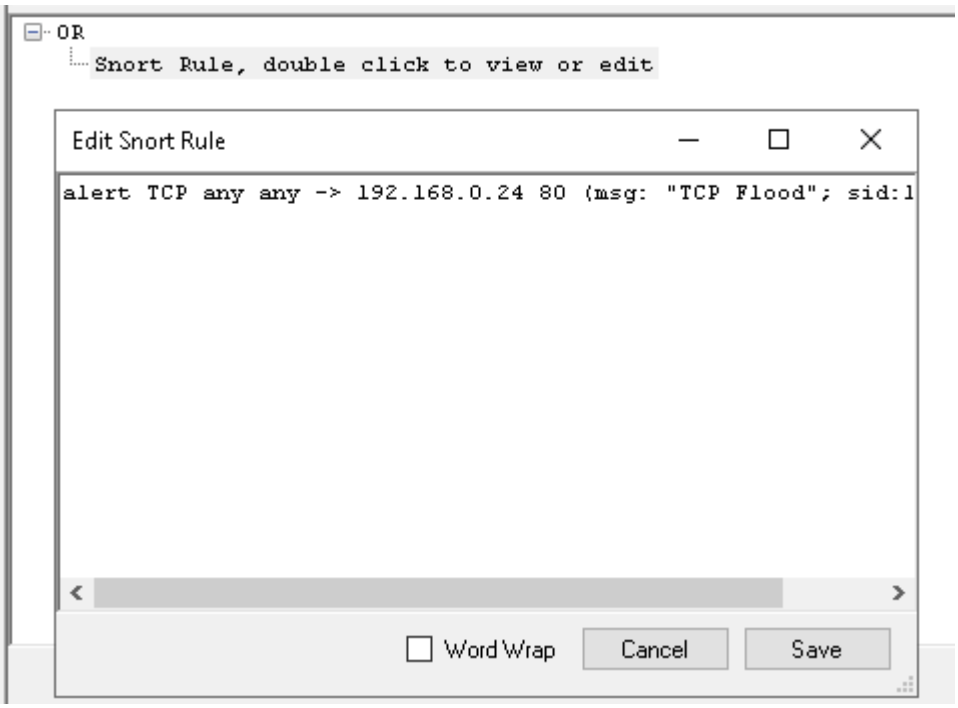
Fuente: www.fireeye.com

Esta herramienta ofrece la posibilidad de crear indicadores de compromiso teniendo en cuenta más de 20 criterios encontrados en los sistemas operativos. Pero como el objetivo es análisis del ataque en la red, se encontró una opción que pudo haber servido para la creación del indicador, pero a la hora de probarlo no funcionó.

La herramienta ofrece la opción de importar una regla snort (

Figura **19**), por lo tanto, eso se hizo

Figura 19 : Creación de indicador de compromiso en Mandiant IOCe



Fuente: Autor

La regla utilizada en snort es la “alert TCP any any -> 192.168.0.24 80 (msg: "TCP Flood"; sid:1000001;)” (Figura 20), una regla sencilla que debería alertarme cualquier tráfico proveniente hacia mi dispositivo en el puerto 80. El programa exporta un archivo tipo .IoC

Figura 20: Indicador de compromiso exportado “Snortdos.ioc”

Fuente: Autor

La forma de evaluar el indicador de compromiso creado es con otra herramienta llamada “IOC Finder”

Figura 21 : Herramienta IOC Finder

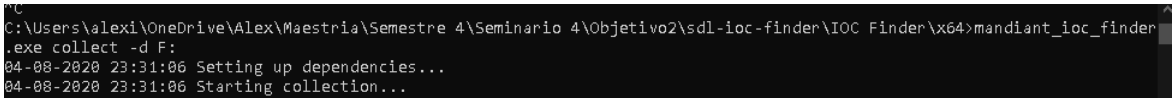
Fuente: www.fireeye.com

Esta herramienta es un ejecutable por consola (Figura 21), la cual pide dos pasos: recolectar la evidencia del indicador y reportar hallazgos de acuerdo al indicador seleccionado. Siguiendo los comandos establecidos en una guía creada en España se pudo replicar el reporte y el diagnostico [41].

Antes de iniciar los dos pasos, se crearon dos capturas de tráfico de red donde se evidencia una denegación de servicio mediante la modalidad de envió de paquetes tipo tcp Syn al puerto 80 con la IP 192.168.0.24 tal como se creó en el indicador de compromiso.

Con el comando “mandiant_ioc_finder.exe collect -d F: (Figura 22), recolectamos los logs de tráfico capturados en wireshark y Tcpdump.

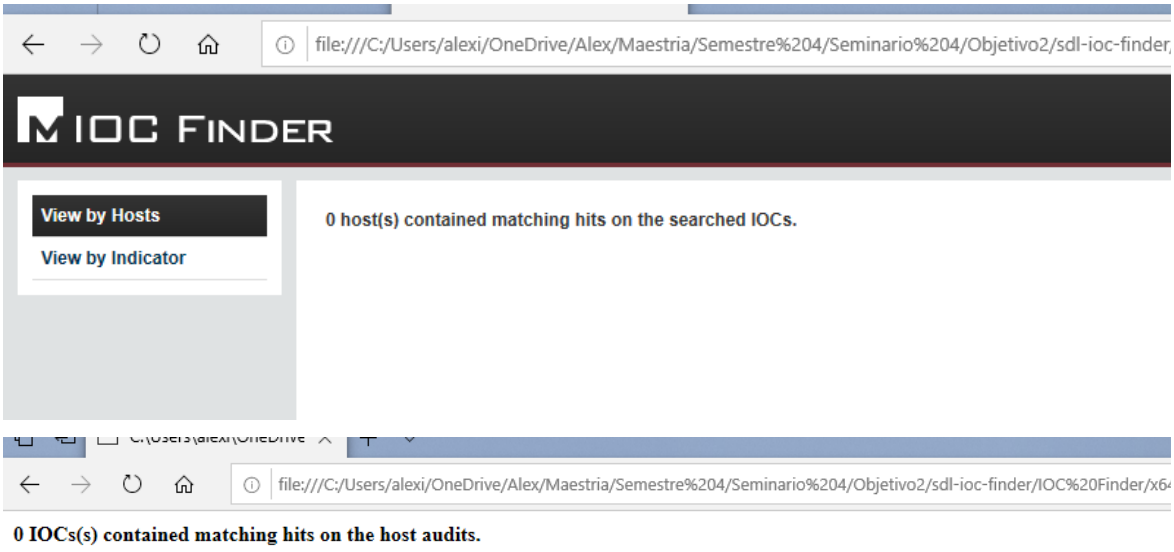
Figura 22 : Command prompt windows



Fuente: Autor

Lo siguiente fue generar el reporte con respecto a la recolección de datos que el software realizó (Figura 23). El reporte se comparó con respecto al indicador de compromiso generado en snort, el comando es “mandiant_ioc_finder.exe report –i snortdos.ioc –t html” , se utiliza para comparar la recolección de datos con el indicador y posteriormente exportarlo a html.

Figura 23 : Reporte IOC finder



Fuente: Autor

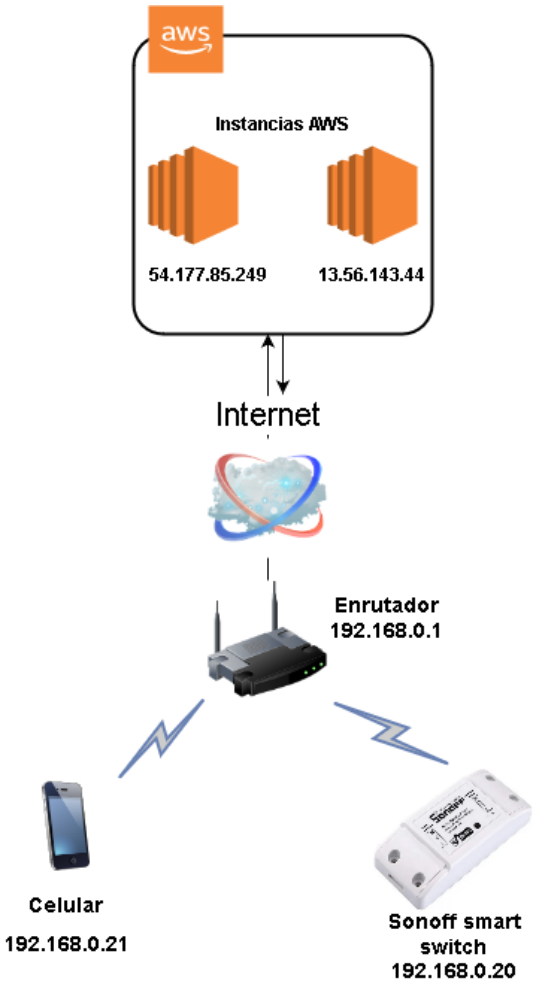
El resultado final fue un reporte en blanco indicando que no se encontró ninguna coincidencia del indicador de compromiso y los logs generados. Lo cual deja un par de preguntas, la herramienta de IOC Finder no fue diseñada para comparar las reglas snort o la herramienta no es capaz de analizar las tramas de wireshark/Tcpdump.

Como esta propuesta tampoco dio un buen resultado, se decidió replicar los ataques informáticos definidos en la fase 1 para poder identificar los indicadores de compromiso que se ven en la red en tiempo real. Con la capacidad de recrear los ataques podemos analizar el comportamiento de estos para más adelante en la fase 3 poder implementar los controles y las alertas necesarias para mitigarlos.

Funcionamiento del smart switch

Al estudiar el funcionamiento del Smart switch se pudo observar su modelo de funcionamiento, la arquitectura de red del Sonoff smart switch (Figura 24). Al realizar un rastreo de paquetes salientes y entrantes al dispositivo Smart switch se observa que este se comunica con dos servidores nube hosteados en Amazon.

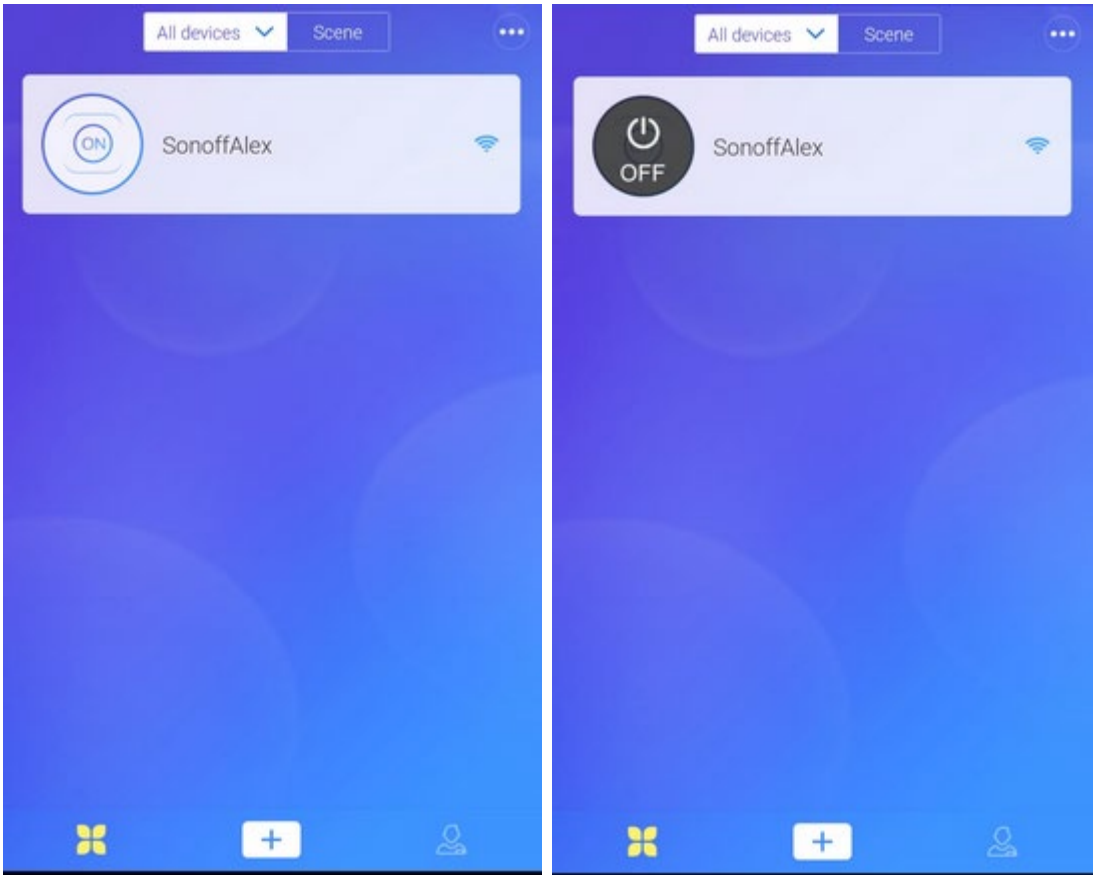
Figura 24 : Arquitectura red Sonoff smart switch



Fuente: Autor

Primero se analizó el comportamiento del celular que tiene instalada la aplicación que administra el dispositivo smart switch llamada “eWeLink” (Figura 25). La aplicación permite básicamente encender o apagar el dispositivo smart switch.

Figura 25 : Aplicación eWeLink

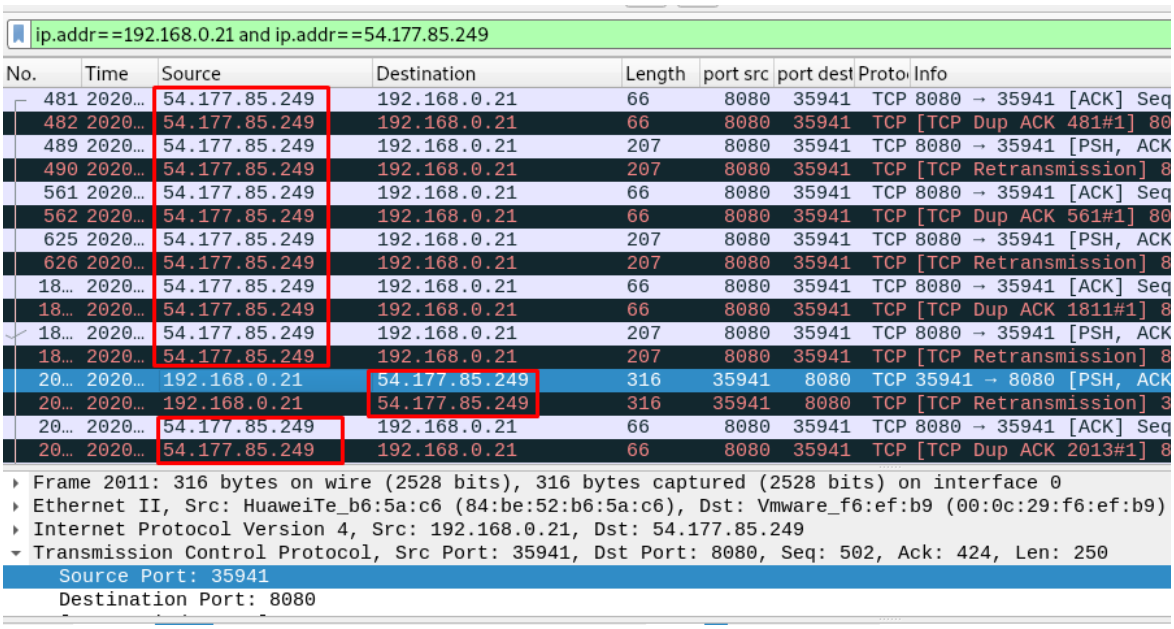


Fuente: Autor

Cuando se abre la aplicación y se envía el comando de encendido o apagado, el celular envía esta petición a un servidor web (

Figura 26).

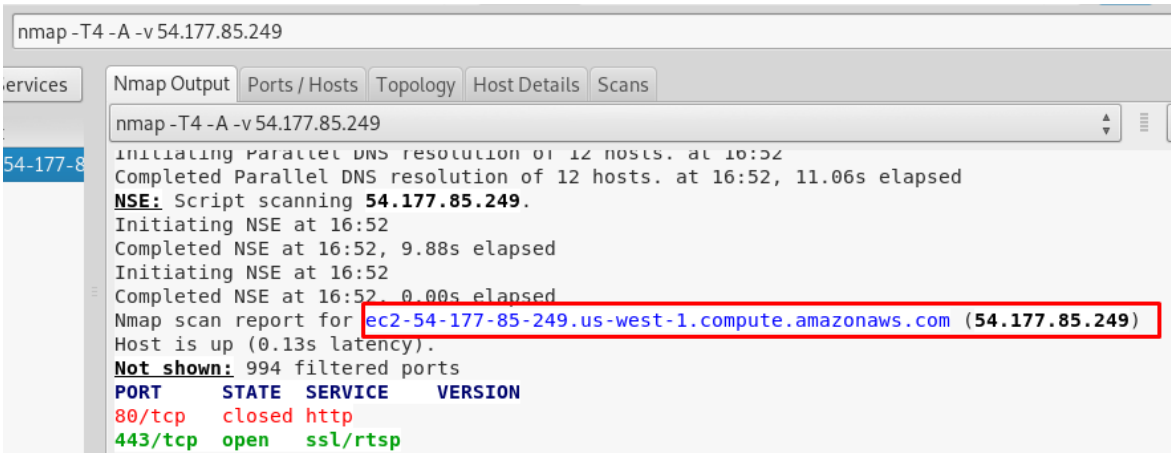
Figura 26 : Análisis de tráfico celular



Fuente: Autor

Al hacerle un escaneo de puerto a esta IP **54.177.85.249** se encontró que corresponde a la IP de un servidor en la nube de Amazon (Figura 27).

Figura 27 : Escaneo IP servidor Amazon

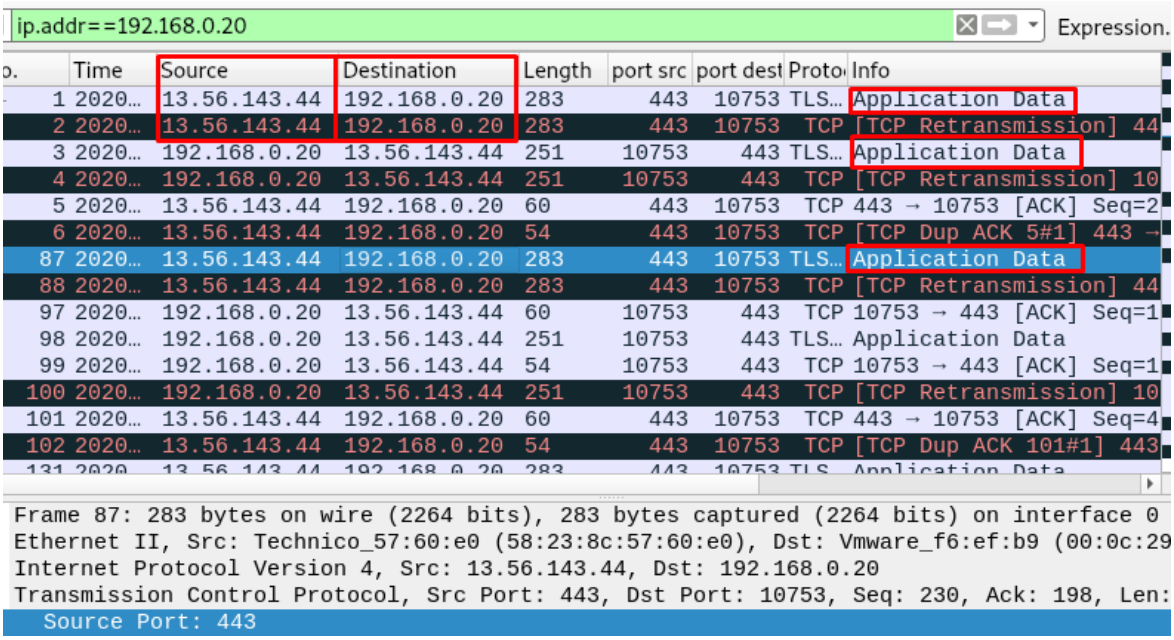


Fuente: Autor

Tiene varios puertos abiertos, pero esto no fue de nuestro interés, solo nos importa saber cómo funciona este dispositivo en la red IoT.

Lo siguiente fue hacerle seguimiento al smart switch, para saber cómo le llega a este la petición de encender o apagar.

Figura 28 : Análisis de tráfico al smart switch

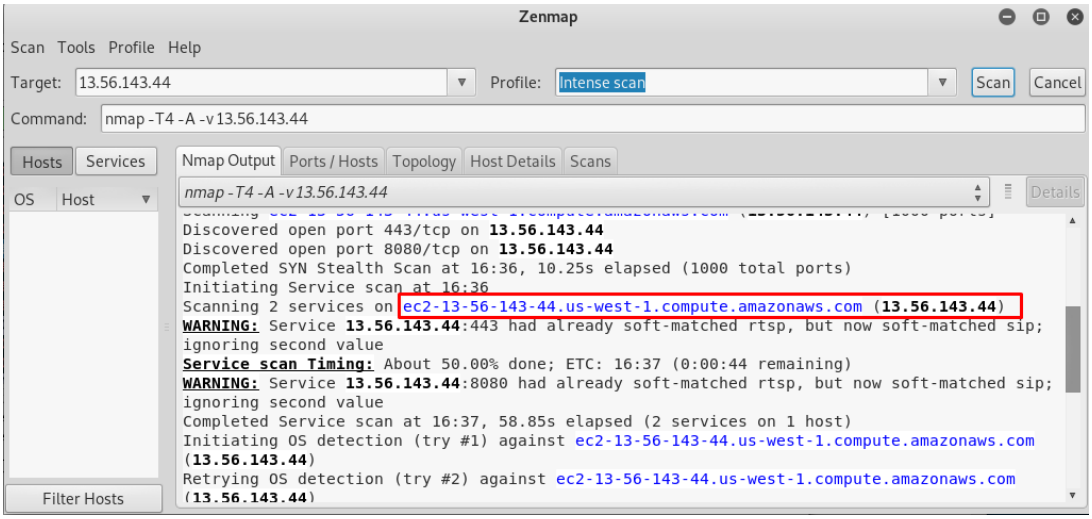


Fuente: Autor

Como se puede observar (Figura 28), cuando se lanza el comando desde el celular para activar el smart switch, primero el celular envía la petición al servidor web (**54.177.85.249**) y luego otro servidor web (**13.56.143.44**) le reenvía la petición al smart switch en su puerto 443 encriptado con TLS mediante un paquete de application data.

Se procedió a escanear este otro servidor web y se encontró que también es una maquina en la nube de AWS (Figura 29), por lo tanto, el dispositivo se comunica con dos servidores aparentemente.

Figura 29 : Análisis servidor web



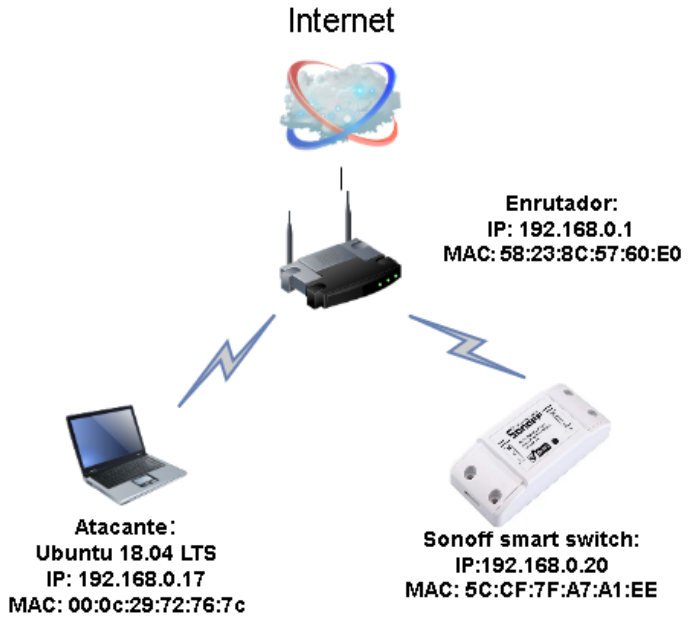
Fuente: Autor

Una vez obtenida la información del funcionamiento del dispositivo se procedió a realizar los ataques informáticos definidos en la primera fase.

Indicador de Compromiso ataque de Des-autenticación

Arquitectura red para el ataque propuesta Figura 30.

Figura 30 : Arquitectura de red ataque des autenticación

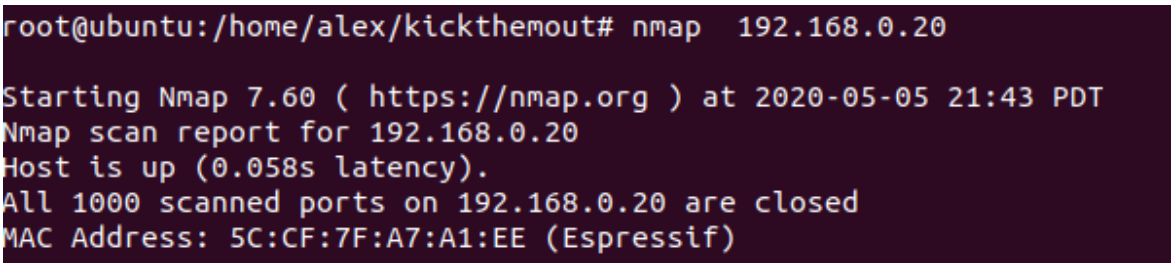


Fuente: Autor

Gracias al análisis del funcionamiento del smart switch, sabemos que, si le deshabilitamos la salida a internet al dispositivo smart switch, entonces le realizaríamos una denegación de servicio. Aquí es donde el ataque de desautenticación tiene un gran impacto pues lo que busca este ataque es quitar a un dispositivo de la red inalámbrica y dejarlo sin conexión a esta.

Para aprovechar este ataque se utilizó el software llamado kickthemout, cuyo repositorio y documentación se puede encontrar en www,github.com [42]. Este software es de libre uso y desarrollado para las plataformas de Linux. Lo que dicen los desarrolladores es que fue creado para sacar a todos los dispositivos de la red wifi y quedarse con el ancho de banda para uno solo "A tool to kick devices out of your network and enjoy all the bandwidth for yourself. It allows you to select specific or all devices and ARP spoofs them off your local area network".
Primero que todo se escaneó la red para verificar la IP del dispositivo smart switch (Figura 31),

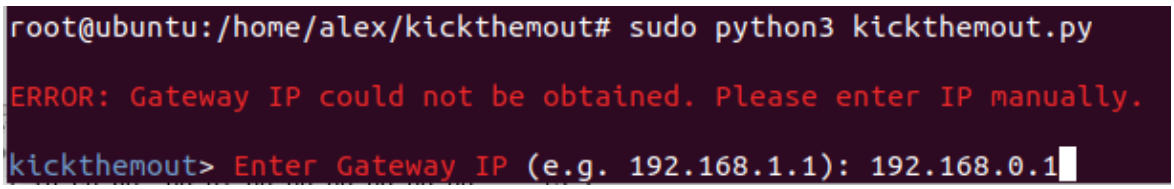
Figura 31 : Escaneo de red con nmap



Fuente: Autor

Y efectivamente validamos que la IP del smart switch es la 192.168.0.20 toda vez tiene la dirección MAC previamente definida. A continuación, iniciamos la aplicación, con la ayuda del framework de Python en su versión 3.0 se ejecuta el comando para iniciar el software "kickthemout" (Figura 32)

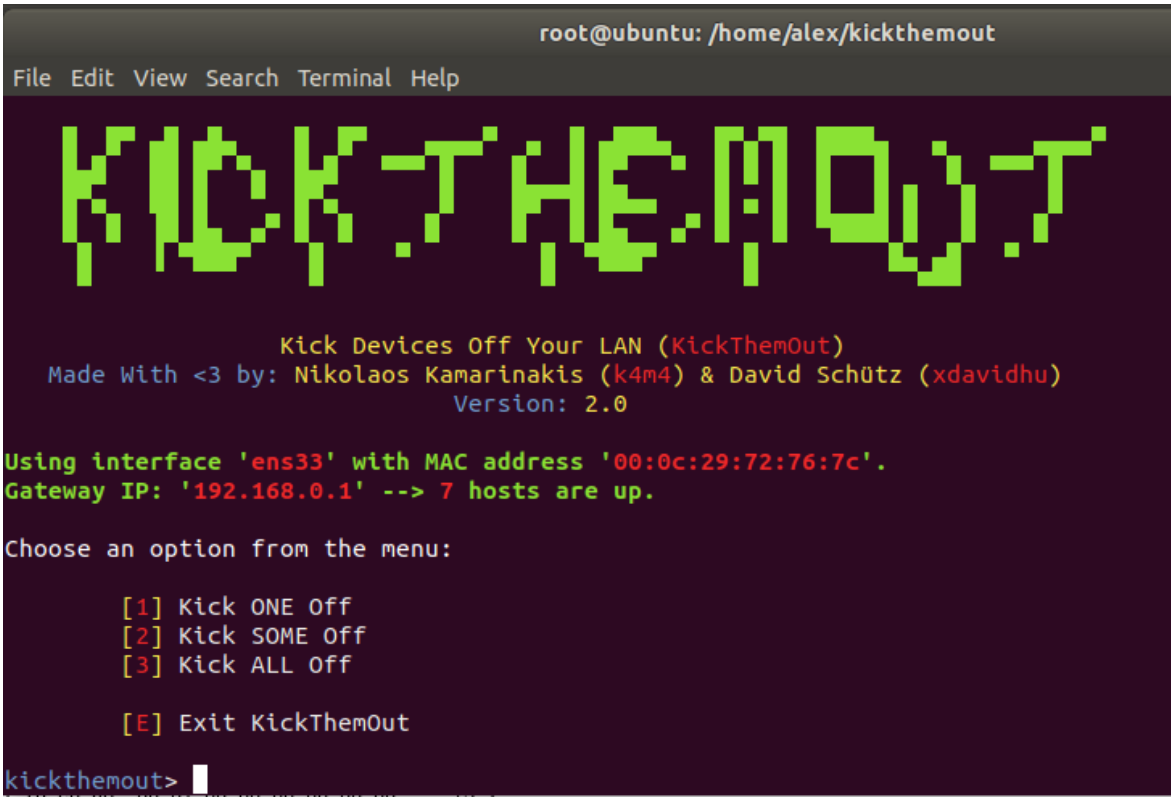
Figura 32 : Inicio de software kickthemou



Fuente: Autor

Se le debe indicar al software cual es la puerta de enlace, en este caso es la 192.168.0.1 el cual corresponde al enrutador local (Figura 33).

Figura 33 : Puerta de enlace kickthemout

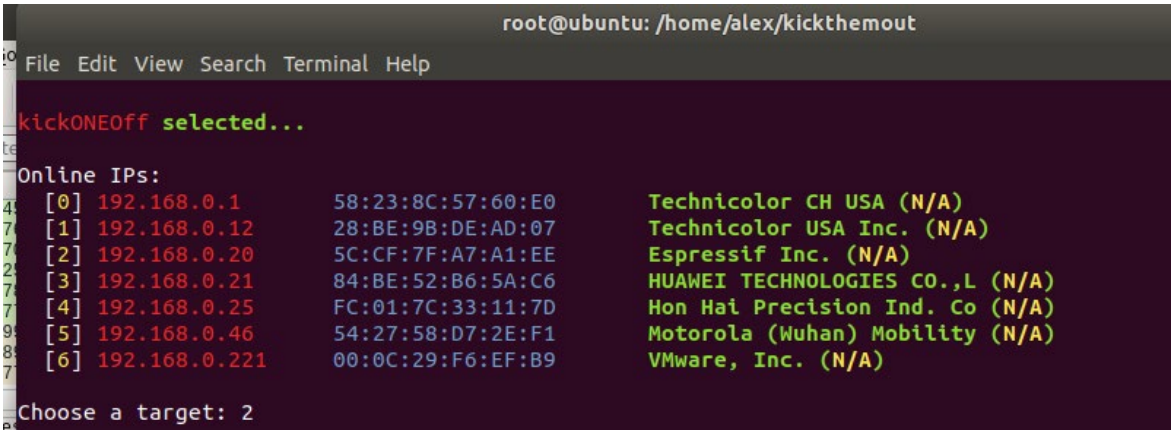


Fuente: Autor

Una vez iniciado el programa aparece la información de mi dirección MAC, mi interfaz de salida, la cantidad de dispositivos encontrados en la red y un dialogo al cual seleccionamos la primera opción (

Figura **34**), quiere decir que vamos a quitar solamente a un dispositivo de la red.

Figura 34 : Selección de victima



Fuente: Autor

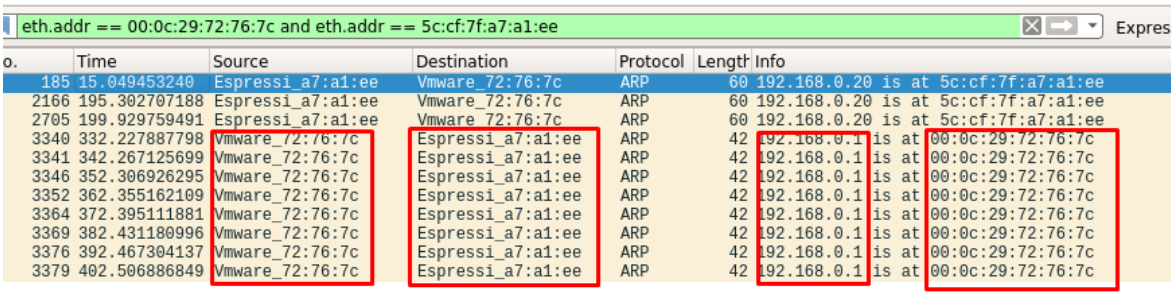
Ahora en la nueva ventana, el software muestra los dispositivos en la red, escogemos la opción 2 toda vez este corresponde al smart switch (Figura 35).

Figura 35 : Inicio del ataque

Fuente: Autor

Como se puede evidenciar, el ataque se ejecutó, ahora revisamos mediante un capturador de paquetes de red, en este caso wireshark (Figura 36), lo que está ocurriendo en la red con el dispositivo.

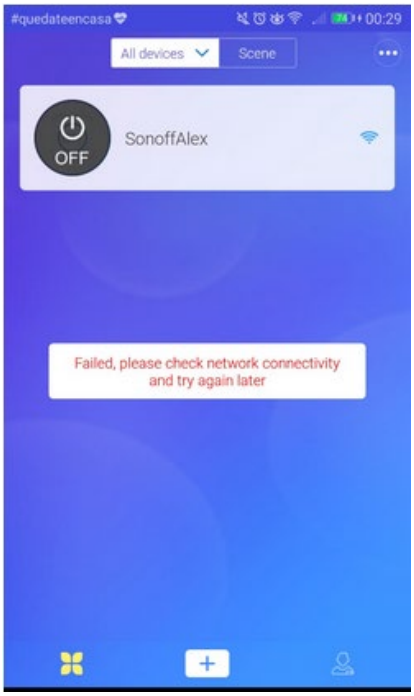
Figura 36 : Captura de tráfico wireshark



Fuente: Autor

Como se puede observar, el ataque lo que hizo fue un envenenamiento de la tabla ARP del smart switch. La máquina Linux atacante conocida como (Vmware_72:76:7c) le envió peticiones con el protocolo ARP al smart switch (Espressi_a7:a1:ee) envenenando su tabla ARP. El atacante le hace creer al smart switch que el enrutador con la IP 192.168.0.1 está ubicado en la dirección MAC 00:0c:29:72:76:7c, y así engañó al smart switch para que creyera que el enrutador con la IP 192.168.0.1 ahora es el atacante Linux, de esta manera envenena la tabla ARP del smart switch dándole información falsa sobre la dirección IP y MAC real del enrutador.

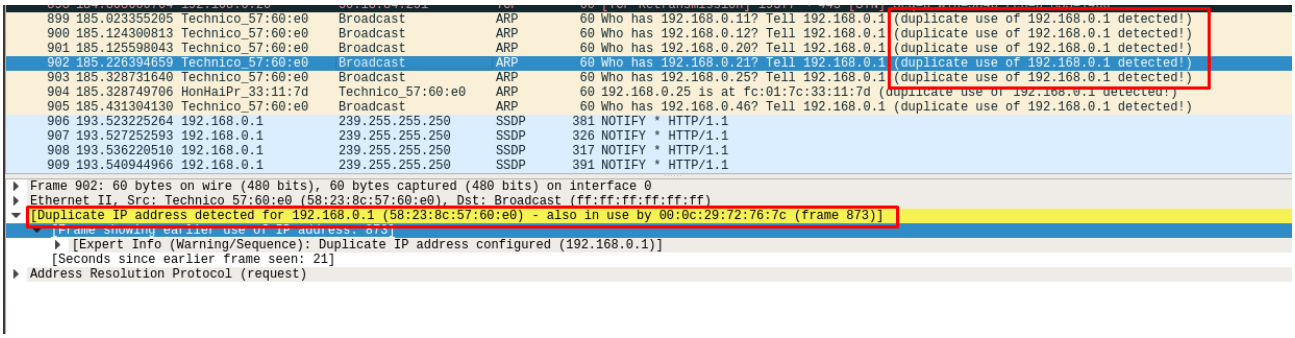
Figura 37 : Estado de desconexión del dispositivo



Fuente: Autor

Así queda la interfaz de la aplicación una vez se realiza el ataque (Figura 37), esta no es capaz de encontrar al dispositivo e indica que existen problemas en la red wifi a pesar de que el celular si se encuentra conectado en ella.

Figura 38 : Captura de tráfico wireshark con alerta



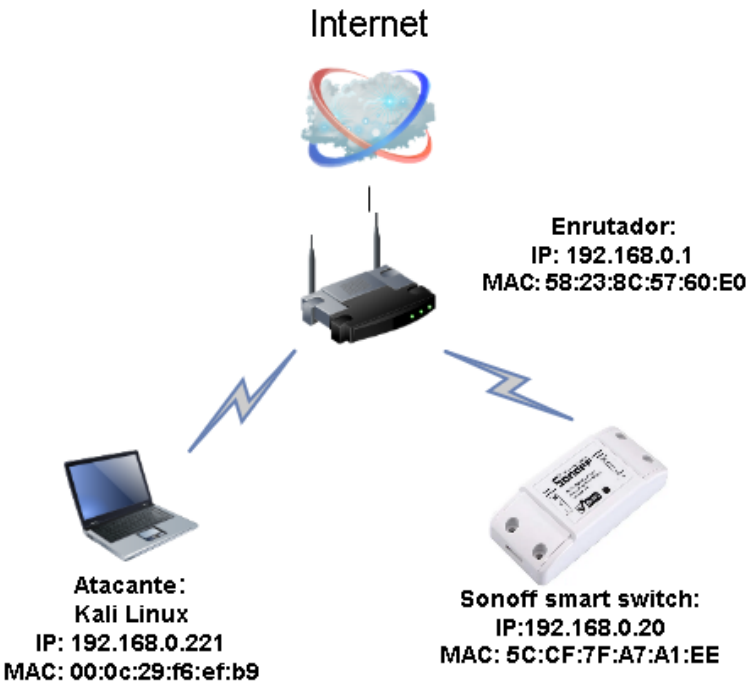
Fuente: Autor

Al revisar el tráfico de red general se encontraron unas alertas dadas por el software de wireshark (Figura 38). La alerta indica que hay un duplicado de dirección IP 192.168.0.1 la cual corresponde al enrutador. Estos paquetes alertados en la red, con el mensaje en amarillo, correspondientes al protocolo ARP son nuestro indicador de compromiso para poder detectar futuros ataques.

Indicador de Compromiso ataque DoS por flooding

Para la ejecución de este ataque se utilizó el sistema operativo de Kali Linux y se probaron diferentes tipos de flood de tráfico, todos con éxito, aquí la arquitectura de red Figura 39

Figura 39 : Arquitectura de red



Fuente: Autor

Con base en la investigación realizada sobre métodos para denegación de servicio en sistemas IoT [27] se realizó el primer ataque con el comando hping3 (Figura 40). Hping3 es una herramienta orientada a ensamblar y analizar paquetes TCP/IP.

"hping3 -S -w 64 -p 80 --flood --rand-source 192.168.0.20"

Figura 40 : Ataque con hping3

```
root@kali:~# hping3  -S -w 64 -p 80 --flood --rand-source 192.168.0.20
HPING 192.168.0.20 (eth0 192.168.0.20): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

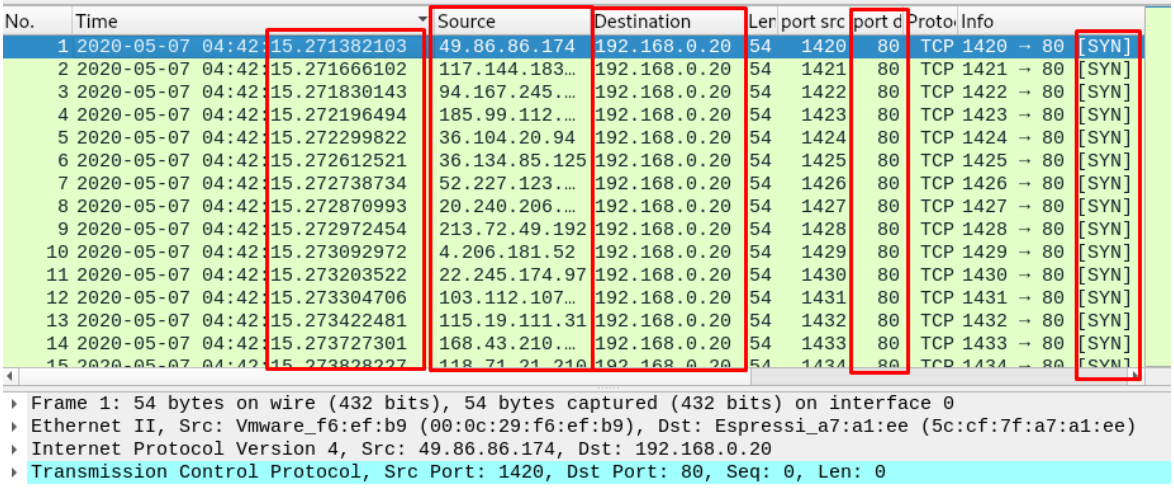
Fuente: Autor

Los parámetros que se utilizaron fueron el "-S" el cual indica envió de paquetes tipo SYN, "-w" significa el tamaño configurado del Windows size del paquete, "-p" indica el puerto a atacar, "—flood" indica el inundamiento de paquetes a gran velocidad e ilimitados y el "–rand-source" indica direcciones IP aleatorias.

En este caso no hace falta indicar un tamaño de los paquetes enviados toda vez el smart switch falla con el tamaño por defecto el cual es 54 bytes por paquete.

En todas las pruebas realizadas se encuentra que el switch es demasiado vulnerable a los ataques por inundación pues este empieza a fallar desde los primeros 1000 o incluso menos paquetes enviados en el modo "flood", si no se indica el modo "flood" el dispositivo funciona correctamente y el ataque no es efectivo. Esto se debe a que el modo "flood" envía miles de paquetes por segundo y si no se configura entonces el programa envía paquetes a la velocidad de uno por segundo.

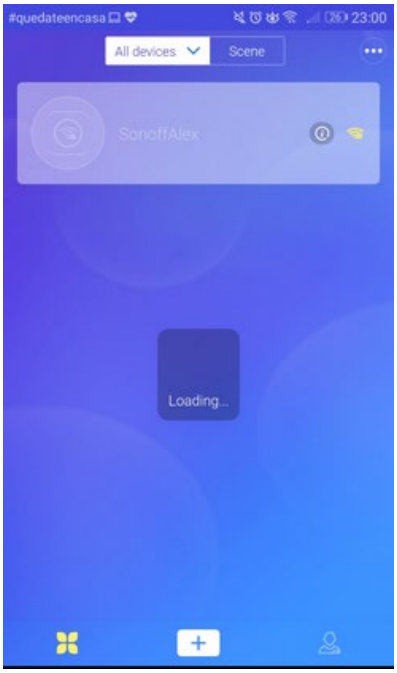
Figura 41 : Captura de tráfico del ataque hping3



Fuente: Autor

Como se puede observar en la captura (Figura 41), el intervalo de tiempo entre cada paquete enviado son milésimas de segundo, esto es lo que causa la denegación del servicio, la velocidad del envió y la cantidad de paquetes enviados. Se puede observar las direcciones IP aleatorias elegidas por hping3, el destino el cual fue el smart switch, el puerto 80 y el paquete SYN.

Figura 42 : Aplicación “eWeLink” sin servicio



Fuente: Autor

La aplicación del celular no encuentra al dispositivo smart switch durante el ataque por hping3 (Figura 42), lo identifica como desconectado o apagado.

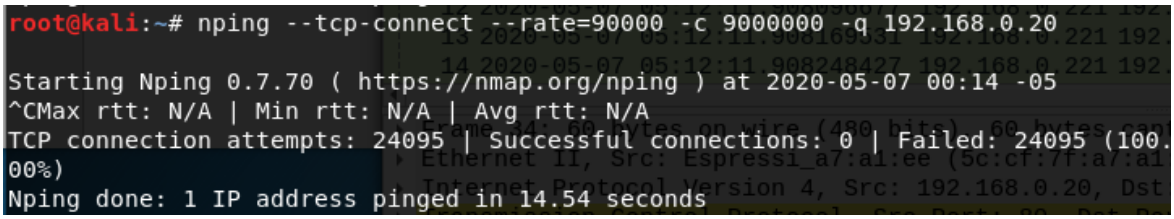
El indicador de compromiso analizado en este ataque es la velocidad con la que se envían los paquetes, la cantidad de paquetes en un lapso de tiempo mínimo, y las IP que algunas son muy sospechosas y generan alertas.

El siguiente ataque en probarse fue también tomado de la investigación en ataques de denegación de servicio [27], es con una herramienta llamada “nping” (

Figura 43). Nping es otro programa orientado a la creación de paquetes de red, el comando utilizado fue

“nping --tcp-connect --rate=90000 -c 9000000 -q 192.168.0.20”

Figura 43 : Ataque con nping



Fuente: Autor

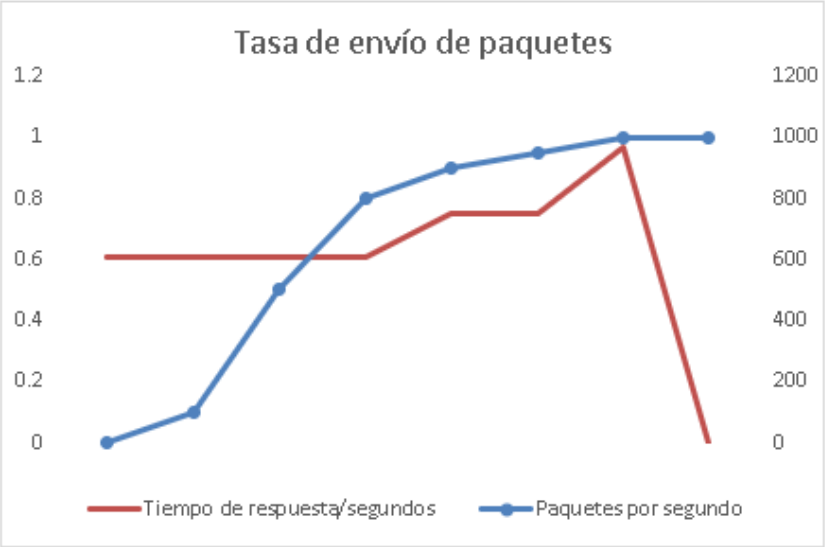
Los parámetros utilizados "—tcp-connect" indica que se conectara con el protocolo tcp, "—rate=90000" indica la tasa de paquetes enviados en un segundo, "-c" indica la cantidad total de paquetes a enviar y "-q" indica que el programa muestre poca información durante su ejecución.

Uno de los hallazgos establecidos fue que el comando no es efectivo sin el parámetro de "--rate" toda vez por defecto la cantidad de paquetes enviados por segundo es de 1 lo cual no fue suficiente para hacer caer al dispositivo.

Al realizar varias pruebas de velocidad en la tasa de los paquetes enviados se encontró que el dispositivo falla y se le niega su servicio es a partir de y exactamente de 1001 paquetes por segundo. Si se configura el nping con el parámetro "—rate=1000" o menor, el comando no es capaz de afectar el funcionamiento del smart switch, pero cuando se lanza el comando con una tasa de envío de 1001 paquetes o más por segundo si deja de funcionar correctamente "—rate=1001". Sin importar la cantidad total de paquetes enviados, el dispositivo deja de funcionar es a partir de una tasa de envío de 1001 paquetes por segundo.

Figura 44 : Gráfica ilustrando pruebas de envío por segundo

Paquetes por segundo	Tiempo de respuesta /segundos
0	0.61
100	0.61
500	0.61
800	0.61
900	0.75
950	0.75
1000	0.97
1001	0

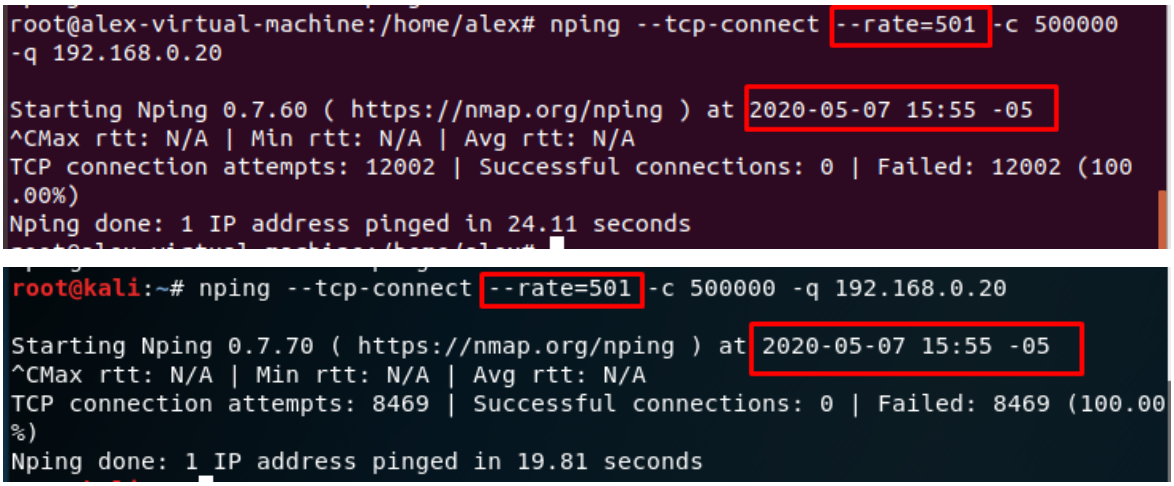


Fuente: Autor

Tras descubrir este hallazgo, se decidió hacer una gráfica que ilustrara mejor el tiempo de respuesta del smart switch contra los paquetes por segundo enviado (Figura 44). Por defecto el smart switch responde al comando de encendido/apagado en 0.61 segundos, cuando los paquetes se acercan cada vez más al umbral de 1001 paquetes por segundo va disminuyendo solo un poco en su tiempo de respuesta. Al enviar 1000 paquetes por segundo el dispositivo se retarda en responder un 59% más pero no se percibe toda vez el tiempo de respuesta sigue siendo inferior a un segundo. Cuando finalmente se envían los 1001 paquetes por segundo ya no hay tiempo de respuesta puesto que el dispositivo no responde.

Como prueba adicional se decidió realizar el ataque con una segunda maquina atacante (Figura 45), de esta manera tener dos hping3 simultáneos sobre el smart switch.

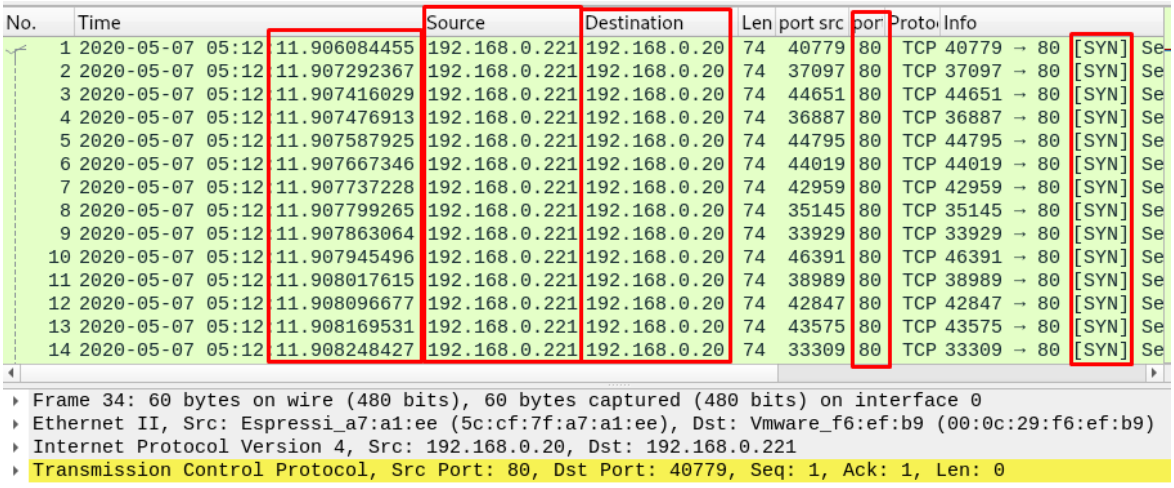
Figura 45 : Prueba con dos atacantes



Fuente: Autor

Se encontró que al realizar el ataque simultáneo configurando la tasa de paquetes enviados en 500 el dispositivo sigue funcionando tal como se había aclarado en la prueba anterior (Figura 45) . Pero cuando ambos ataques simultáneos se configuran en 501 paquetes por segundo ya el dispositivo falla y se le niega el servicio, esto debido a que ambos ataques están enviando 1002 paquetes por segundo y ya habíamos descubierto que el umbral tolerable era de 1000 paquetes por segundos enviados hacia el smart switch.

Figura 46 : Captura de tráfico del ataque nping



Fuente: Autor

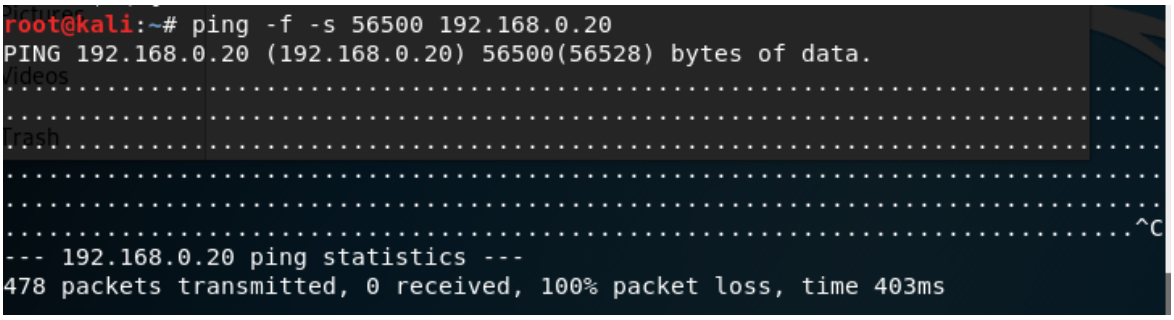
En esta captura de paquetes con wireshark (Figura 46)se puede ver la gran velocidad con el que el programa envía estos paquetes de inundación, que son milisegundos de diferencia. Se establece la IP del atacante y la IP del dispositivo smart switch, el puerto 80 viene configurado por defecto y son paquetes de tipo SYN. En la aplicación se ve el mismo panorama como sucedió con el ataque por hping3.

El indicador de compromiso será la velocidad tan corta entre cada paquete, la cantidad total de paquetes enviados en un lapso de tiempo y el exceso de peticiones SYN de una misma IP.

Por último, se probó un ataque de inundación de paquetes utilizando el comando “ping” (Figura 47). “Ping” es un comando nativo de Windows y de Linux que se utiliza principalmente para verificar conexión entre dos puntos, A y B. En este caso, al cambiar un poco los parámetros lograremos realizar una denegación de servicio al dispositivo smart switch.

“ping -f -s 56500 192.168.1.100”

Figura 47 : Ataque con ping

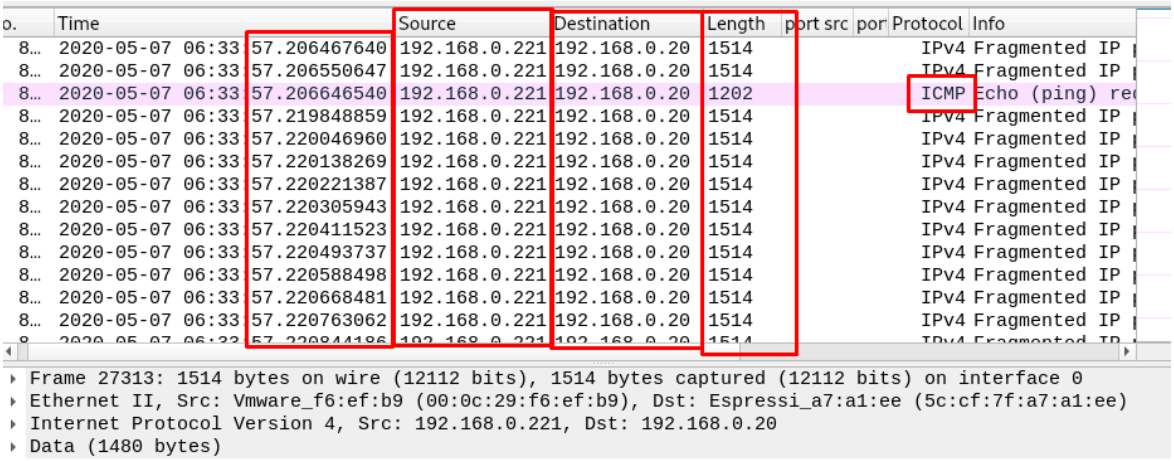


Fuente: Autor

Donde el parámetro “-f” indica flood o inundación de paquetes a toda velocidad, “-s” indica el tamaño del paquete y por defecto corresponde a 56 bytes y la IP de la víctima.

Durante las pruebas se encontró que el dispositivo tolera una inundación de paquetes ping hasta un tamaño de 30000 bytes, a partir de aquí ya el dispositivo deja de funcionar. Si el ataque se configura con la variable “-s” menor a 30000 el ataque no hará efecto en el dispositivo.

Figura 48 : Captura de tráfico ataque Ping



Fuente: Autor

La captura del tráfico en wireshark (Figura 48) mostró la velocidad de envío de paquetes, y como en los anteriores ataques es de milisegundos, es importante esta velocidad para el funcionamiento del ataque. Se observó la IP del atacante y la víctima, el protocolo que se observa es el ICMP y los paquetes están fragmentados toda vez el tamaño es demasiado grande para uno solo.

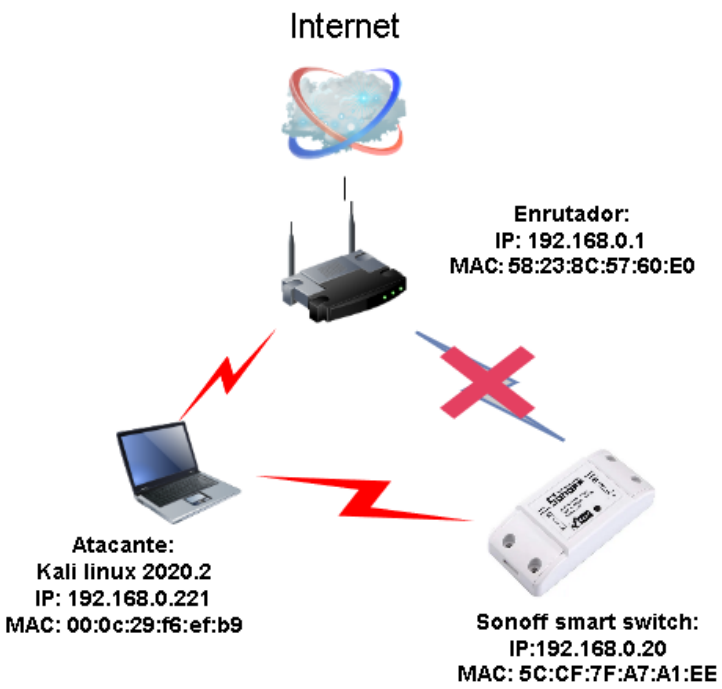
En la aplicación de “eWeLink” el dispositivo muestra el mismo mensaje indicando que no se encuentra el smart switch y que existen problemas de red.

El indicador de compromiso para este ataque es la velocidad con la que se reciben los paquetes, la cantidad total de paquetes enviados en tan poco tiempo, el tamaño irregular de los paquetes y por último la IP del mismo destinatario que no cambia.

Indicador de Compromiso ataque Man in the middle

La arquitectura para la ejecución de este ataque fue la siguiente Figura 49

Figura 49 : Arquitectura ataque de Man in the Middle



Fuente: Autor

Con el montaje de este ataque se logró que la comunicación entre el smart switch y el enrutador fuera interceptada sin que el dispositivo smart switch tenga conocimiento de esto.

Utilizando la técnica de hombre en el medio fue posible realizar una denegación del servicio del dispositivo smart switch. En el blog [43] el autor nos comparte una técnica de denegación de servicio con la herramienta ettercap mediante un script realizado por él. Ettercap es una herramienta desarrollada para realizar ataques de hombre en el medio, interceptar comunicaciones de dispositivos, analizar los dispositivos de red y otras operaciones de hacking que se pueden hacer con él.

Fue necesario construir un script personalizado ya que con la herramienta de Ettercap, el ataque de hombre en el medio por si solo no causa una denegación del servicio, pero con el script propuesto sí. Con este script el software de Ettercap logra interceptar el Smart switch logrando que redirija su tráfico al atacante y una vez se redirige el tráfico de red este se corta, haciendo que el tráfico de la red del dispositivo no llegue a su destinatario.

Lo primero fue construir el script propuesto en el blog (Figura 50), se debe guardar en la ruta de ettercap **/usr/share/ettercap.**

Figura 50 : Script para ettercap

*dos.elt *Untitled Document 1

```
if (ip.src == '192.168.0.20' || ip.dst == '192.168.0.20')
{
drop();
kill();
msg("Packet Dropped\n");
}
```

Fuente: Autor

El script es realizado en un idioma bastante simple y directo, básicamente está construido por un condicional "if" el cual pregunta si el paquete tiene de origen la ip "192.168.0.20" (el smart switch) o de destino, si el paquete cumple con este condicional entonces el script va a matar este paquete mediante un RST dado por el comando drop() y kill(). El último paso dentro del script es mostrar el resultado en la consola mediante el mensaje de "packet dropped" o "paquete tumbado".

El siguiente paso es compilar el script con ettercap para que este lo pueda entender, el mismo ettercap trae un compilador, se debe abrir la consola y ejecutar el comando

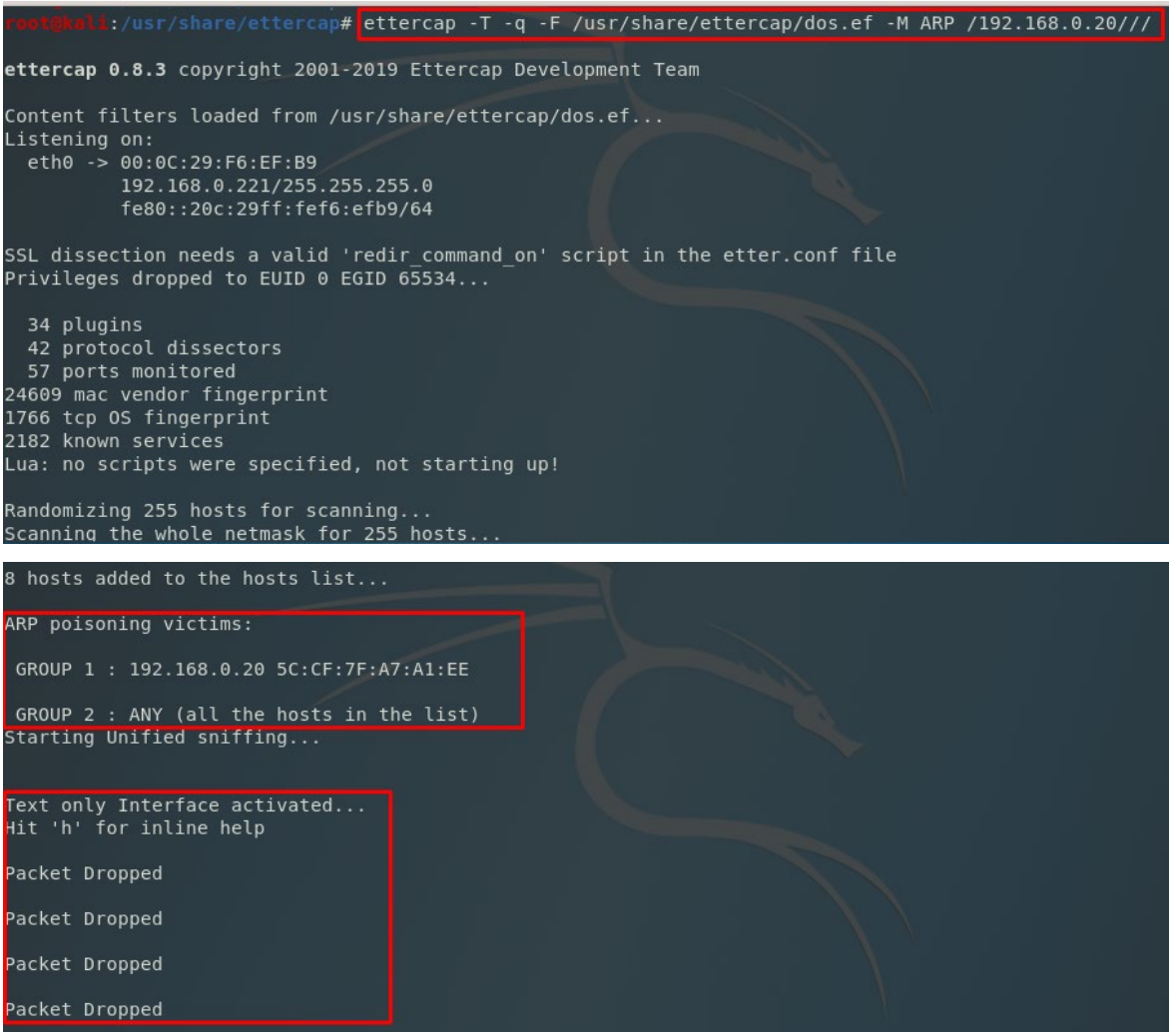
etterfilter dos.elt -o dos.ef

Lo que hizo este comando fue compilar el script guardado como "dos.elt" y ahora ettercap lo convirtió en su propio lenguaje con el nombre de "dos.ef". Ya con el archivo compilado se le puede decir a ettercap que lo utilice directamente. Por último, construimos el comando de ataque, el cual es

ettercap -T -q -F /usr/share/ettercap/dos.ef -M ARP /192.168.0.20///

Donde "-T" indica el modo silencioso en la consola para que no nos llene de mensajes, "-q" para cargar el script, "-F" para ubicar la ruta del script, "-M" para ejecutar un ataque de hombre en el medio por envenenamiento de ARP y por último nuestra víctima 192.168.0.20 (el smart switch), los /// son importantes como requisito del comando.

Figura 51 : Ejecución de ataque por mitm



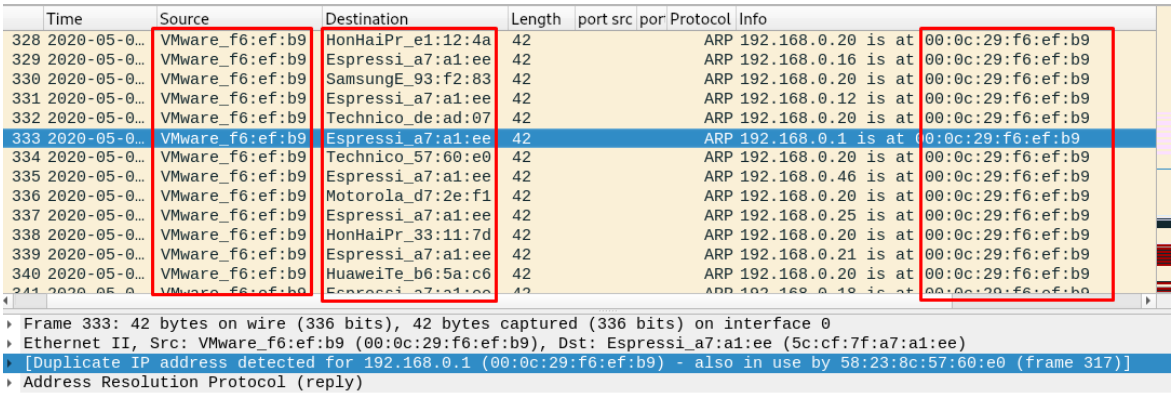
Fuente: Autor

Como se puede evidenciar con el lanzamiento del ataque (

Figura **51**), el ettercap primero busca y encuentra todos los hosts dentro de la red local, en este caso encontró 8. Lo siguiente es que el ettercap crea dos grupos, GROUP 1 y GROUP2. En el group 1 se incluye a la víctima y en el group 2 se incluyen a todos los otros dispositivos de la red. Luego envenena la tabla ARP de nuestra víctima, el smart switch, y la manipula haciéndolo creer que todos los host que tiene guardado en ella ahora se encuentran con la dirección MAC del atacante 00:0c:29:f6:ef:b9. Ahora a los dispositivos del group 2, les envenena la tabla ARP y les hace creer que el smart switch ahora tiene la dirección MAC y la IP del atacante Kali Linux. De esta manera el

tráfico dirigido saliente del smart switch llega al Kali Linux y el tráfico entrante hacia el smart switch también llega al Kali Linux.

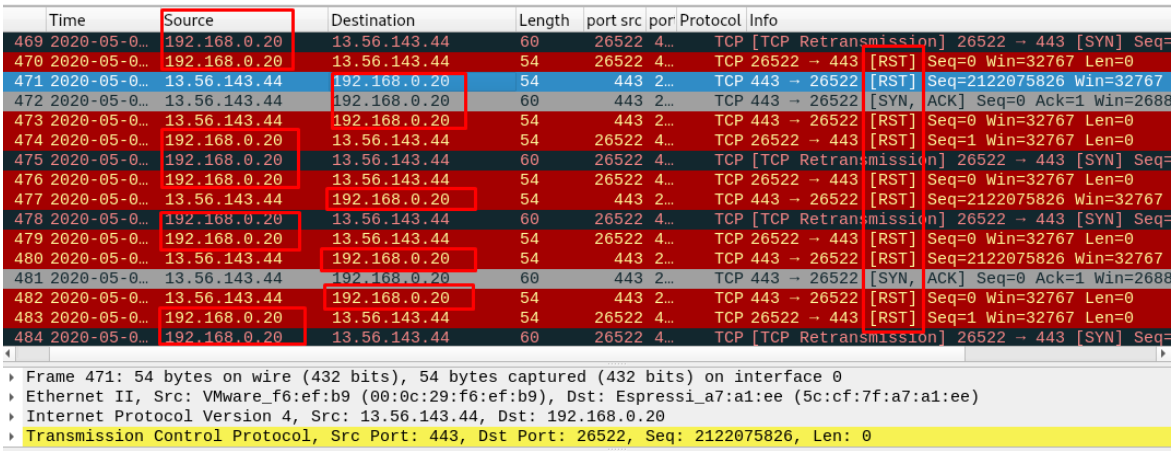
Figura 52 : Captura de paquetes con ataque de Mitm



Fuente: Autor

Como se puede observar (Figura 52), el atacante envía paquetes con el protocolo ARP a todos los hosts de la red y les dice que el dispositivo con la IP 192.168.0.20 ahora tiene la dirección MAC del atacante Kali Linux y al Sonoff smart switch le dice que todos los hosts que él conoce ahora tienen la MAC del atacante. Este constituye la primera fase del ataque y es aquí donde se realiza el hombre en el medio.

Figura 53 : Captura de paquetes RST con ataque de Mitm

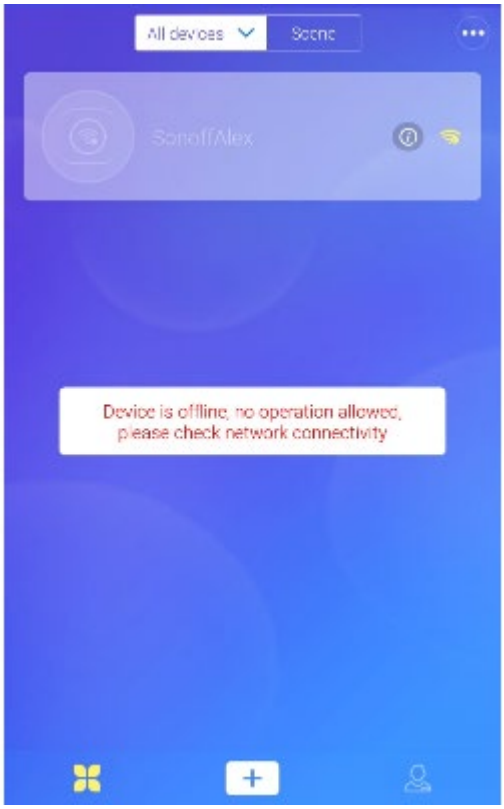


Fuente: Autor

En esta evidencia podemos observar cómo se aplica la segunda fase del ataque (Figura 53), la cual es dar de baja a todos los paquetes que salen o entran hacía la IP 192.168.0.20, en otras palabras, hacia o desde del smart switch mediante la utilización de la bandera RST. En la Figura 51 se evidencia como tras ejecutar el script, en la interfaz de la consola se observa el mensaje “Packet dropped”, lo que indica que el paquete fue “reseteado” o RST. La bandera RST lo que hace es reiniciar la conexión de manera abrupta, causando una interrupción de tráfico en la comunicación del paquete en la red.

Como complemento, en la columna de Info (Figura 53) se observa que la conexión fue reseteada, por lo tanto, el dispositivo se quedó sin comunicación con el mundo exterior, aquí es donde se evidencia la denegación del servicio con el exterior, con el internet. Si observamos bien una de las IP que intenta crear la conexión con el smart switch corresponde al servidor web de Amazon descubierto anteriormente, 13.56.143.44, si el dispositivo no tiene comunicación con internet entonces no se puede comunicar con él y por ende no funciona el dispositivo.

Figura 54 : Resultado visto en el aplicativo “eWelink”



Fuente: Autor

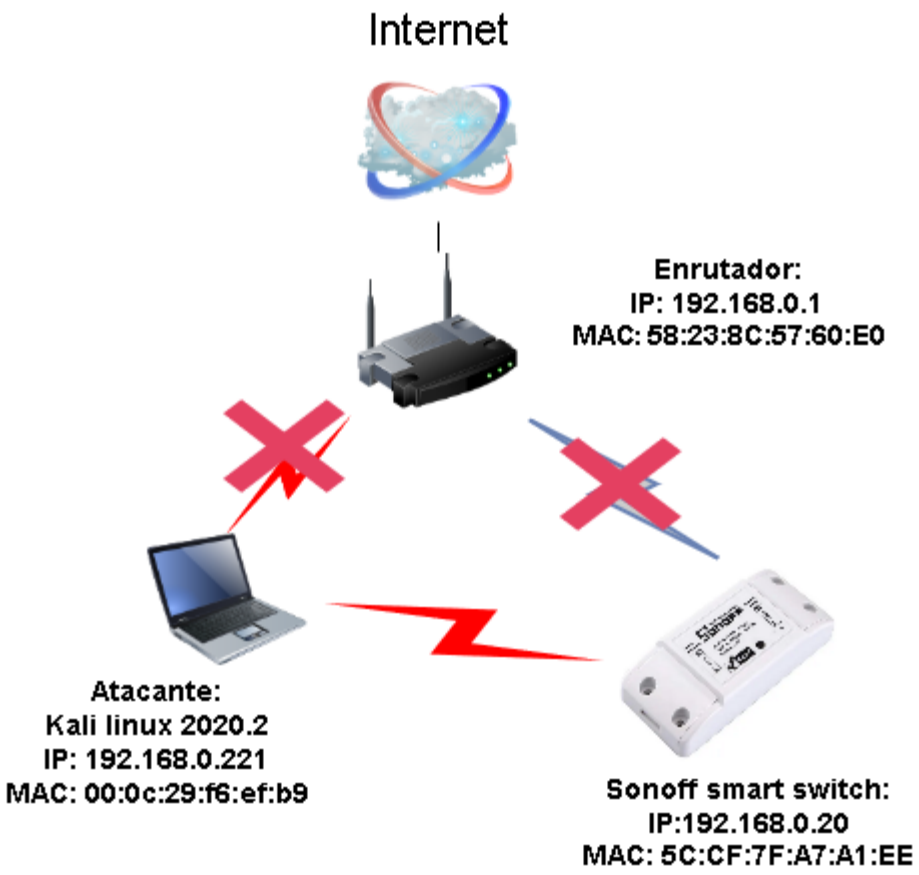
Así aparece en la aplicación el resultado del ataque (Figura 54), esta piensa que el dispositivo se encuentra apagado o desconectado de la red.

Este ataque deja mucho rastro, evidencia en la red y por lo tanto es fácil de evidenciar su indicador de compromiso. El atacante lanza constantemente paquetes con el protocolo ARP hacia todos los hosts de la red, el atacante corta las conexiones entrantes/salientes utilizando la bandera de los paquetes tcp denominada RST lo cual no es un comportamiento normal y en el tráfico de la red se evidencia un escaneo de todos los hosts que se encuentran en esta.

Indicador de Compromiso ataque ARP poisoning

La arquitectura para la ejecución de este ataque fue la siguiente (Figura 55)

Figura 55 : Arquitectura ataque de ARP poisoning



Fuente: Autor

Lo que ocurre con este ataque es que el dispositivo smart switch es engañado y piensa que el atacante corresponde al enrutador, por lo tanto, dirige su tráfico a este. Pero como el atacante no es ningún enrutador el dispositivo se queda sin comunicación con el mundo exterior y por lo tanto se le niega el servicio.

Para la realización de este ataque se decidió implementar la herramienta de bettercap [44], conocida como la navaja suiza para ataques en WiFi, Bluetooth Low Energy, secuestro de información inalámbrico y reconocimiento de redes incluso para MITM. Con esta herramienta se realizó el ARP spoofing.

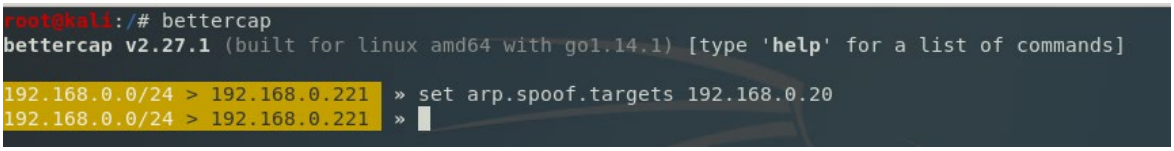
Bettercap ya viene instalado en Kali Linux (Figura 56), entonces lo que se debe hacer es iniciar el componente con el comando “bettercap” en la terminal de comandos.

Figura 56 : Inicio de ataque con bettercap

Fuente: Autor

Luego se le debe indicar al arp spoof (se llama spoof toda vez se entiende como un engaño a la tabla ARP) la IP de la víctima mediante el comando (Figura 57) **set arp.spoof.targets 192.168.0.20**

Figura 57 : Parametrización del ataque

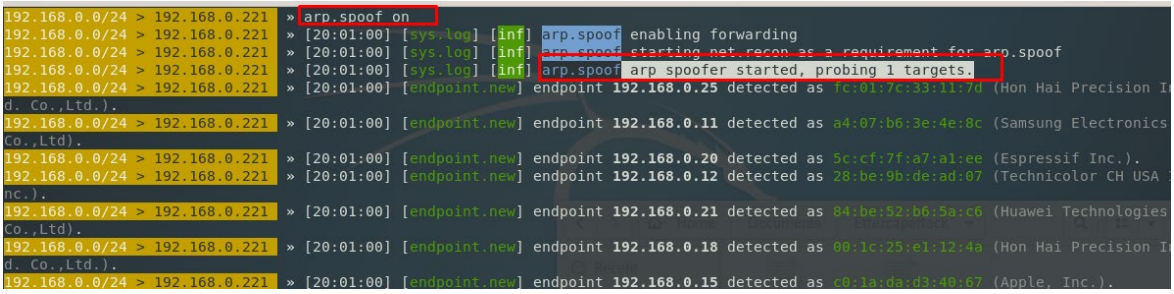


Fuente: Autor

Donde la IP 192.168.0.20 corresponde al smart switch víctima. Finalmente se lanza el ataque con el comando (

Figura 58) "**Arp.spoof on"**

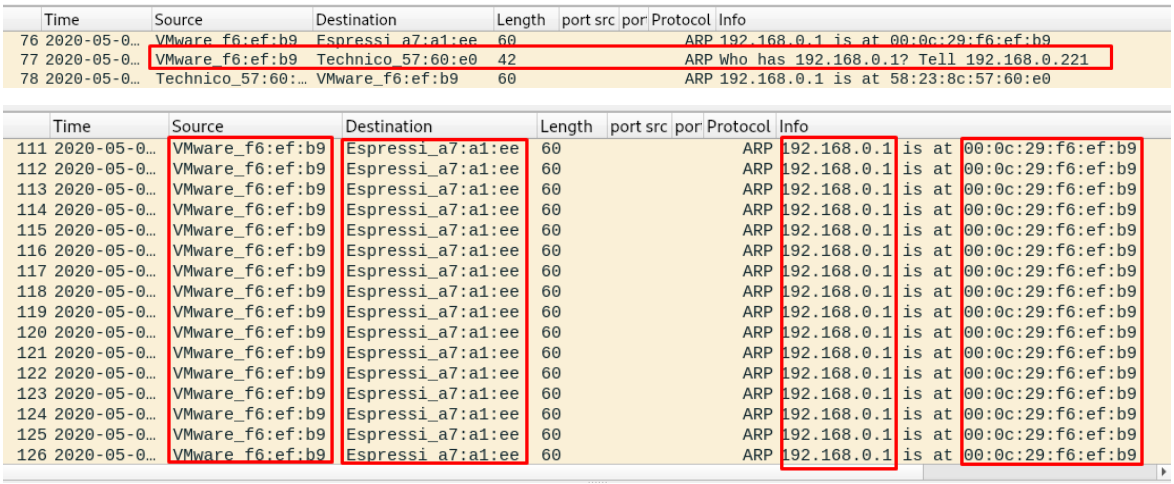
Figura 58 : Inicio de ataque con bettercap



Fuente: Autor

Una de las características de este ataque es que la herramienta fue muy fácil de configurar, lo único que se necesitó fue conocer la IP de la víctima y ya el ataque fue posible. La misma herramienta identifica los demás dispositivos en la red. Al analizar el tráfico en la red encontramos lo siguiente Figura 59

Figura 59 : Captura de tráfico con wireshark bettercap



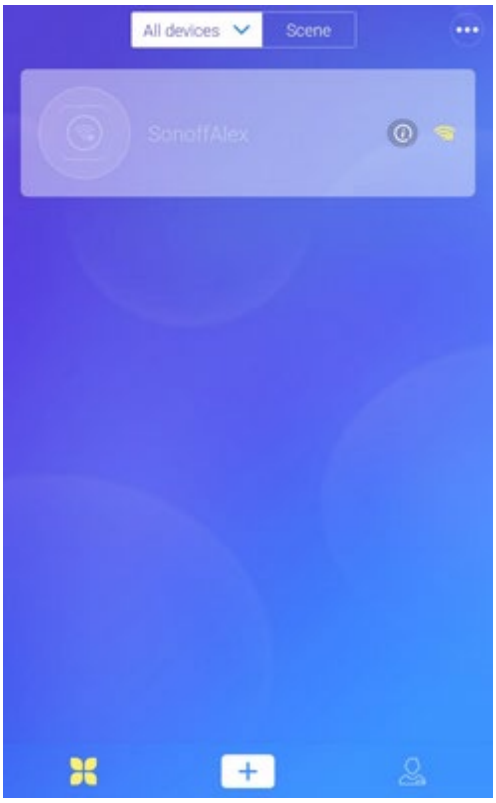
	Time	Source	Destination	Length	port src	por	Protocol	Info
76	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
77	2020-05-0…	VMware_f6:ef:b9	Technico_57:60:e0	42			ARP	Who has 192.168.0.1? Tell 192.168.0.221
78	2020-05-0…	Technico_57:60:…	VMware_f6:ef:b9	60			ARP	192.168.0.1 is at 58:23:8c:57:60:e0

	Time	Source	Destination	Length	port src	por	Protocol	Info
111	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
112	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
113	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
114	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
115	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
116	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
117	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
118	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
119	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
120	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
121	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
122	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
123	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
124	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
125	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9
126	2020-05-0…	VMware_f6:ef:b9	Espressi_a7:a1:ee	60			ARP	192.168.0.1 is at 00:0c:29:f6:ef:b9

Fuente: Autor

En la primera imagen de la Figura 59 la herramienta identifica al enrutador de la red local ubicado en la IP 192.168.0.1. Luego la máquina atacante (Kali Linux – Vmware_f6:ef:b9) le envía paquetes a la víctima (smart switch – Espressi_a7:a1:ee) con el protocolo ARP de manera continua indicándole que ahora el enrutador con la IP 192.168.0.1 cambió de dirección MAC y tiene la dirección MAC del atacante. De esta manera el smart switch no es capaz de encontrar al enrutador lo cual le causa una denegación de servicio toda vez no es capaz de salir a internet.

Figura 60 : Resultado visto en el aplicativo “eWelink”



Fuente: Autor

Así se evidencia en el aplicativo de “eWeLink” el resultado del ataque (Figura 60), el dispositivo no se encuentra conectado.

Este ataque también es muy evidente de detectar. El indicador de compromiso es la repetición de envió de paquetes con el protocolo ARP, la cantidad de paquetes enviados por segundo que no es normal y el escaneo de red que realiza la herramienta.

2.2.3 Clasificación de los indicadores de compromiso

La clasificación de los indicadores de compromiso se realiza mediante una extrapolación de la clasificación de los ataques informáticos elegidos en la fase 1, de acuerdo con la puntuación dada por el marco de Common Vulnerability Scoring System (CVSS). La puntuación nos indica que nivel

de impacto tienen los ataques elegidos y qué tan peligrosos son de tener en la red. El CVSS se utiliza para evaluar la gravedad y el nivel de amenaza de las vulnerabilidades de seguridad en los sistemas y software.

Al conocer el nivel de impacto del ataque también sirve para clasificar el nivel del indicador de compromiso, ya que el indicador de compromiso es toda la evidencia que deja la materialización del ataque informático. Si el ataque informático se clasifica como de alto impacto entonces por consiguiente su indicador de compromiso también será de alto impacto, por ejemplo, el ataque puede afectar el flujo del tráfico de la red al traer paquetes allí que antes no estaban.

Operado por el Foro de Equipos de Seguridad y Respuesta a Incidentes (FIRST), el CVSS utiliza un algoritmo para determinar tres puntajes de calificación de gravedad: Base, Temporal y Ambiental. Los puntajes son numéricos; oscilan entre 0.0 y 10.0, siendo 10.0 el más severo [45].

Figura 61 : Puntaje de CVSS

Rating	CVSS Score
None	0.0
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Fuente: [45]

De acuerdo con la calculadora dispuesta por el desarrollador FIRST Inc (Figura 61), se describen las siguientes métricas para calcular el puntaje y el impacto del ataque:

- **Vector de ataque:** Esta métrica refleja el contexto por el cual es posible la explotación de vulnerabilidades. Se dispone de las opciones: Network, Adjacent, Local, Physical.
- **Complejidad del ataque:** Esta métrica describe las condiciones más allá del control del atacante que deben existir para aprovechar la vulnerabilidad. Se dispone de las opciones: Baja o alta.

- **Privilegios requeridos**: Esta métrica describe el nivel de privilegios que debe tener un atacante antes de explotar con éxito la vulnerabilidad. Se dispone de las opciones: Ninguno, Bajo, Alto.
- **Interacción del usuario:** Esta métrica captura el requisito de que un usuario humano, que no sea el atacante, participe en el compromiso exitoso del componente vulnerable. Se dispone de las opciones: Ninguno o Requerido.
- **Alcance:** Esta métrica verifica si una vulnerabilidad en un componente vulnerable impacta los recursos en componentes más allá de su alcance de seguridad. Se dispone de las opciones: Sin alterar o Alterado.
- **Confidencialidad:** Esta métrica mide el impacto en la confidencialidad de los recursos de información administrados por un software o sistema de información debido a una vulnerabilidad explotada con éxito. Se dispone de las opciones: Alto, Bajo, Ninguno.
- **Integridad:** Esta métrica mide el impacto en la integridad de una vulnerabilidad explotada con éxito. La integridad se refiere a la confiabilidad y veracidad de la información. Se dispone de las opciones: Alto, Bajo, Ninguno.
- **Disponibilidad:** Esta métrica mide el impacto en la disponibilidad del componente afectado como resultado de una vulnerabilidad explotada con éxito**.** Se dispone de las opciones: Alto, Bajo, Ninguno.

Estas métricas son las utilizadas para calcular el nivel de amenaza de un ataque informático, la calculadora se puede encontrar en la página de First [45].

La calculadora utilizada se encuentra en la versión 3.1, a continuación, los resultados obtenidos de acuerdo con el ataque realizado.

Tabla 4 : Resultados obtenidos con la calculadora CVSS

CALCULADORA CVSS VERSIÓN 3.1										
Métricas										
Nombre del ataque	**Vector de ataque**	**Complejidad del ataque**	**Privilegios requeridos**	**Interacción del usuario**	**Alcance**	**Confidencialidad**	**Integridad**	**Disponibilidad**	**Puntuación**	**Resultado**

Deautentication attack	Network	High	None	None	Unchanged	None	None	High	5.9	Medio
Dos by flooding	Network	Low	None	None	Unchanged	None	None	High	7.5	Alto
MITM	Network	High	None	None	Changed	None	None	High	6.8	Medio
ARP Poisoning	Network	High	None	None	Changed	None	None	High	6.8	Medio

Fuente: Autor

Finalmente se obtuvieron los resultados de los ataques realizados al Smart switch (Tabla 4). Como se puede observar, el ataque que se clasifica como más peligroso y de **Alto** impacto fue la denegación de servicio por flooding. Esto se debe a que el ataque es muy fácil de realizar y no requiere de mayor complejidad para su ejecución. Los ataques de MITM y ARP Poisoning se parecen mucho y por lo tanto obtuvieron el mismo puntaje. Estos ataques requirieron de un poco más de conocimiento, instalación y procedimiento para su ejecución. Estos ataques interfieren en el alcance del ataque toda vez intervienen en dos o más sistemas conectados al mismo de su ejecución, se consideran como de impacto tipo **Medio**. Finalmente, el ataque por desautenticación fue el de menor impacto toda vez su alcance solo se limita a un mismo dispositivo, esta fue la diferencia por la cual no obtuvo el mismo puntaje que los ataques de MITM o el ARP Poisoning. Este ataque también se considera de impacto tipo **Medio.**

De esta manera, gracias a esta clasificación de los ataques realizada, se le puede dar una clasificación directamente relacionado a los indicadores de compromiso expuestos en la fase 2. Los indicadores de compromiso analizados en la investigación son registros que quedan en la red, estos registros son de gran alarma ya que pueden causar mucho tráfico indebido en la red de datos. Por consiguiente, cada indicador de compromiso expuesto queda clasificado de acuerdo con el nivel de impacto obtenido en la tabla 4 de resultados del ataque, así (Tabla 5):

Tabla 5: Clasificación indicadores de compromiso

Ataque informático	Indicador de Compromiso	Tipo de Indicador de compromiso	Clasificación
Deautentication attack	Paquetes ARP repetitivos	Trafico inusual en la red	Medio impacto
Dos by flooding	Inundación de paquetes	Trafico inusual en la red	Alto impacto
MITM	Paquetes con bandera RST repetitivos	Trafico inusual en la red	Medio impacto
ARP Poisoning	Paquetes ARP repetitivos	Trafico inusual en la red	Medio impacto

Fuente: Autor

2.3 Fase 3

La fase 3 propone diseñar e implementar la arquitectura en la que debe operar el software de monitoreo y alertas en la red IoT. Al diseñar e implementar la arquitectura de red se puede configurar el software que va a alertar sobre los ataques vistos en la red, si no se conoce los dispositivos de la red entonces queda muy difícil configurar las herramientas para realizar dicho monitoreo. Lo primero y más importante fue elegir el software de monitoreo, el cual tiene las funciones de alertamiento, seguido de allí se analiza la arquitectura de red en el hogar IoT y finalmente se crean las reglas de alertamiento en el software elegido.

A continuación, se describe el procedimiento llevado a cabo para instalar y ejecutar la arquitectura de red necesaria para monitorear y alertar los ataques provenientes al dispositivo Smart Switch.

2.3.1 Búsqueda del software para monitorear la red

En el mercado actual para soluciones de seguridad en la red existen muchas empresas que ofrecen este tipo de productos, cada una con diferentes alcances/configuraciones/precios y por lo tanto para poder hacer una elección correcta y objetiva sobre la herramienta a utilizar se realizó un estudio para comparar las diferentes soluciones.

En la investigación titulada “Modelo para la gestión de incidentes de seguridad en redes industriales SCADA a través del algoritmo de predicción Filtro Kalman” [46] los autores evaluaron 10 sistemas de detección de intrusos disponibles en el mercado y los compararon entre si, con el objetivo de hacer la mejor elección para la realización y montaje de un sistema de detección en una red SCADA, SCADA es acrónimo para sistemas de Supervisión, Control y Adquisición de Datos.

Si bien en la investigación presente no se considera un escenario de tipo SCADA, si se considera un análisis de tráfico en la red, y al igual que un escenario con un sistema SCADA, el Smart switch y el SCADA necesitan de una protección en la red. Por lo tanto, se pueden utilizar los mismos criterios de elección del sistema SCADA para la investigación presente del Smart Switch.
Los sistemas de detección de intrusos en la red analizados por los investigadores fueron: SNORT, Suricata, Kismet, Bro IDS, Ossec, Tripwire, Samhain, Fortinet, Radware y Palo Alto. Con unas características establecidas por ellos, se definió el mejor NIDS para la investigación y este fue el SNORT.

Con más de 5 millones de descargas y más de 600,000 usuarios registrados, Snort es el sistema de prevención de intrusos más utilizado a nivel mundial [47]. Por su popularidad esto hace que se encuentre muchísimo material de apoyo en internet, en diferentes lenguajes y distintas plataformas que pueden ser de gran ayuda para la investigación. En la investigación [48] los investigadores concluyeron que snort consumía menor cantidad de recursos de procesamiento comparado con Suricata y también concluyeron que Snort detectaba menos falsos positivos que Suricata al analizar un mismo archivo de paquetes, por lo tanto afirmaron que Snort tiene una efectividad de detección 10% superior a la de Suricata.

2.3.2 Tabla comparativa con el software de monitoreo en la red

Los investigadores propusieron unas características a evaluar con el fin de determinar cuál sistema de detección de intrusos sería el ideal para lograr una buena seguridad en la red SCADA [46].

Tabla 6: Características sistemas de IDS

No	Características del IDS	Puntos	Consideración
1	Virtualización	9	Dada la facilidad de las plataformas o máquinas virtuales, el IDS debe poder ser instalado en este tipo de sistemas, con ello, se podría tener movilidad a la hora de fijar un IDS en otra red.
2	Alertas en tiempo real	9	Teniendo en cuenta que es un filtro predictivo, las alertas deben darse en tiempo real para poder actuar ante cualquier posible evento de seguridad.
3	Personalización de reglas	9	Los sistemas SCADA como elementos tecnológicos particulares, requieres de soluciones que se ajusten a sus necesidades, esto es, que se puedan personalizar reglas acordes al sistema y el proceso organizacional.
4	Modo NIDS	9	Parte fundamental de la detección es que se logre identificar desde la red cualquier posible anomalía, por ello debe tener funciones de NIDS (Network -IDS)
5	Solo IDS	10	La disponibilidad de la plataforma es lo primordial, por lo cual, los IDS debe ser configurados SOLO en modo lectura o monitoreo, no se puede permitir que queden en modo IPS (de forma nativa). Por esta razón, este factor (a diferencia de los demás) tiene un aumento en la puntuación. Un sistema IPS debe cursar el tráfico por el dispositivo, aumentando los falsos positivos hasta que el sistema se estabilice, pero esto puede tomar más tiempo de lo normal.

6	Open Source	9	Los presupuestos del proyecto son reducidos, por lo cual la necesidad es que el IDS pueda solventar dicha necesidad.
7	Multiprocesador	9	Muy alineado con la característica 1 y 9.
8	Protocolo Modbus	9	Los protocolos de comunicación de tipo Modbus son los usados para los sistemas PLC, por lo cual, es necesario que el IDS soporte dicho protocolo.
9	Multiplataforma	9	Es importante, en consideración con el ítem 1, que se permita la instalación del IDS acorde a los recursos y plataformas disponibles en la organización, para con ello dar más escalabilidad.
10	Automatización	9	Opción de poder automatizar tareas en la detección.
11	Experiencia de los investigadores	9	La experiencia en IDS para SCADA de los proveedores, comunidades o grupos de interés en fundamental para tener sistemas estables y actualizados.
	Total	**100**	

Fuente: Modelo para la gestión de incidentes de seguridad en redes industriales SCADA a través del algoritmo de predicción Filtro Kalman [46]

Según el sistema de ponderación realizada por los investigadores, se crearon 11 diferentes categorías a evaluar, cada categoría con un puntaje diferente entre 9 y 10 (Tabla 6).Teniendo en cuenta estos criterios de cumplimiento se sigue a analizar la tabla final donde se elige el sistema IDS para realizar la investigación

Tabla 7 : Tabla comparativa IDS vs Características

No.	Caracteristicas	Peso	SNORT	Suricata	Kismet	Bro IDS	Ossec	Tripwire	Samhain	Fortinet	Radware	Palo Alto
1	Virtualización	9	x	x	x	x	x	x	x			
2	Alertas en tiempo real	9	x	x		x	x	x		x	x	x

3	Personalización de reglas	9	x	x		x	x					
4	Modo NIDS	9	x	x			x		x	x	x	x
5	Solo IDS	10	x		x	x				x		
6	Open Source	9	x	x		x	x		x			
7	Multiprocesador	9		x	x							x
8	Protocolo Modbus	9	x	x			x	x		x		x
9	Multiplataforma	9	x	x	x	x	x	x	x	x	x	x
10	Automatización	9	x	x	x	x	x	x		x	x	x
11	Experiencia de los investigadores	9	x									
Totales		100	91	81	46	64	72	45	36	55	36	54

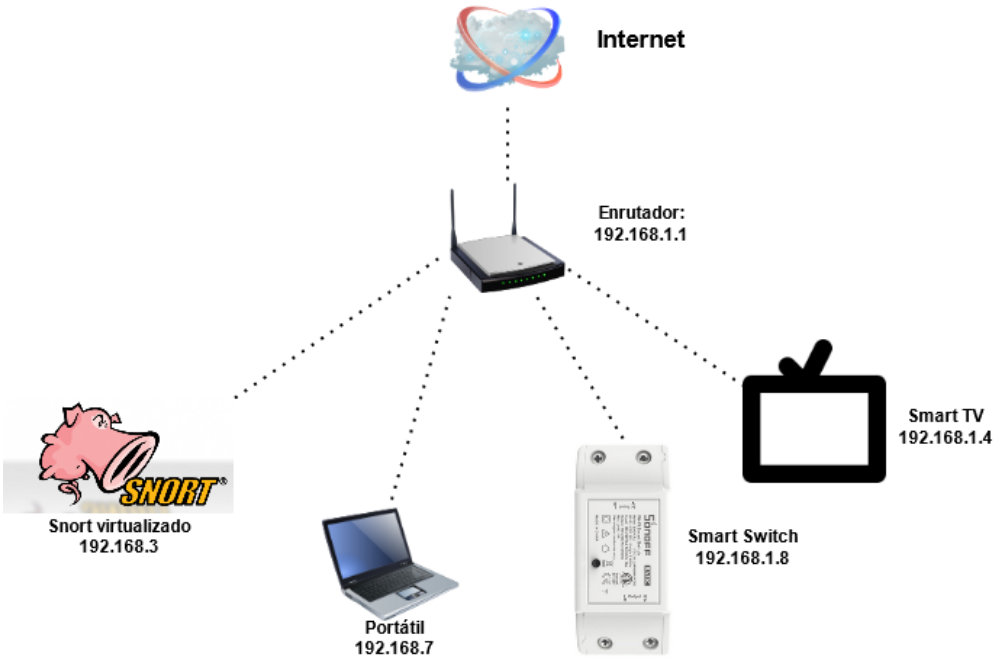
Fuente: Modelo para la gestión de incidentes de seguridad en redes industriales SCADA a través del algoritmo de predicción Filtro Kalman [46]

En la Tabla 7 se hace la comparación de unas características establecidas que deben cumplir los sistemas versus los diferentes sistemas de intrusos investigados. Se marca con una “x” si el sistema cumple con dicha característica, adicional a esto a cada característica se le agregó un valor “peso” para así poder sumar y determinar cuál es el sistema con mejor puntuación. A la final el sistema elegido y que mejores características obtuvo fue el SNORT, sistema el cual se decidió implementar en la investigación presente.

2.3.3 Instalación/configuración software de monitoreo en la red

Para la instalación e implementación del sistema Snort se debe utilizar la arquitectura común y corriente de un hogar, es decir, la arquitectura de red que tendría un hogar sin intervención de un experto, solamente como se instala la red al pagar por el servicio de internet a un proveedor. La arquitectura común y corriente es la siguiente y es con la cual se trabajó (Figura 62):

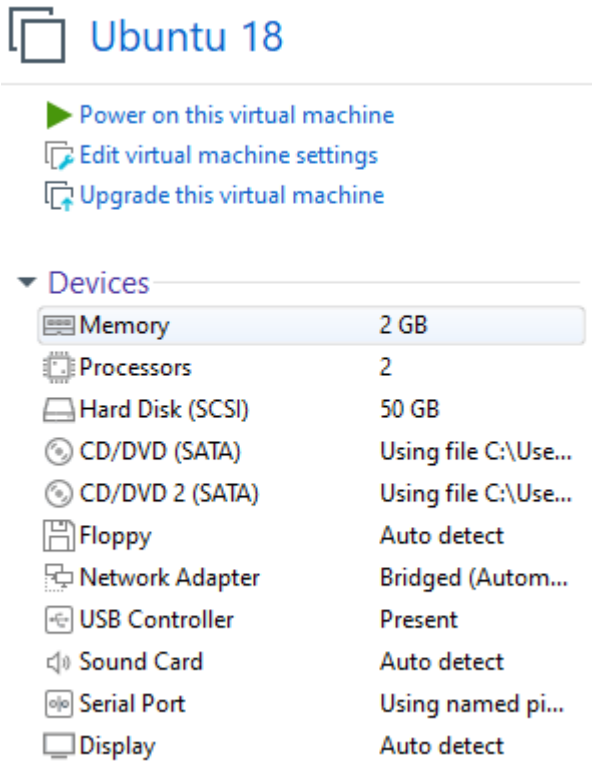
Figura 62 : Arquitectura de red Snort



Fuente: Autor

El sistema elegido para instalar el sistema de Snort fue un ambiente virtualizado con el sistema operativo de Ubuntu en su versión 18.04. La máquina configurada cuenta con los siguientes recursos (Figura 63):

Figura 63 : Recursos en máquina virtual



Fuente: Autor

La máquina virtual cuenta con 2 GB de memoria ram, 2 procesadores virtuales y un disco duro de 50 GB, Snort por lo general no requiere de gran capacidad de cómputo para operar.

La instalación de snort es bastante simple de realizar toda vez con la configuración básica de snort se pueden configurar las reglas requeridas en la investigación. Primero es el comando de instalación con el "**apt-get**" (Figura 64):

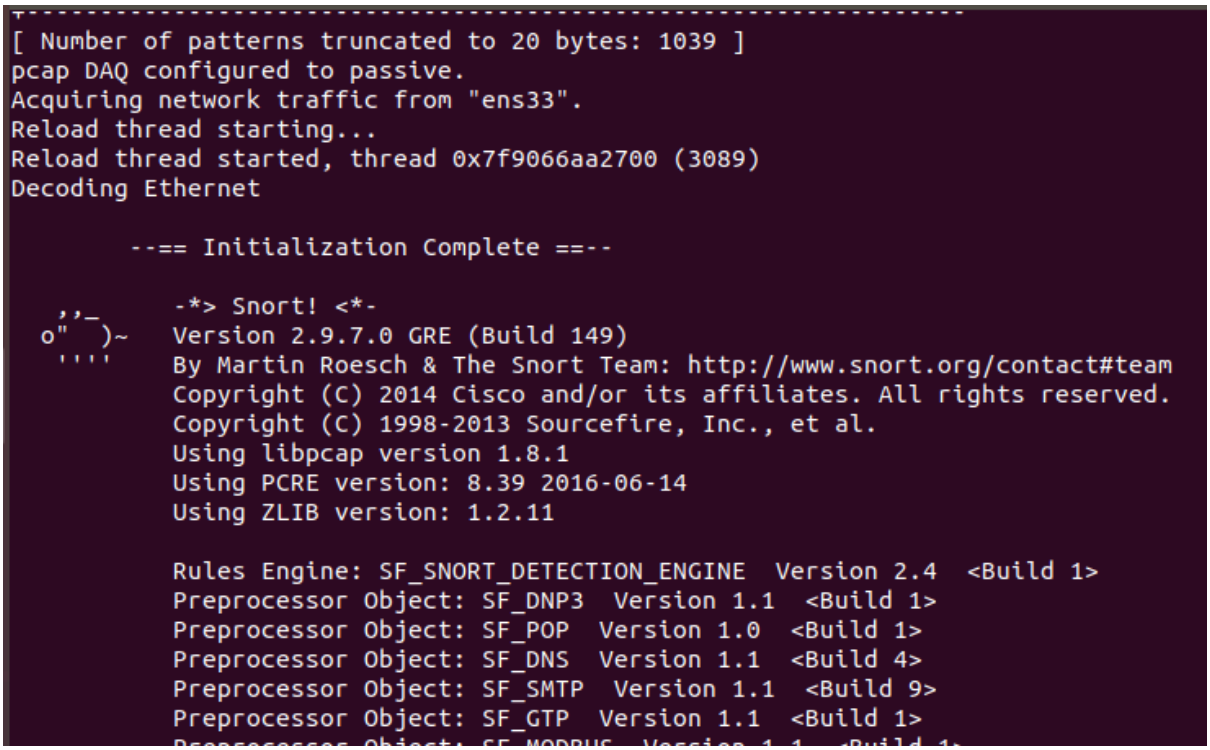
Figura 64 : Instalación Snort apt-get

```
root@alex-virtual-machine:/home/alex# sudo apt-get install snort*
```

Fuente: Autor

Al finalizar la instalación nos debe aparecer un mensaje así Figura 65

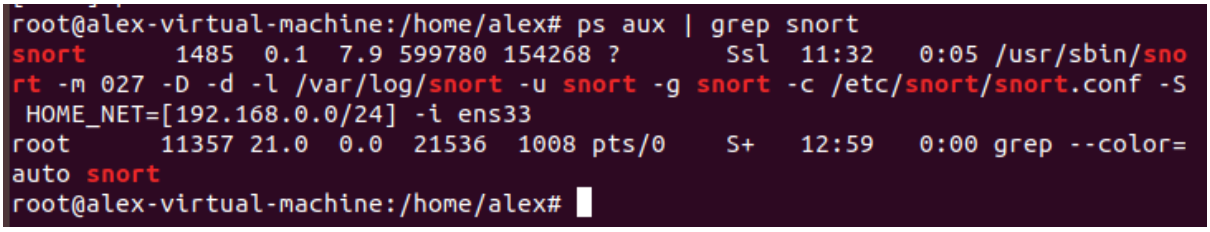
Figura 65 : Finalización de instalación



Fuente: Autor

Como se puede observar, se instaló y se trabajó con la versión 2.9.7.0 de Snort. Una vez instalado se revisa el servicio en ejecución con el comando **ps aux | grep snort** (Figura 66)

Figura 66 : ps aux | grep snort



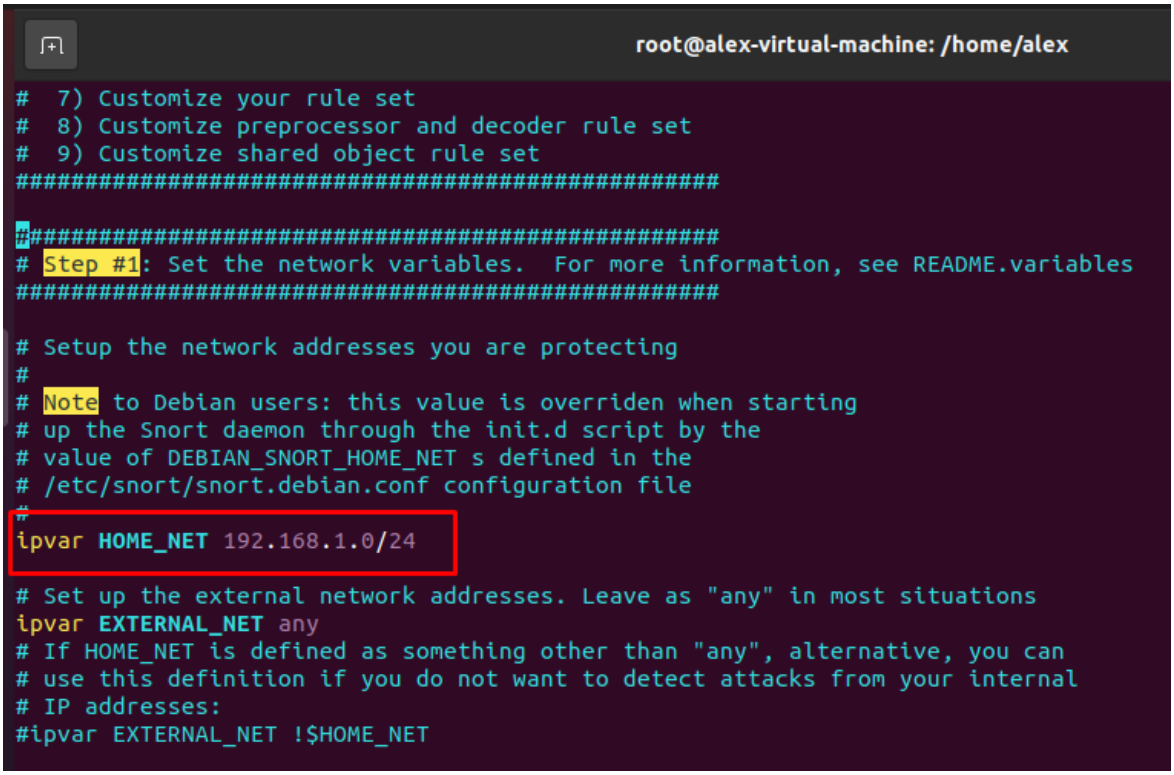
Fuente: Autor

Con este comando se puede notar el servicio en ejecución, se puede ver el usuario Snort es quien está corriendo dicho proceso. Para iniciar o detener el servicio se utilizan los comandos **sudo /etc/init.d/snort stop** y **sudo /etc/init.d/snort start**.

A continuación, se debe ingresar al archivo de configuración de Snort y cambiar la variable de red, el archivo se puede encontrar en la ruta **/etc/snort/snort.conf.**

Se edita el archivo y le indicamos la subred en la cual Snort va a trabajar en su variable $HOME_NET, para este caso es la 192.168.1.0/24 (Figura 67)

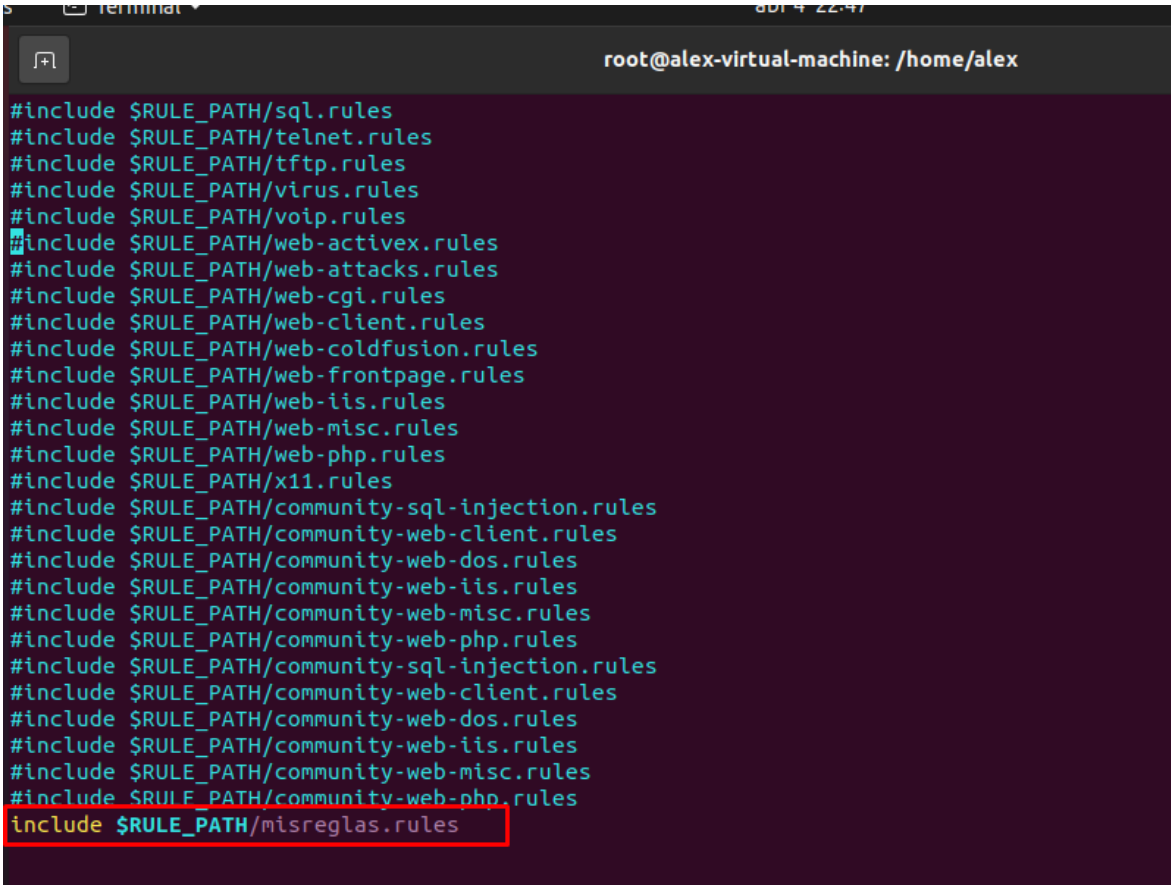
Figura 67 : Configuración archivo snort



Fuente: Autor

Para una configuración básica lo siguiente es cargar las reglas que Snort utiliza para la detección de intrusos, para este ejercicio se generó unas reglas personalizadas. Estas reglas se habilitan y se llaman en el archivo de configuración **/etc/snort/snort.conf** (Figura 68).

Figura 68 : Reglas permitidas

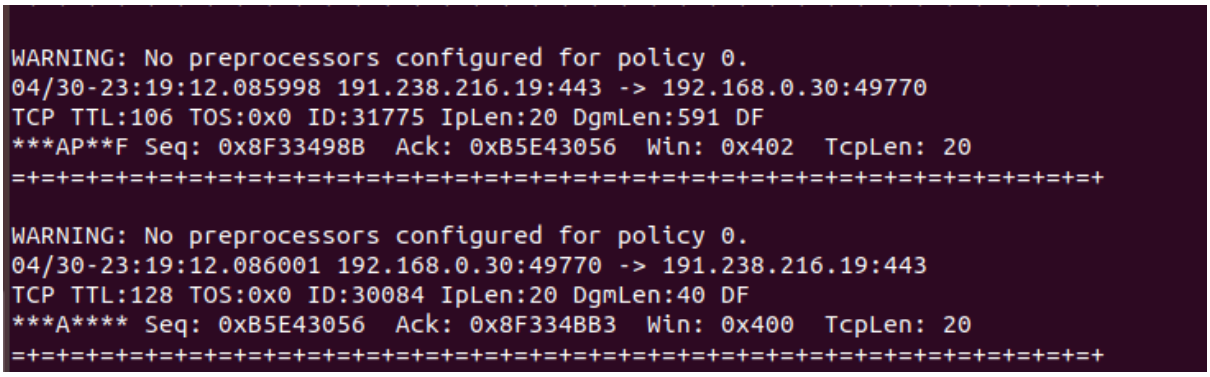


Fuente: Autor

Las reglas que se vayan a utilizar se les quita el símbolo # para que funcione o si no el sistema pensará que es un comentario. La única regla habilitada es la regla personal llamada “misreglas.rules”, más adelante se comentará sobre su contenido, esta regla se ingresa manualmente en el archivo **snort.conf** de configuración.

Lo siguiente es probar los distintos modos del snort, el primer modo es en su estado Sniffer (Figura 69). Para iniciar en Sniffer mode se utiliza el comando **sudo snort -v**, con el parámetro -v este solo muestra la información de la cabecera del paquete (IP /TCP/UDP/ICMP). El tráfico se ve así:

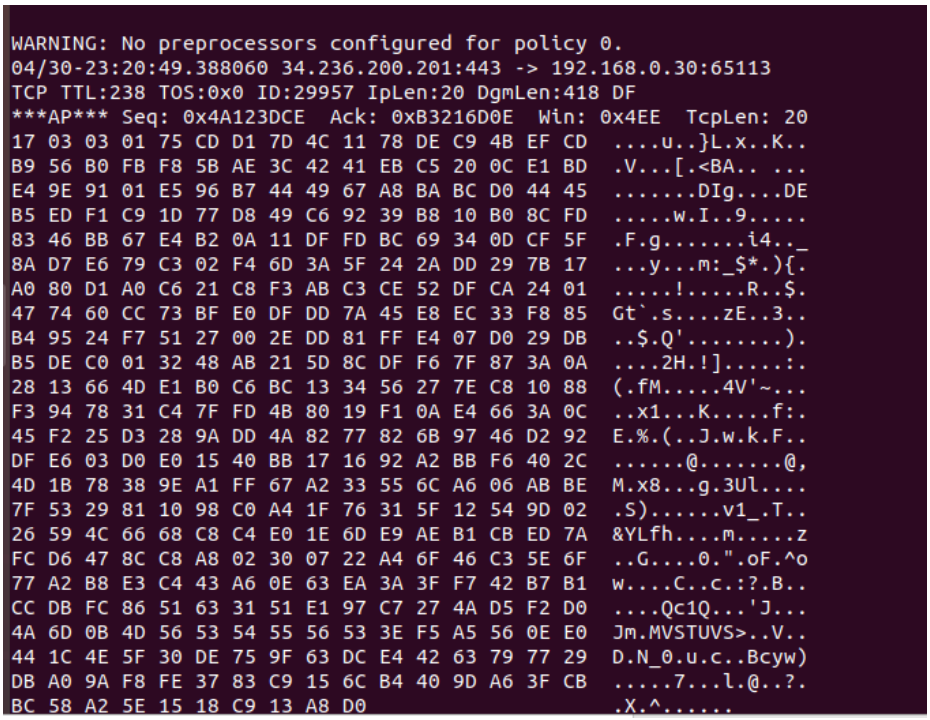
Figura 69 **:** Snort modo Sniffer



Fuente: Autor

Lo siguiente es probar Snort en Sniffer mode pero con lectura adicional de la capa de datos del paquete, mediante el comando **sudo snort -vd (**Figura 70)

Figura 70 : Snort modo Sniffer con lectura de capa de datos



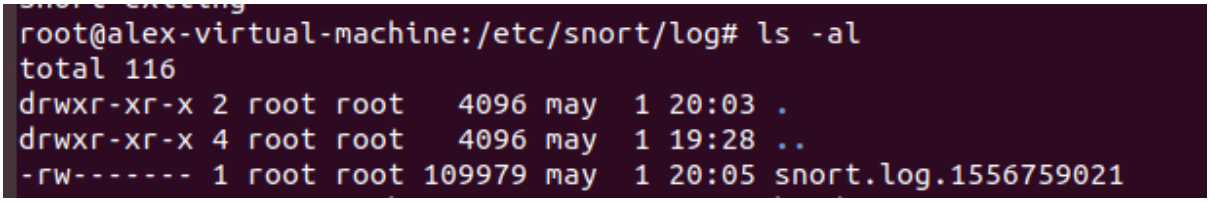
Fuente: Autor

Se aprecia esta vez la lectura de los datos que contiene el paquete en específico. El siguiente modo es en Log mode. Este modo es útil para exportar el tráfico de la red y guardarlo en archivos para analizarlos posteriormente en otras herramientas, como por ejemplo en Wireshark o TCPdump.

Con el comando **snort -de -l /etc/snort/log**, le indicamos a snort que arranque en modo Log, el parámetro -l establece el modo y se le especifica la ruta donde guardará el archivo. El parámetro -de indica que va a guardar los datos de la capa de aplicación y los datos de capa dos.

Al revisar la ruta de guardado se encuentra lo siguiente (Figura 71) :

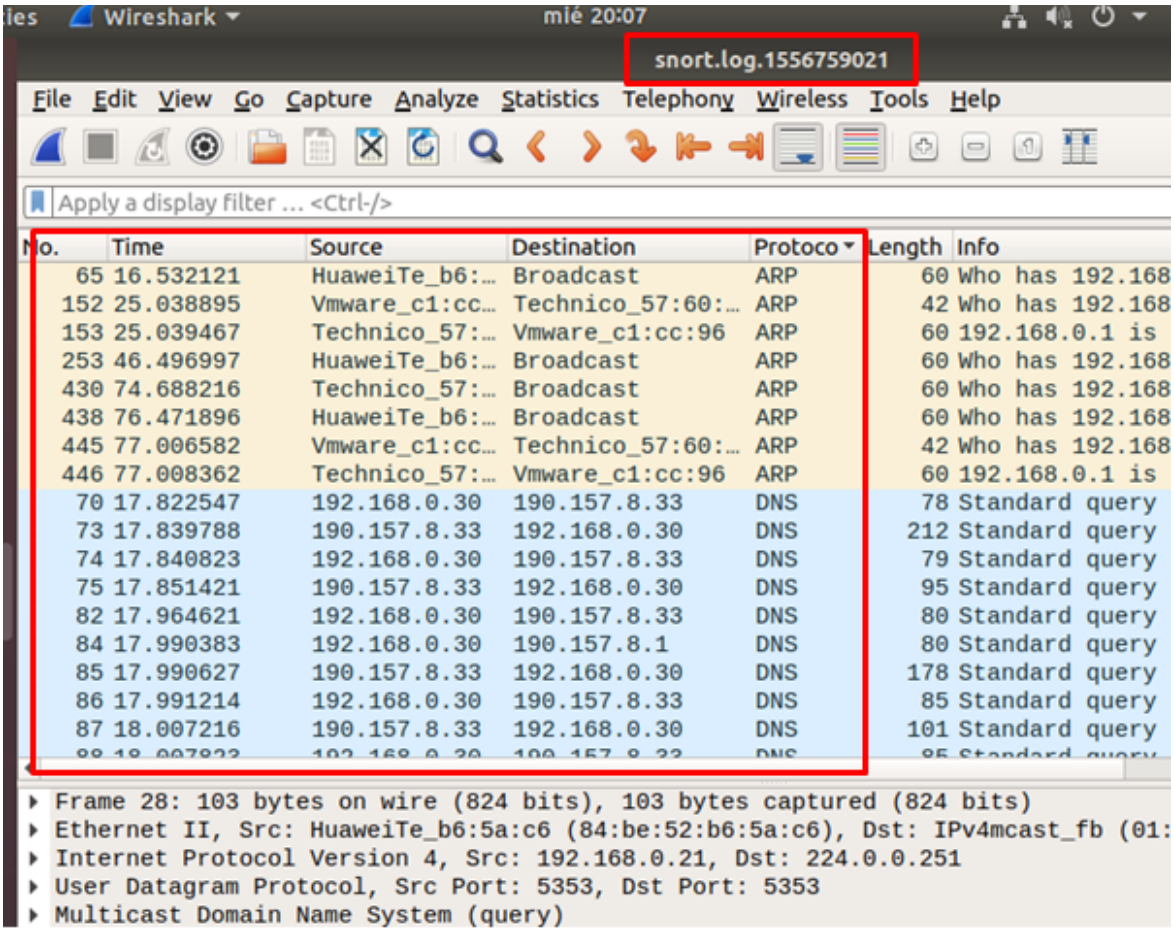
Figura 71 : Log recogido por Snort



Fuente: Autor

Se encuentra un archivo de tipo log, este archivo “snort.log.1556759021” es compatible con las herramientas de wireshark y Tcpdump como se había dicho anteriormente. Al abrirlo en wireshark se puede observar el tráfico capturado por Snort (Figura 72):

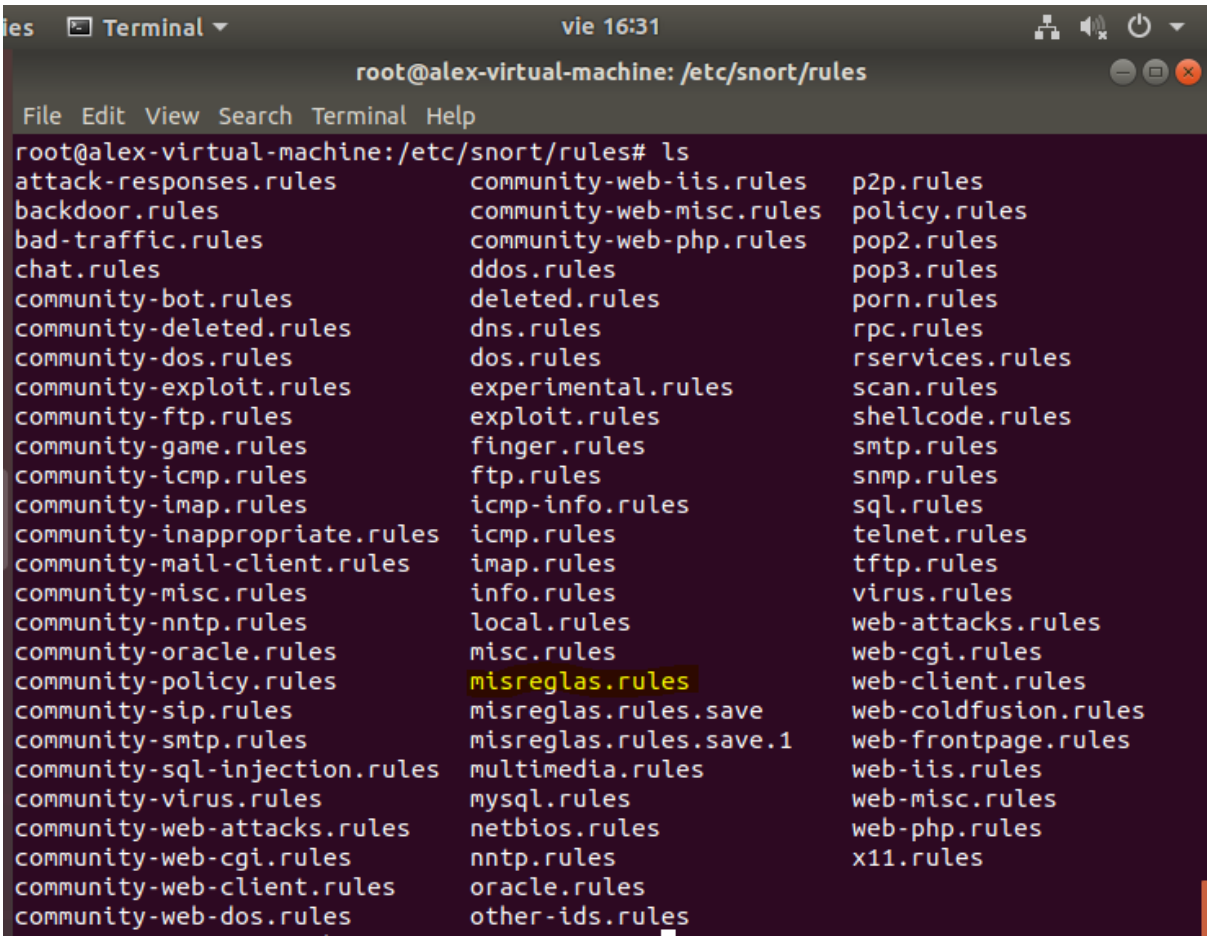
Figura 72 : Análisis de log de snort en Wireshark



Fuente: Autor

Por último, el modo de detección de intrusos, para ejecutar este modo primero debemos tener listas las reglas. Para este laboratorio se configuraron 3 reglas las cuales se explican más adelante. Las reglas personales se deben guardar en la ruta **/etc/snort/rules (**Figura 73**).**

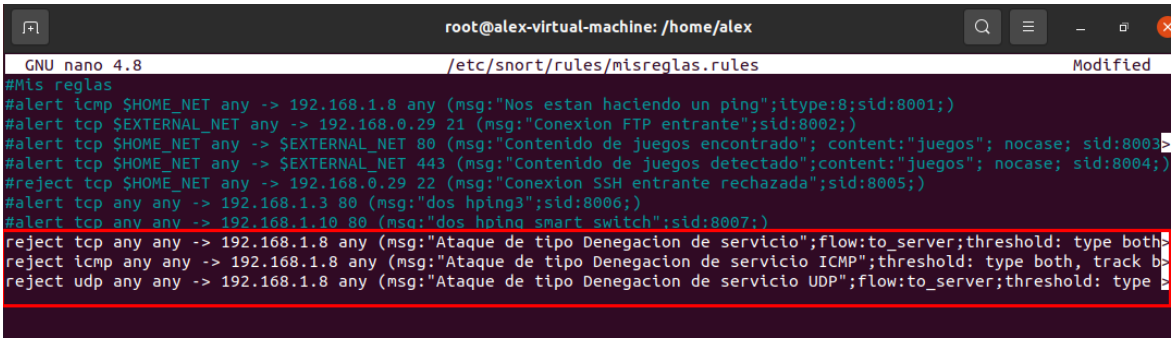
Figura 73 : Directorio de reglas snort



Fuente: Autor

Dentro de esta ruta se encuentra el archivo con las reglas personalizadas que se crearon, el archivo es denominado “misreglas.rules”. Las reglas establecidas en este archivo nos sirven para detectar los diferentes ataques de denegación de servicio definidos previamente (Figura 74).

Figura 74 : Regla personalizada Snort



Fuente: Autor

Para poder detectar los ataques de tipo ARP se configuró el preprocesador dentro de snort para que haga dicho trabajo. En el manual de configuración de snort nos muestra la estructura básica para implementar y habilitar dicho preprocesador [49].

Figura 75 : Estructura configuración preprocesador snort

The third example configuration has unicast detection enabled.

```
preprocessor arpspoof: -unicast
preprocessor arpspoof_detect_host: 192.168.40.1 f0:0f:00:f0:0f:00
preprocessor arpspoof_detect_host: 192.168.40.2 f0:0f:00:f0:0f:01
```

Fuente: Documentación Snort [49]

Donde se debe especificar la IP y la dirección MAC del dispositivo que queremos vigilar (Figura 75). Esta configuración se debe hacer en el archivo de **/snort.conf,** el cual es el archivo raíz de la configuración de snort. Se configura con el parámetro de -unicast toda vez los paquetes de ARP son de tipo unicast o en otras palabras viajan hacia un destino. Así se configura para la investigación presente

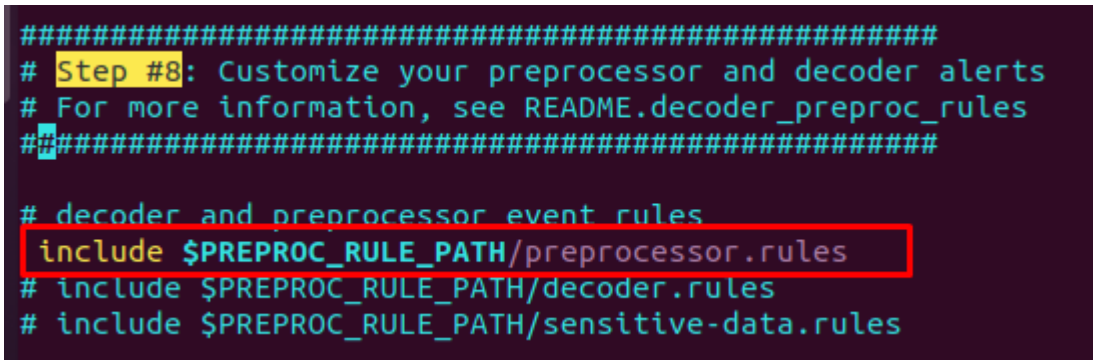
Figura 76 : Estructura configuración preprocesador snort

```
# ARP spoof detection.  For more information, see the Snort Manual - Configuring Snort - Preprocessors - ARP Spoof Prep
rocessor
 preprocessor arpspoof: -unicast
# preprocessor arpspoof
 preprocessor arpspoof_detect_host: 192.168.1.1 50:0F:F5:A5:AC:68
 preprocessor arpspoof_detect_host: 192.168.1.8 5C:CF:7F:A7:A1:EE
```

Fuente: **Autor**

Donde la IP 192.168.1.1 corresponde al enrutador y el 192.168.1.8 corresponde al Smart switch (Figura 76). Se debe vigilar igual al enrutador toda vez también es víctima de los ataques a su tabla ARP. El siguiente paso es habilitar las alertas del preprocesador, para habilitarlas se debe ir al mismo archivo **/snort.conf** y habilitar la siguiente línea de código (Figura 77):

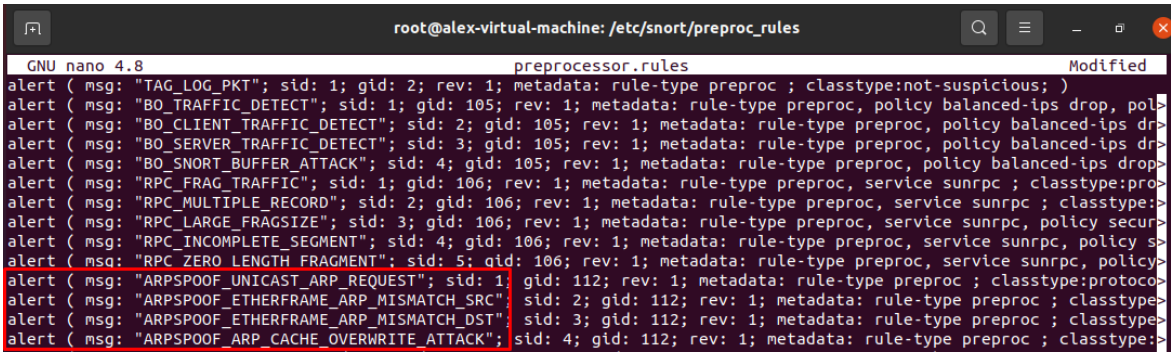
Figura 77 : Configuración alertas preprocesador



Fuente: **Autor**

Al instalar snort este no quedó con el directorio de "preprocessor.rules" entonces haciendo una búsqueda se pudo obtener de un proyecto en Github, https://github.com/threatstream/snort. Se debió copiar el directorio de "prepocessor_rules" en el directorio de snort para que funcionara las alertas del preprocesador. Al configurar el preprocesador se habilitan las reglas correspondientes, dentro del archivo de "preprocessor.rules" se encuentran las alertas que se activan ante los ataques al protocolo ARP (Figura 78):

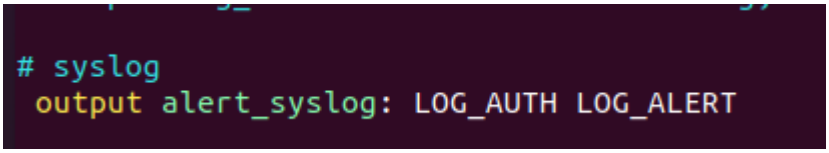
Figura 78 : Reglas del preprocesador ARP



Fuente: **Autor**

Por último, es muy importante habilitar la línea de código que permite que Snort guarde sus alertas en el archivo de "auth.log" de nuestro sistema operativo (Figura 79). En el mismo archivo de **snort.conf**

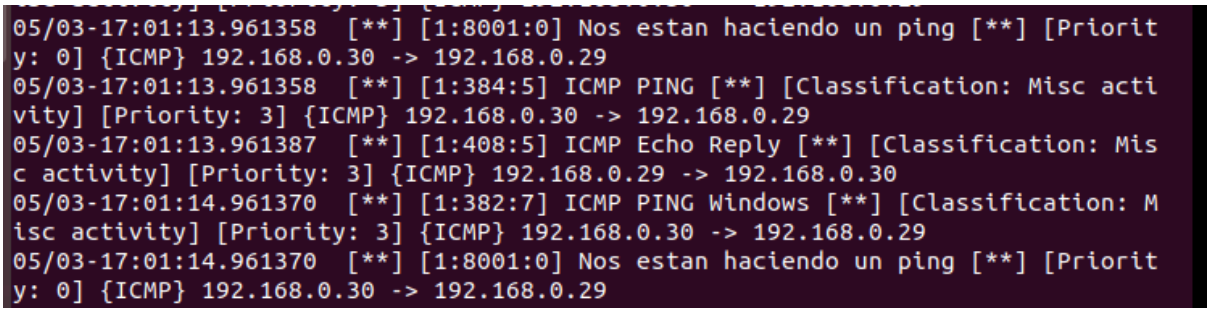
Figura 79 : Habilitar Logs del sistema



Fuente: **Autor**

Ahora para ejecutar snort en modo detección y que utilice las reglas configuradas utilizamos el comando **snort -d -h 192.168.1.0/24 -A console -c /etc/snort/snort.conf -i ens33 -s**, donde -A console indica el modo alerta por consola, -i indica la interfaz de escucha, -c indica la ruta del archivo configuración con las reglas establecidas, -s indica que envie las alertas al syslog y -d indica que revise la capa de aplicación. Cuando Snort detecta una alerta la muestra de la siguiente manera (Figura 80).

Figura 80 : Alertas mostradas por snort



Fuente: **Autor**

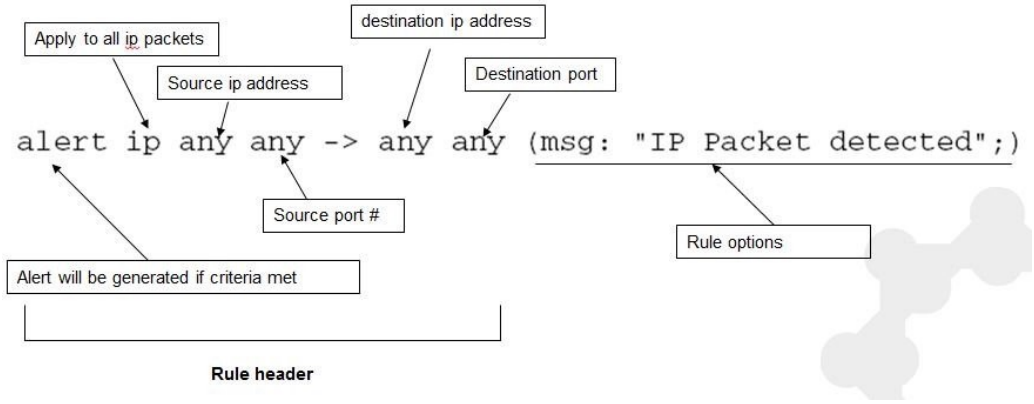
Es muy importante revisar que las alertas configuradas si estén quedando guardadas en el archivo de "Auth.log", con el siguiente comando se puede revisar las últimas líneas registradas en el "**tail -f /var/log/auth.log".** Si por algún motivo no se evidencian cambios en el archivo de log, se debe verificar bien que el comando de snort este correctamente digitado como se mostró anteriormente y que su archivo de configuración tenga los parámetros mostrados previamente. A veces puede

que sea necesario reiniciar el servicio de syslog, toda vez este servicio es quien escribe los logs en Ubuntu, con el comando **"sudo service rsyslog restart".**

2.3.4 Creación de reglas basadas en los indicadores de compromiso hallados

La estructura básica de una regla se explica de la siguiente manera:

Figura 81 : Alertas mostradas por snort



Fuente: What is Snort? [50]

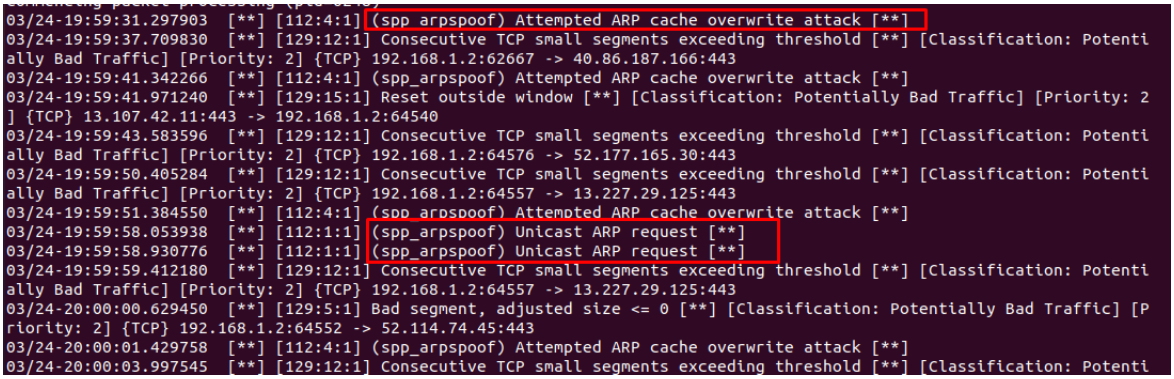
La regla se divide en dos partes, en la cabecera y en las opciones de la regla (

Figura **81**). En la cabecera se agrega la acción (alert, block, reject), seguido del protocolo (tcp, udp, icmp, etc) luego la IP Origen, puerto origen, -> indica la dirección de fuente a destino, IP destino, puerto destino. Las opciones de la regla pueden variar, las más comunes son: mensajes, id único para identificar el numero de la regla, parámetros, revisión de la versión de la regla para un mantenimiento y el threshold el cual es el más útil que se detalla más adelante.

Regla para el ataque de Des-autenticación

Cuando se realizó el ataque de Des-autenticación con el software de “kickthemout” se evidenció el manejo de los paquetes ARP en el indicador de compromiso, también se observó que había duplicidad en algunas IP en los registros de ARP, con este indicador se concluye que se debe vigilar los paquetes ARP en la red. Una vez configurada el preprocesador para vigilar los paquetes ARP en el snort, se ejecuta el ataque de “kickthemout” sobre la víctima, el Sonoff y se obtienen las siguientes alertas

Figura 82 : Alertas ARP en snort



Fuente: **Autor**

El preprocesador alerta el unicast repetitivo con el protocolo de ARP a las IP configuradas previamente y alerta sobre una manipulación en las tablas de ARP de los dispositivos (Figura 82), con esta alerta configurada ya se pudo detectar el ataque de Des autenticación al dispositivo Smart switch.

Reglas para los ataques de denegación de servicio

Como ya se analizó en el objetivo anterior, los ataques por denegación por servicio son muy fáciles de ejecutar, es importante crear una regla con buena parametrización para evitar falsos positivos en la red. Con la utilización del parámetro “reject” el sistema de detección de intrusos cancela las peticiones de conexiones entrantes, lo cual ayuda a mitigar el ataque de DoS.

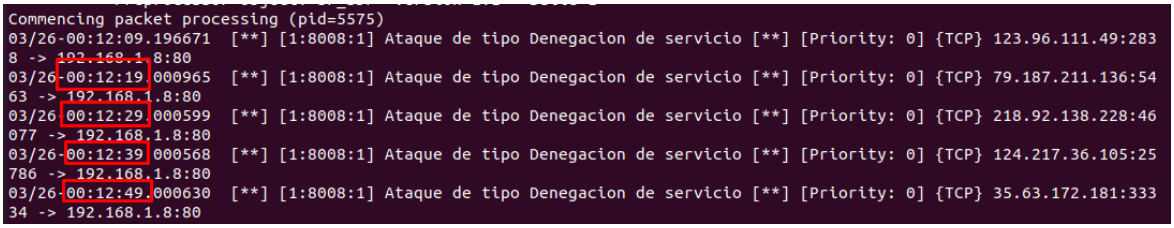
Mediante el parámetro “treshold” [51] se le configura a snort la capacidad para alertar cada ***x*** cantidad de segundos y también según la cantidad de alertas que llegan en un rango ***y*** de tiempo.

Por ejemplo, se le puede configurar que muestre la alerta de denegación de servicio si la victima recibe 30 paquetes TCP en 10 segundos. Con este parámetro se evitan muchos falsos positivos y se evidencian mejor los ataques recibidos.. Se crea la siguiente regla: **reject tcp any any -> 192.168.1.8 any (msg:"Ataque de tipo Denegacion de servicio";flow:to_server;threshold: type both, track by_dst, count 30, seconds 10 ;sid:8008;rev:1; Priority:1;)**

Se le configura la opción “both” para que tenga en cuenta la cantidad de alertas en 10 segundos y que cuente cada 30 paquetes detectados. La opción “by_dst” es para que se fije en el destino, en este caso el Smart switch y no le importe el origen del paquete, “count” es la cantidad de paquetes a detectar y “seconds” es la frecuencia con la que va a mostrar la alerta. El parametro “sid” establece el identificador único que se le pone a las reglas, “rev” hace referencia a la versión de la regla para posteriores optimizaciones y “Priority” establece la prioridad de la regla, con este parámetro se buscaran las alertas para enviar los correos.

El parámetro “Flow” indica la dirección en la que se va a analizar los paquetes para la regla estipulada [52]. En este caso el Flow es hacia el servidor toda vez para este ejercicio el Smart switch actúa como el servidor y el atacante es el cliente, este parámetro es útil para eliminar falsos positivos de la red.

Figura 83 : Alerta ataque de denegación de servicio



Fuente: **Autor**

En la práctica se logra observar que la alerta se dispara cada 10 segundos como fue configurada (Figura 83), también se aprecia la IP de la víctima y la IP del atacante, en este caso mediante el ataque de HPING3.

Ante el ataque por NPING con la misma regla se obtienen los mismos resultados con la regla configurada. Esto es debido al threshold configurado toda vez este es el parámetro que nos indica que el dispositivo está recibiendo demasiados paquetes en una cantidad corta de tiempo. El ataque por NPING fue realizado con el comando : **"nping --tcp-connect --rate=90000 -c 9000000 -q 192.168.1.8",** en el segundo objetivo se explican los parámetros utilizados.

El último ataque establecido en la investigación de tipo flood fue el realizado con el comando de "ping". Este ataque nos reveló algo importante y es el de tener en cuenta el protocolo del paquete. El ataque con ping se ejecuta con el comando "**ping -f -s 56500 192.168.1.100**", pero como era de esperarse este ataque utiliza paquetes de tipo ICMP y no de tipo TCP como se venía trabajando con los ataques anteriores. Al utilizar la regla anterior para detectar la denegación de servicio se observó que no funcionaba, esto debido a que el ataque con "ping" utiliza otro tipo de protocolo, el ICMP. Por lo tanto se procedió a crear una regla para la identificación de ataques de denegación de servicio con los paquetes de tipo ICMP, la regla creada fue la siguiente: **reject icmp any any -> 192.168.1.8 any (msg:"Ataque de tipo Denegacion de servicio ICMP";threshold: type both, track by_dst, count 30, seconds 10 ;sid:8009;rev:1; Priority:1;)**

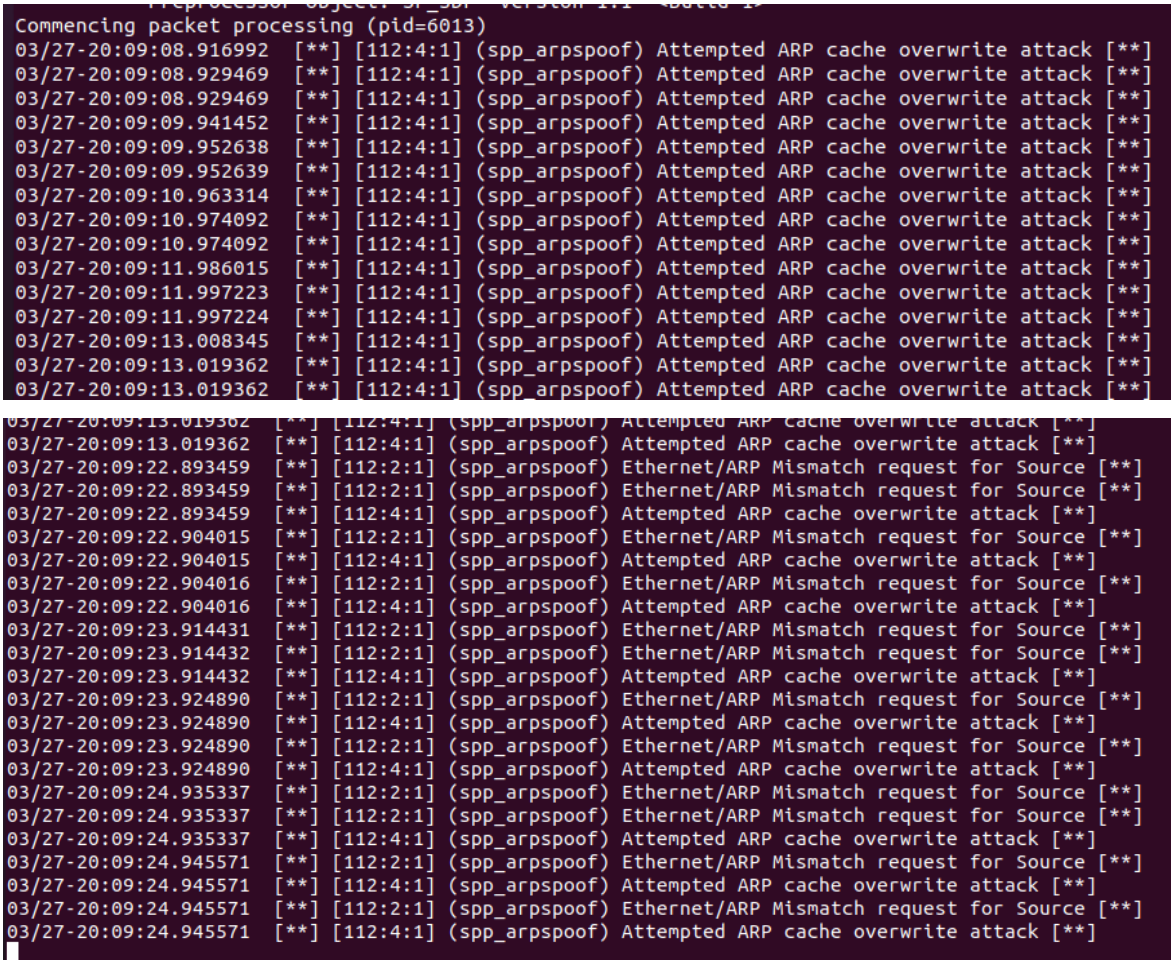
Donde se cambia el protocolo de TCP a ICMP, otro cambio a notar es que no se utiliza el parámetro de "Flow" toda vez snort no es capaz de identificar la dirección del ataque con los paquetes ICMP. Ahora se crea una regla para los ataques UDP, toda vez un atacante también podría utilizar este protocolo en sus vulneraciones, la regla creada es igual a la de TCP, solo se cambia el protocolo**: reject udp any any -> 192.168.1.8 any (msg:"Ataque de tipo Denegación de servicio";flow:to_server;threshold: type both, track by_dst, count 30, seconds 10 ;sid:8008;rev:1;)**

Reglas para los ataques de hombre en el medio

Básicamente el patrón común con los ataques del hombre en el medio es que estos necesitan manipular las tablas ARP de los dispositivos a los cuales se les quiere hacer el daño. Como ya previamente se había configurado el preprocesador de snort, este ataque con el software de

ETTERCAP se hace de fácil detección toda vez el snort está monitoreando en tiempo real la IP y la dirección MAC del dispositivo Smart switch. Cuando se ejecuta el ataque de ETTERCAP las alertas que se observan son las siguientes:

Figura 84 : Alerta de ataque ARP a Smart switch



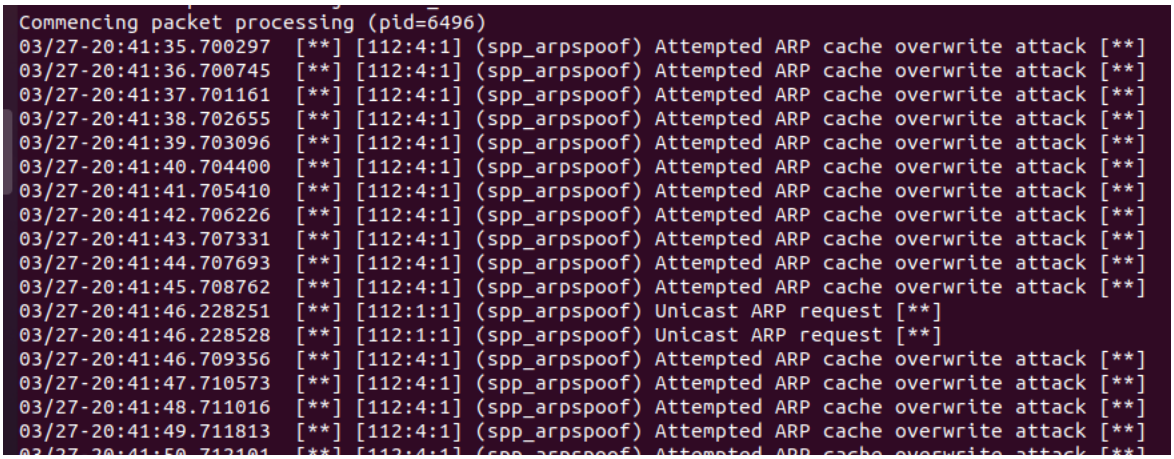
Fuente: **Autor**

En esta ocasión vemos una alerta diferente al primer ataque analizado (Figura 84), el mensaje de "Ethernet/ARP Mismatch request for Source" hace referencia a qué el paquete ARP es distinto a la información que se tenía en la fuente. Con lo anterior queda funcionando el preprocesador también para detectar ataques de hombre en el medio.

Reglas para los ataques ARP poisoning

El ataque establecido para esta práctica fue realiza con el software de “bettercap”. Este ataque fue bastante fácil de ejecutar y sabiendo que también manipula la tabla de ARP entonces se entiende que con la configuración del preprocesador y el seguimiento a la IP y dirección MAC del dispositivo sonoff es suficiente para detectar el ataque. Al ejecutar el snort con el preprocesador igual configurado que en el ataque de “des-autenticación” se obtienen los mismos resultados exitosos de detección (Figura 85):

Figura 85 : Alerta de ataque ARP a Smart switch



Fuente: **Autor**

El mensaje de “Attempted ARP cache overwrite attack” hace referencia a que el ataque intenta sobre escribir la tabla ARP de la víctima configurada. Con la implementación de estas reglas en el snort y la configuración adicional previamente estipulada, ya el dispositivo smart switch puede encontrarse monitoreado.

2.3.5 Configurar las alertas encontradas para la notificación por correo electrónico

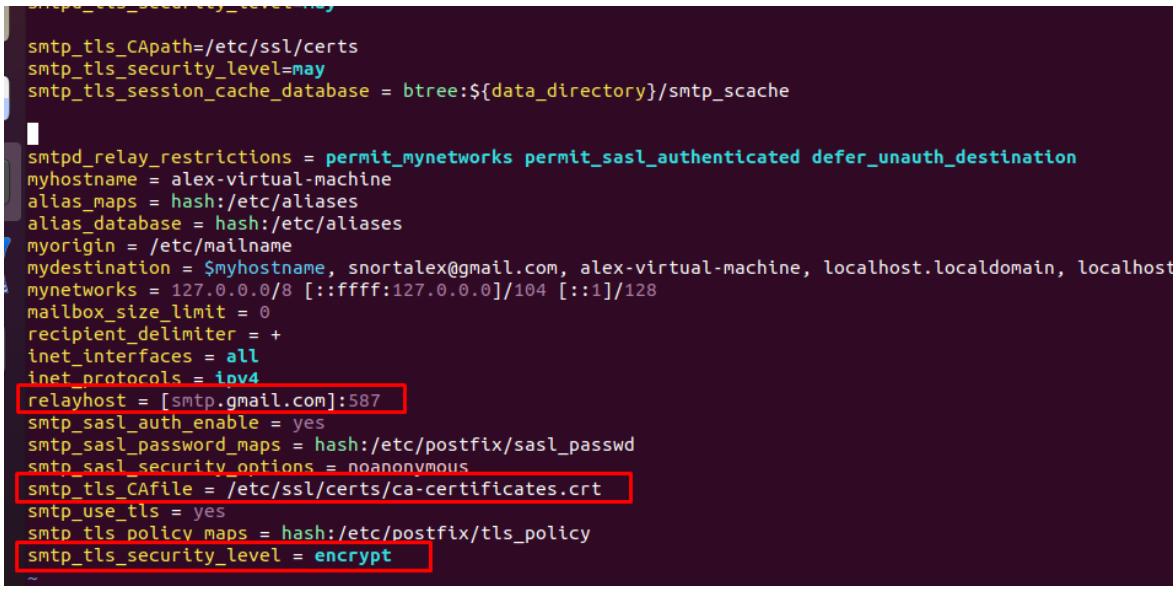
Para lograr esta parte de la instalación se encontró una guía ofrecida por un Blogger llamado “Manuel Franco” [53]. Dentro de la explicación se encuentra que para poder recibir alertas de snort por correo se deben hacer dos cosas adicionales, 1) configurar un servidor de correos, en este caso

con el software de postfix, 2) instalar un software de vigilancia de actividad del sistema llamado swatch.

Primero, la instalación de postfix, la cual no tuvo mayores complejidades, leyendo la guía de instalación del mismo Manuel el procedimiento es bastante claro y directo. Postfix es conocido por ser un agente de transferencia de correos para el sistema operativo de Ubuntu y es por eso que se configura para el desarrollo de la investigación actual.

Durante la instalación del servicio de postfix lo único importante para realizar es configurar el archivo main de este, el archivo se encuentra en la ruta **/etc/postfix/main.cf** y quedó con los siguientes parámetros configurados adicionales:

Figura 86 : Configuración main de servicio postfix



Fuente: **Autor**

Se configura el servicio para que envíe correos a la plataforma de Gmail, toda vez Gmail es una plataforma segura y confiable de utilizar (Figura 86). Se debe tener en cuenta el certificado y la ruta a utilizar, por defecto ya el servicio de postfix trae el certificado TLS solo toca agregarle la ruta al archivo y muy importante añadir la última línea para que se encripte la comunicación con el servicio de Gmail o no va a funcionar el envío de correos. Como una observación adicional, se tiene en cuenta que el servicio de Google solo permite enviar 500 correos al día por ser un usuario gratuito de Gmail, siguiendo la guía de Manuel se pudo correr el servicio con éxito [54].

Lo siguiente y más crucial es instalar el servicio de swatch. Swatch es el agente que revisa los logs y envía un correo cuando encuentra algún parámetro especificado en su archivo de configuración[53]. Lo único diferente que se debe hacer con esta instalación es crear nuestros propios patrones de búsqueda en el archivo de configuración de swatch ubicado en la ruta **/etc/swatchrc**

Figura 87 : Configuración servicio swatch

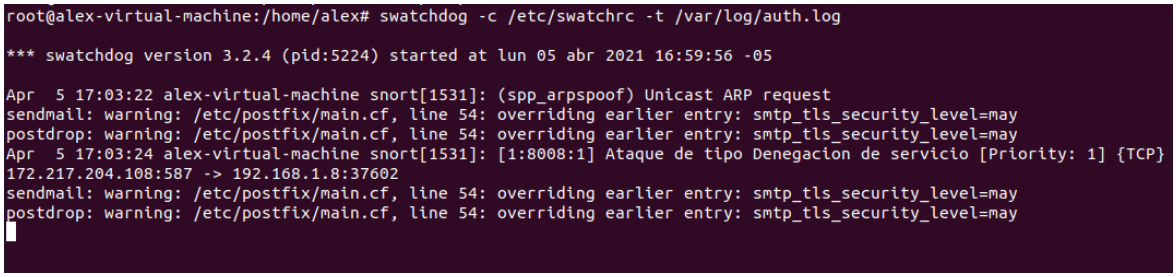
Fuente: **Autor**

El archivo de configuración de swatch siempre debe llevar la palabra clave “watchfor” lo cual simboliza el patrón de búsqueda que el programa va a monitorear (Figura 87), el patrón debe estar encerrada en dos slash (/) , de tal manera que quede así / texto / [55]. Dentro de lo configurado en el archivo encontramos un “echo” lo cual le indica al sistema que muestre de manera gráfica cada que envía un correo, el comando “mail” para que ejecute la tarea de enviar correos con postfix, “subject” se refiere al asunto del correo y el “threshold” es parecido al utilizado por snort, se refiere

a que envíe un solo correo cada 60 segundos siempre y cuando vea el parámetro buscado. Los parámetros buscados son “Priority 1” los cuales los definimos en la regla personal de Snort, el otro parámetro a buscar en los logs es la alerta de “ARP cache” el cual es el texto que aparece en el log de snort cada que el preprocesador encuentra algo sospechoso en los paquetes ARP y por último la alerta de “Mismatch” la cual también es para buscar los ataques de ARP en los logs, se hacen las reglas así para evitar falsos positivos de ARP.

Para ejecutar el comando de swatch y empezar a recibir correos se lanza el siguiente comando **swatchdog -c /etc/swatchrc -t /var/log/auth.log.** Donde -c indica la ruta del archivo de configuración, el cual definimos anteriormente y -t indica los logs que este va a monitorear. Como snort guarda los logs en el archivo de “auth.log” entonces cada que snort escriba un registro allí el software de swatch lo escanea y monitorea para mandar correos únicamente cuando se encuentre con los parámetros establecidos de búsqueda (Figura 88).

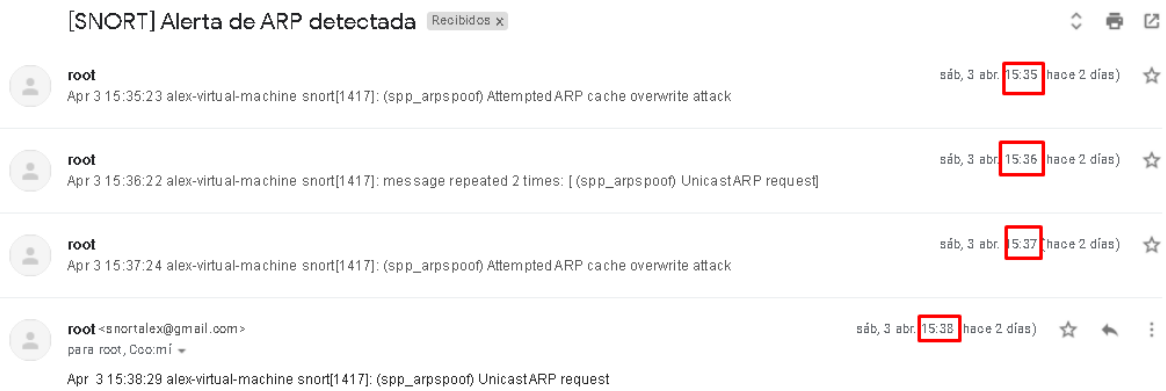
Figura 88 : Ejecución servicio de swatch



Fuente: **Autor**

Cada que swatch detecta actividad en el log de snort, este revisa su configuración y envía un correo de acuerdo con su threshold configurado, en este caso no más de un correo por minuto. Finalmente, en la bandeja de correo se observa de la siguiente manera:

Figura 89 : Correo Gmail con las alertas



Fuente: **Autor**

Como se puede observar en los tiempos de envió (Figura 89), se logra apreciar que el sistema respeta el tiempo estipulado de un minuto entre correo. La alerta enviada fue la de un ataque hacía el protocolo ARP, el cual el sistema detectó y alertó satisfactoriamente.

2.4 Fase 4

Ahora que se identificó la manera de alertar sobre ataques informáticos en la red IoT hacía el dispositivo Smart Switch, es importante que el usuario final se encuentre en condiciones para detener dicho ataque y tenga el conocimiento sobre cómo actuar. Mediante una documentación clara se realizó una planeación sobre cómo manejar incidentes de seguridad en la red IoT. La documentación se divide en dos partes, la primera parte explica cómo mejorar los dispositivos del entorno para reducir el riesgo de ataques informáticos y la segunda explica el plan para actuar en caso de un incidente cibernético.

2.4.1 Investigación sobre el manejo de incidentes de seguridad aplicables al hogar

Partiendo de que el sistema IoT estará en el hogar y que el administrador es el usuario final, se investigó sobre las mejores prácticas que se le puedan enseñar a este con el objetivo de evitar y detener ataques informáticos en su red IoT. En la primera parte se estableció unos pasos a tener en cuenta en la red, información para tener una configuración segura, dispositivos actualizados, hardware moderno y contraseñas robustas para evitar estos ataques. En la segunda parte se compartió el paso a paso para detener los ataques informáticos que se vean alertados por el sistema.

2.4.1.1 Buenas prácticas para la red IoT

Para responder adecuadamente a los incidentes de seguridad informática, es necesario seguir un modelo estructurado y planificado que permita un manejo apropiado de este. El ministerio de tecnologías de la información y las comunicaciones Colombiana (MINTIC) propone un modelo a seguir para la gestión y respuesta de un incidente de seguridad de la información, son 4 etapas: Preparación, Detección y análisis, Contención, Erradicación y Recuperación y actividades post-incidente [56].

1. **Preparación**

   La etapa de preparación consiste en cómo se puede tener las aplicaciones y la tecnología más seguras para evitar posibles fallas de seguridad en sus sistemas. Algunas de las practicas para asegurar la red y sus dispositivos se describen a continuación.

   Actualización de hardware y software

   En la actualidad los dispositivos tecnológicos se dividen en dos componentes principales, su hardware y su software. El hardware es compuesto por las partes físicas que componen el dispositivo como tal y la otra parte es el software. El software es la parte lógica que hace funcionar al hardware. A medida que pasa el tiempo los fabricantes desarrollan mejoras para sus dispositivos tecnológicos y los aplican al software mediante actualizaciones programadas.

Las actualizaciones pueden traer diferentes cambios en los dispositivos como por ejemplo: Corregir errores imprevistos, cambiar el diseño del programa, agregar funciones nuevas, quitar funciones obsoletas, agregar componentes de seguridad e incluso hasta mejorar el rendimiento del hardware.

Es muy importante mantener los dispositivos tecnológicos actualizados para evitar que sufran de daños o de fallas de seguridad descubiertas que antes no estaban. Los dispositivos no actualizados son muy fáciles de vulnerar por parte de los ciberdelincuentes y es por esto que la recomendación más importante es la de mantener los dispositivos actualizados de acuerdo con la tecnología del fabricante [57].

El proceso de actualización varía de acuerdo con el fabricante del dispositivo. Se ilustra cómo se actualiza el Sonoff Smart switch utilizado en la investigación.

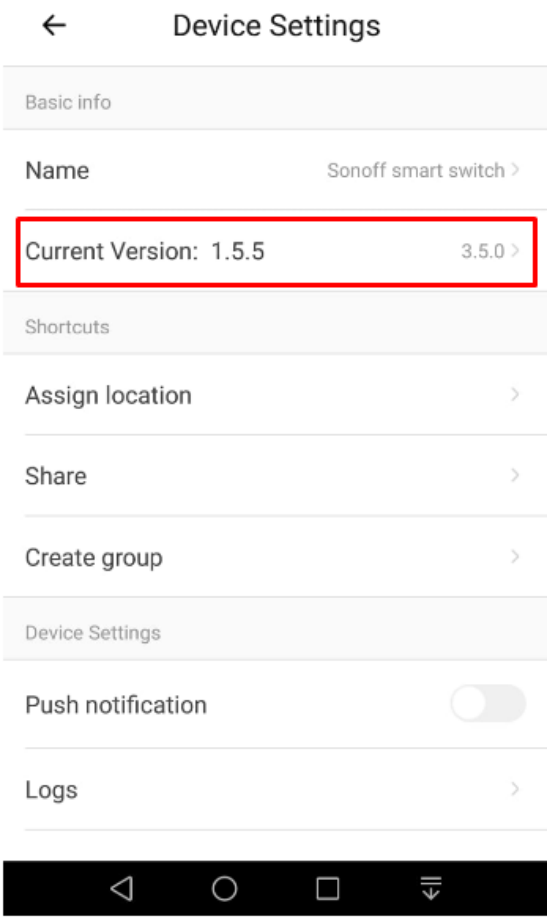
- Abrir la app eWeLink vista en el capítulo de "Marco teórico". A continuación seleccionamos el dispositivo y damos click en la opción derecha arriba, en los tres puntos (Figura 90)

Figura 90: Opciones para actualizar sistema Sonoff Smart switch

Fuente: **Autor**

- A continuación, seleccionamos la opción de “Current Version” (Figura 91)

Figura 91: Opciones para actualizar sistema Sonoff Smart switch



Fuente: **Autor**

- Por último, si el dispositivo requiere de una actualización aparecerá el siguiente botón(Figura 92)

Figura 92: Opciones para actualizar sistema Sonoff Smart switch

Fuente: **Autor**

Le seleccionamos el botón y esperamos que se actualice, muy importante no desconectar el internet del celular y tampoco desconectar el dispositivo Smart switch de la corriente de energía durante este proceso. Se recomienda revisar si hay actualizaciones nuevas por lo menos una vez cada tres meses, toda vez continuamente los fabricantes suelen lanzar parches de seguridad que pueden beneficiar a nuestros dispositivos.

Hardware:

El punto de conexión clave en el hogar IoT es el enrutador inalámbrico que conecta a todos los dispositivos de la casa en una red local y les brinda el acceso hacia internet. Los enrutadores inalámbricos disponen de un protocolo de seguridad que permite que la conexión entre los

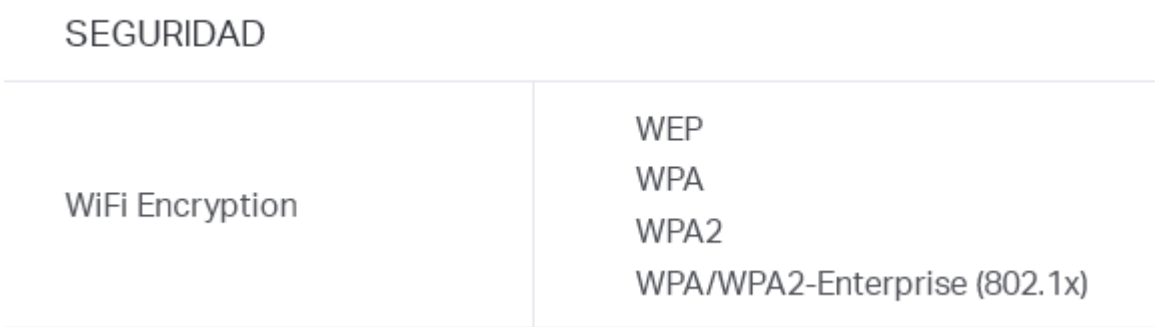
dispositivos y el mismo sean más seguras. Actualmente existen 4 protocolos seguros que utilizan los enrutadores inalámbricos y estos son: Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), Wi-Fi Protected Access 2 (WPA 2) y Wi-Fi Protected Access 3 (WPA 3).

El protocolo de seguridad más moderno es el WPA 3, lanzado en el año 2018. Se considera que el protocolo aún es bastante nuevo y muchos dispositivos IoT no cuentan con el soporte para conectarse a este tipo de redes y solo unas pocas marcas de enrutadores soportan esta tecnología. La recomendación es verificar en el empaquetado del dispositivo IoT si tiene soporte para WAP 3 y si es así hablar con el proveedor de internet para solicitar un enrutador con tecnología WAP 3 toda vez es el protocolo de conexión más seguro de la actualidad contra hackers [58].

Lo más común es que los enrutadores de la actualidad estén configurados con el protocolo WPA2, para revisar esto basta con observar la marca y la referencia del enrutador del que disponemos en el hogar. Se busca el enrutador en la página del fabricante y allí nos muestra el protocolo que utiliza. La recomendación técnica es llamar al proveedor del internet y pedirle que nos verifique si la conexión wifi funciona con el protocolo WPA2 y si es negativo que nos la habilite para el hogar.

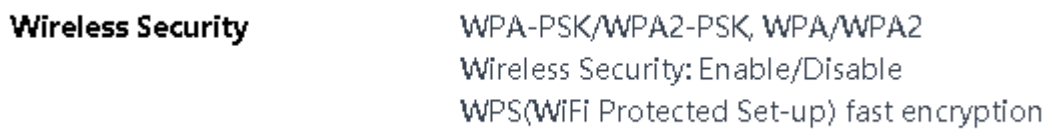
A continuación, se busca en las páginas de distintos fabricantes sobre la descripción de los enrutadores wifi que ofrecen en el mercado. Algunos ejemplos de enrutadores comerciales aparecen a continuación, de esta manera el usuario puede tener una referencia sobre cual opción buscar para conocer si su enrutador wifi tiene el protocolo WPA2 habilitado (Figura 93, Figura 94, Figura 95):

Figura 93: Protocolos wifi en enrutador tp-link



Fuente: Tp-link[59]

Figura 94: Protocolos wifi en enrutador Tenda



Fuente: Tenda [60]

Figura 95: Protocolos wifi en enrutador Ubiquiti

Fuente: Ubiquiti [61]

Con la evidencia de estos fabricantes se observa como sus dispositivos wifis vienen con el protocolo WPA2 en su seguridad, incluso el fabricante Ubiquiti ofrece el protocolo WPA3 en su dispositivo. Si el usuario desea instalar un enrutador moderno con el protocolo WPA3 lo puede hacer, la recomendación es que se contacte con el proveedor de internet para que pueda lograr una instalación exitosa y sin problemas.

Una sugerencia adicional y de buena práctica es cambiar la configuración de la red Wifi y ponerla como de tipo oculta. El usuario deberá llamar a su proveedor de internet e indicarle que requiere que su red wifi sea una red de tipo oculta para dispositivos, con esta configuración el nombre será oculto para los dispositivos inalámbricos de la zona. Al tener la red wifi oculta, los vecinos y las personas que intenten buscar la red con sus dispositivos no la podrán ver, solo se podrán conectar los clientes que tengan el nombre de la red wifi oculta.

Contraseñas

La configuración de una contraseña que cumpla con unos requisitos mínimos de complejidad es crucial para que la red wifi del usuario final sea más segura y menos probable a ser vulnerada por atacantes. Toda vez el dispositivo IoT se encuentra en una red wifi, es importante asegurar esta red con una contraseña que cumpla con los requerimientos de seguridad mínimos. A continuación, se establecen las mejores recomendaciones para tener una contraseña confiable, robusta y difícil de hackear:

- La contraseña debe tener mínimo 14 caracteres de largo, por ejemplo, una frase clave que solo el dueño la identifique.
- Las contraseñas no deben tener límite de caracteres máximos.
- La contraseñe debe tener mínimo un carácter especial en su composición, por ejemplo: “A!ex@nd3r*2901”
- La contraseña se debe cambiar mínimo dos veces al año.
- No tener “tips” o “ayudas” para adivinar la contraseña.
- No anotar la contraseña en lugares visibles o de fácil acceso.

Estas recomendaciones son dadas por el centro CIS (Center for internet security), un ente regulador y activo sobre los ataques informáticos de la actualidad [62].

2. Detección y Análisis

En esta etapa se hace la detección del incidente de seguridad. La detección del incidente se da mediante el alertamiento de correos que llegan a la bandeja de entrada del usuario configurada en el sistema de monitoreo. Todos los correos configurados por parte de Snort son de alto impacto y por lo tanto todos los correos que lleguen deben ser tratados con prioridad alta y de urgencia para actuar ante la situación.

Es muy importante tener en cuenta que para poder contar con el monitoreo y detección en la red, la máquina virtual configurada en la fase 3 debe estar en funcionamiento antes del ataque o de lo contrario no se podrá detectar el ataque informático a la hora de materializarse.

2.4.1.2 Respuesta a incidentes de ataques informáticos

3. Contención, erradicación y recuperación

Luego de haber detectado el incidente informático es importante contenerlo y erradicarlo. Se debe detener el incidente y reestablecer los sistemas afectados para poder recuperar el servicio.

Con una implementación exitosa del sistema de detección de intrusos planteado en la investigación presente el usuario final será notificado en tiempo real si su dispositivo IoT se encuentra vulnerado. En caso de que el usuario reciba correos electrónicos alertando sobre un ataque al sistema, se recomienda seguir el plan de acción a continuación y hacer los siguientes pasos para detener dicho ataque y mitigar la amenaza.

i) Desconectar el enrutador wifi de la toma de corriente para así poder apagarlo lo más pronto posible.
ii) Llamar al proveedor del servicio de internet en el hogar para hacer tres cosas: cambiar el nombre de la red wifi, cambiar la contraseña wifi de la red y por último pedirle que nos cambie la dirección IP publica para así borrar la IP anterior asignada al hogar y no dejarle huella al atacante.
iii) Denunciar el ataque a las autoridades locales, para Colombia se puede denunciar el ataque informático en el portal https://adenunciar.policia.gov.co/adenunciar/Login.aspx?ReturnUrl=/adenunciar/%20 Primero ingresamos a la opción de “Denuncia Virtual” y segundo a la opción de “Delitos informáticos” (Figura 96, Figura 97)

Figura 96 : Denuncias virtuales

Fuente: Policía Nacional [63]

Figura 97: Delitos informáticos

Fuente: Policía Nacional [63]

En este portal la policía nacional recibe las denuncias informáticas y asisten en el caso, ellos cuentan con personal informático capacitado para atender este tipo de situaciones. De esta manera se detiene el ataque informático en tiempo real, aseguramos nuevamente la red

wifi y denunciamos ante las autoridades pertinentes para tener un respaldo más confiable y recibir un apoyo profesional.

iv) Reestablecer los servicios, conectar el enrutador y configurar nuevamente el Smart switch a la red inalámbrica modificada.

4) Actividades post-incidente

Luego de ser víctimas de un ataque informático, es importante hacer una retroalimentación y ver en qué aspectos se puede mejorar para poder mitigar futuros ataques o para poder responder mejor ante futuros ataques. Se recomienda crear un documento en el cual se registrarán las "lecciones aprendidas" del incidente. En este documento se pueden documentar datos específicos sobre cómo fue el manejo de la situación, cuánto fue el tiempo de respuesta por parte de las autoridades, si hace falta anotar números telefónicos importantes y detalles así, algunos datos que se sugieren anotar son:

- Tiempo total del incidente
- Dificultades encontradas a la hora de seguir el documento
- Números telefónicos importantes para llamar, por ejemplo, el número del prestador de servicio de internet, la policía y hasta algún amigo conocedor del tema que pueda ayudar.
- Qué se puede mejorar para futuros incidentes
- Evaluar si los sistemas estaban actualizados o no antes del ataque
- Seguir las recomendaciones dadas por los expertos

Como documento adjunto (ver Anexo A) a la investigación se añade el archivo "PlandeRespuestaIncidentes.docx" para el usuario final, un documento fácil de seguir y entender para que este tenga más claridad sobre el plan de acción a tener en cuenta ante algún incidente en su red. En este documento se encuentran los pasos detallados de la sección 2.4.1 de esta investigación.

2.5 Fase 5

Finalmente, y lo más importante en toda investigación, es fundamental asegurarse de que el sistema implementado quede funcionando correctamente. En este apartado se documentaron pruebas de funcionamiento del sistema de alertas implementado para asegurar que no hubiese errores durante su ejecución.

2.5.1 Validar el funcionamiento del sistema de alertas

En el siguiente diagrama se describen los diferentes pasos que se deben cumplir para desplegar el sistema sin errores y con una ejecución efectiva (Figura 98).

Figura 98 : Checklist sistema completo

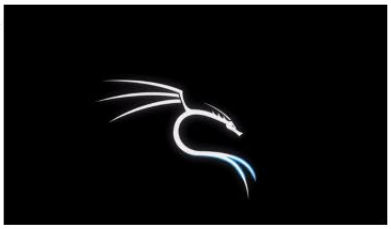
Check List

1. 

Iniciar Snort en modo detección de intrusos ☑

2. 

Iniciar servicio de swatchdog ☑

3. 

Ejecutar ataque a dispositivo ☑

4.
```
Commencing packet processing (pid=6496)
03/27-20:41:35.700297  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:36.700745  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:37.701161  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:38.702655  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:39.703096  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:40.704400  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:41.705410  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:42.706226  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:43.707331  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:44.707693  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:45.708762  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:46.228251  [**] [112:1:1] (spp_arpspoof) Unicast ARP request [**]
03/27-20:41:46.228528  [**] [112:1:1] (spp_arpspoof) Unicast ARP request [**]
03/27-20:41:46.709356  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:47.710573  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:48.711016  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
03/27-20:41:49.711813  [**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
```

Revisar las detecciones y alertas en snort ☑

5.
```
root@alex-virtual-machine:/home/alex# swatchdog -c /etc/swatchrc -t /var/log/aut
h.log

*** swatchdog version 3.2.4 (pid:12016) started at mié 14 jul 2021 13:12:21 -05

Jul 14 13:12:47 alex-virtual-machine snort: [1:8008:1] Ataque de tipo Denegacion
 de servicio [Priority: 1] {TCP} 192.168.1.222:31318 -> 192.168.1.7:80
sendmail: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
postdrop: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
Jul 14 13:13:47 alex-virtual-machine snort: [1:8008:1] Ataque de tipo Denegacion
 de servicio [Priority: 1] {TCP} 192.168.1.222:31318 -> 192.168.1.7:80
sendmail: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
postdrop: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
Jul 14 13:14:57 alex-virtual-machine snort: [1:8008:1] Ataque de tipo Denegacion
 de servicio [Priority: 1] {TCP} 192.168.1.222:31318 -> 192.168.1.7:80
sendmail: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
postdrop: warning: /etc/postfix/main.cf, line 54: overriding earlier entry: smtp
_tls_security_level=may
```

Revisar los logs en swatchdog ☑

6. 

Verificar correo electrónico ☑

7. 

Seguir el documento "Planderespuestaincidentes.docx" ☑

Fuente: Autor

Antes que nada, lo primero que se debe hacer es revisar si el dispositivo Smart switch tiene algún cambio en su IP local, toda vez si el sistema de detección de intrusos tiene una IP errónea entonces nunca se podrá detectar su tráfico. Si el dispositivo Smart switch cambió de IP, este se debe actualizar en el archivo de configuración de snort.

A continuación, iniciamos el sistema de detección de intrusos en modo de detección y que guarde los logs con syslog "**snort -d -h 192.168.1.0/24 -A console -c /etc/snort/snort.conf -i ens33 -s"** (Figura 99)

Figura 99: Inicialización servicio de Snort

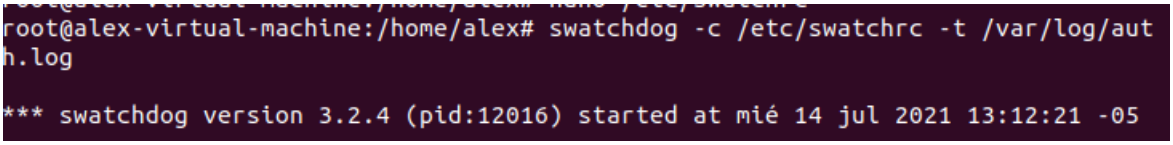
```
root@alex-virtual-machine:/home/alex# snort -d  -h 192.168.1.0/24 -A console -c
 /etc/snort/snort.conf -i ens33 -s
Running in IDS mode

        --== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined :  [ 80:81 311 383 591 593 901 1220 1414 1741 1830
 2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 777
7 7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 8243 8280 83
00 8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444 41080 50
002 55555 ]
PortVar 'SHELLCODE_PORTS' defined :  [ 0:79 81:65535 ]
PortVar 'ORACLE_PORTS' defined :  [ 1024:65535 ]
PortVar 'SSH_PORTS' defined :  [ 22 ]
PortVar 'FTP_PORTS' defined :  [ 21 2100 3535 ]
PortVar 'SIP_PORTS' defined :  [ 5060:5061 5600 ]
PortVar 'FILE_DATA_PORTS' defined :  [ 80:81 110 143 311 383 591 593 901 1220 1
```

Fuente: Autor

Con el servicio de snort arriba lo que sigue es iniciar el servicio de swatchdog, con este servicio el sistema va a estar buscando nuevos logs en el archivo de "Auth.log" configurado por snort, en este archivo el proceso de syslog escribe los logs que Snort detecta como alertas (Figura 100). El comando utilizado es el siguiente "**swatchdog -c /etc/swatchrc -t /var/log/auth.log"**.

Figura 100 : Inicialización servicio de Swatchdog



Fuente: Autor

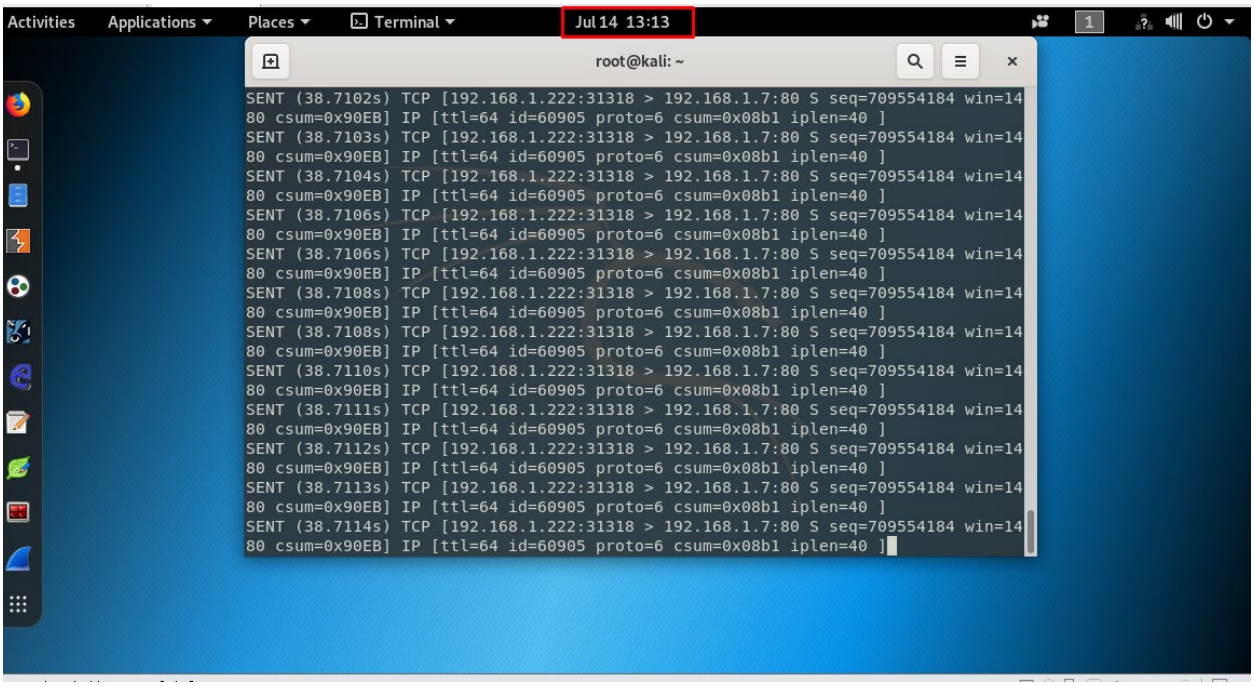
Ya con ambos sistemas arriba y en ejecución se procede a realizar un ataque de denegación de servicio al Smart switch mediante el comando **Nping** en la máquina virtual de Kali Linux (Figura 101).

Figura 101: Inicialización ataque DoS con Nping

Fuente: Autor

El ataque inicia a las 13:13 pm el dia 14 de Julio (Figura 102),

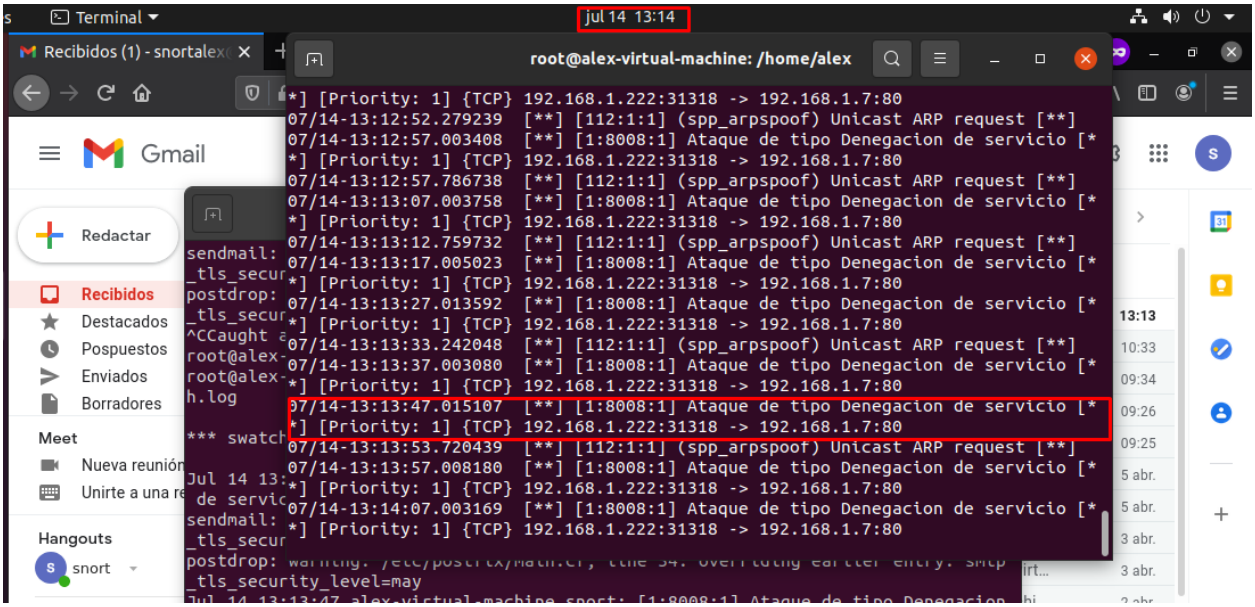
Figura 102 : Ejecución ataque DoS hacía víctima



Fuente: Autor

El sistema de detección efectivamente detecta y alerta la consola a la hora del ataque, 13:13 pm (Figura 103).

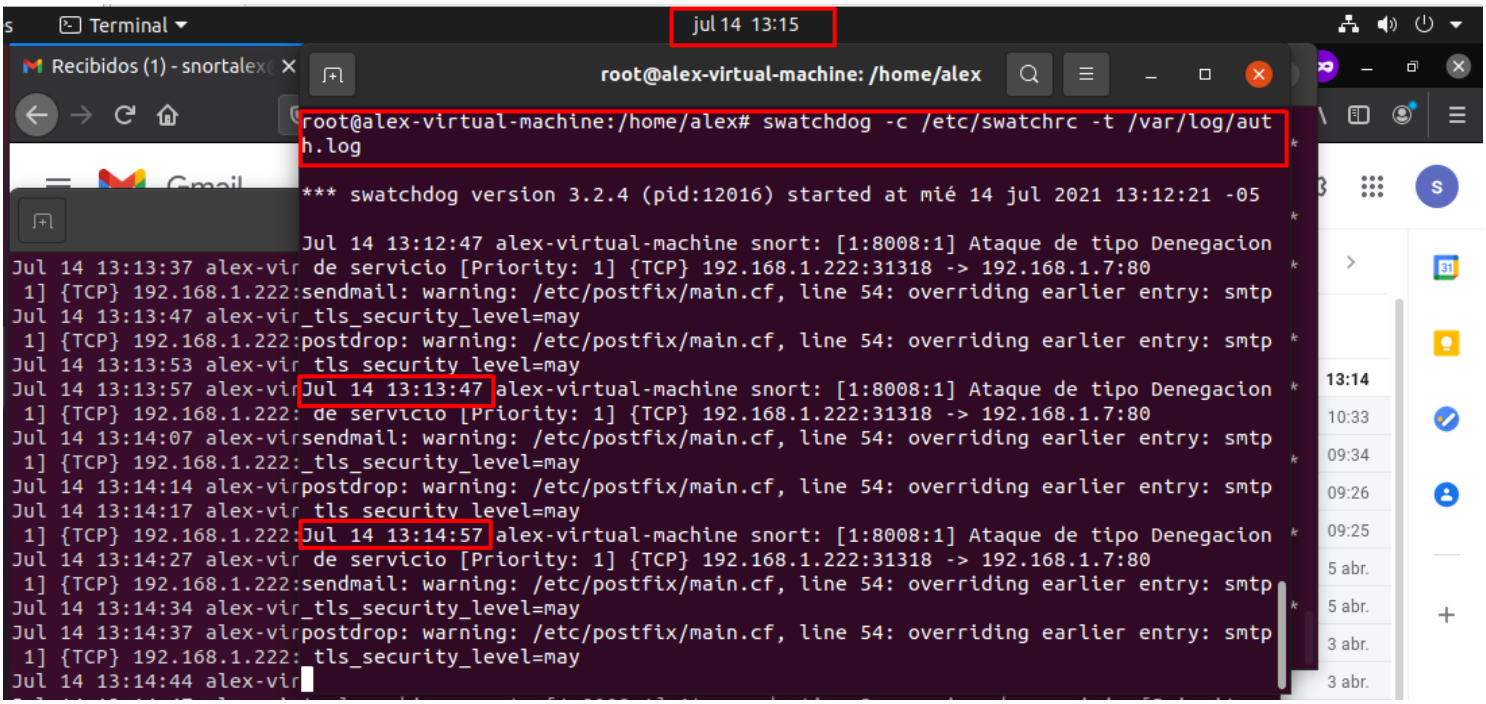
Figura 103 : Detección y alerta del ataque DoS



Fuente: Autor

Se revisa el servicio de swatchdog y se encuentra que se encontraron los logs del ataque alertado a las 13:13 pm (Figura 104),

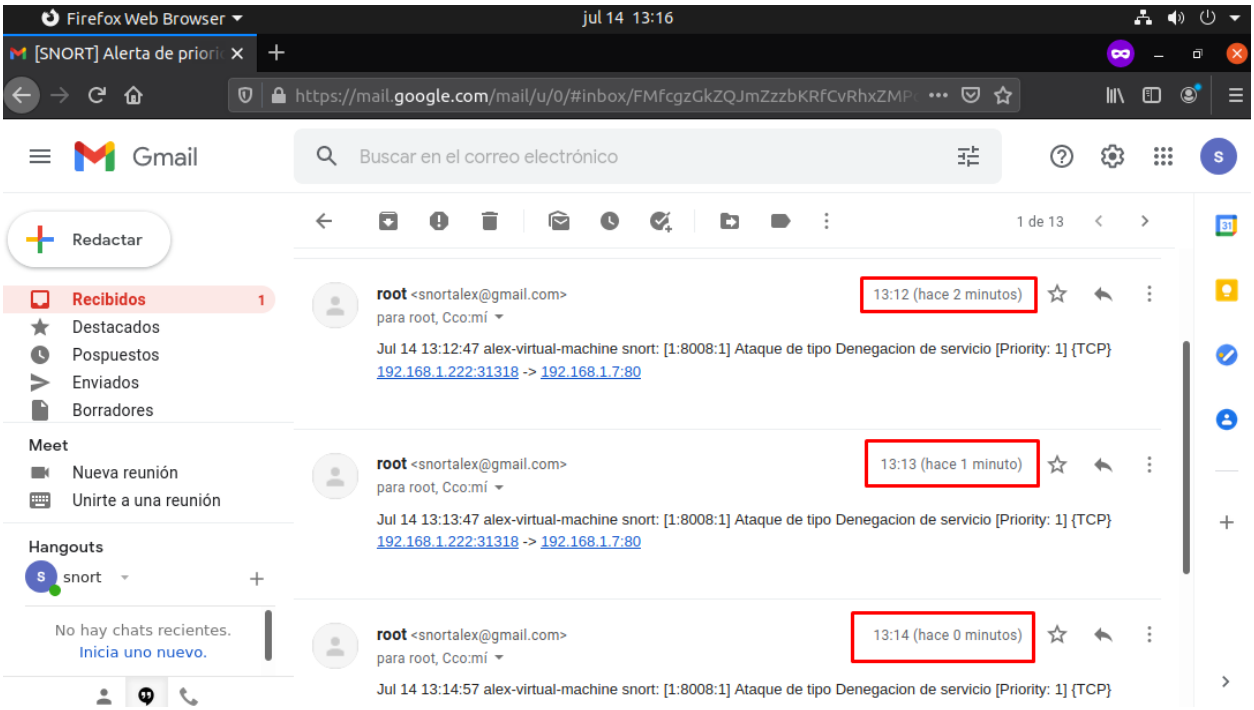
Figura 104 : Servicio de logs de Swatchdog



Fuente: Autor

Por último, como el servicio de swatchdog reportó las alertas significa que debió enviar esos logs al correo, ingresamos al correo configurado y encontramos las alertas detectadas en la hora precisa y con intervalos de 1 minuto entre cada correo tal como se configuró el treshold anteriormente (Figura 105).

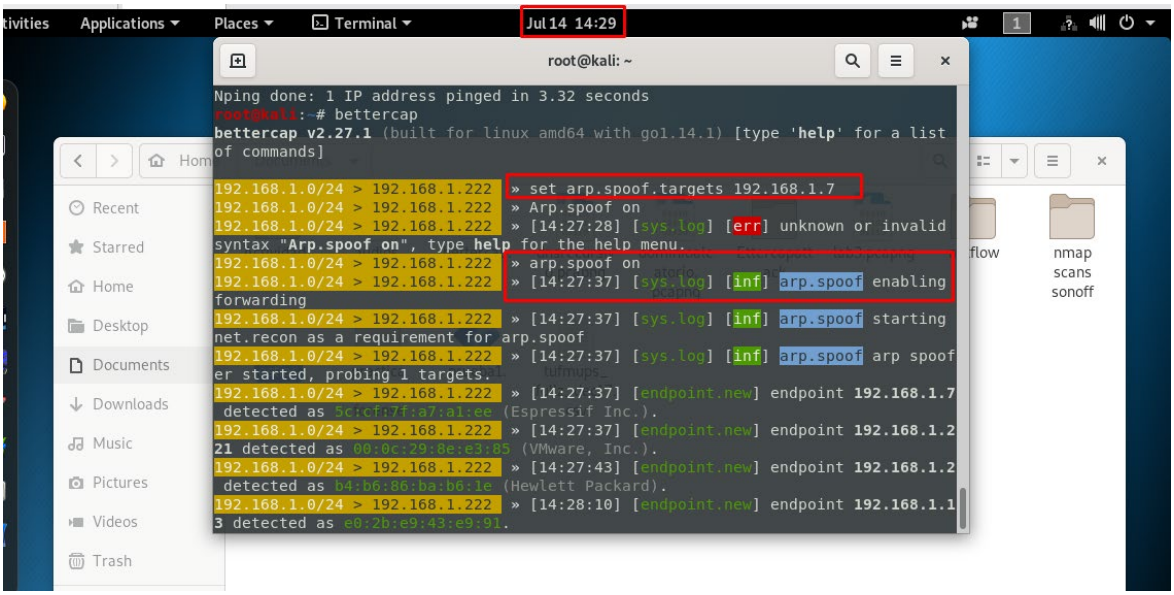
Figura 105: Revisión de notificación en el correo



Fuente: Autor

Se realiza otra prueba para verificar una de las alertas con los ataques de ARP. Iniciamos el programa de Bettercap, para realizar un ataque de ARP poisoning, se inicia el ataque a las 14:27 pm (Figura 106).

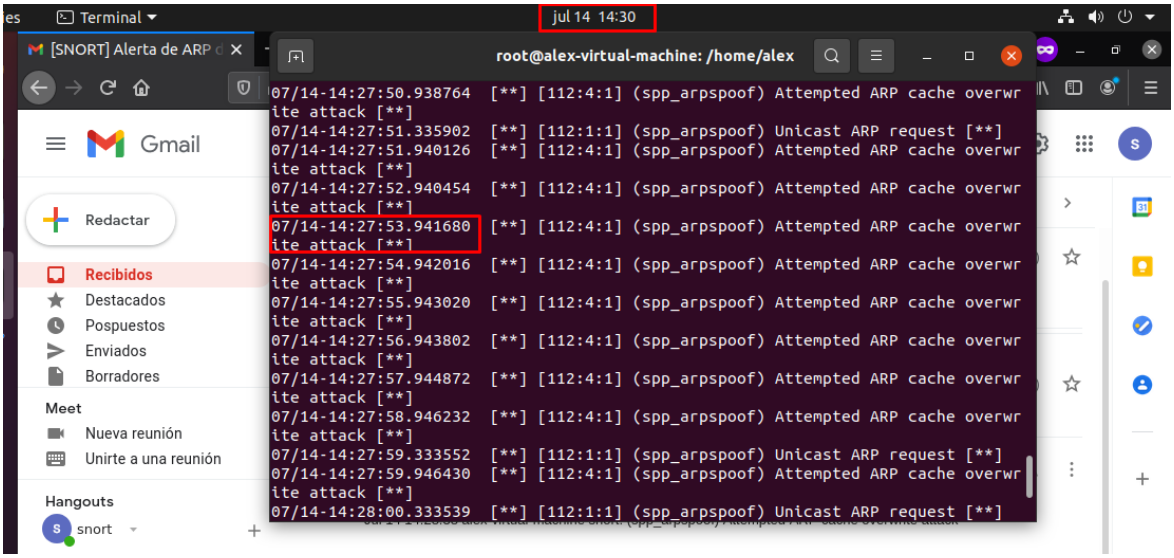
Figura 106: Ejecución ataque ARP con Bettercap



Fuente: Autor

El sistema de Snort efectivamente detecta el ataque a esa hora, 14:27 pm y lo alerta (Figura 107),

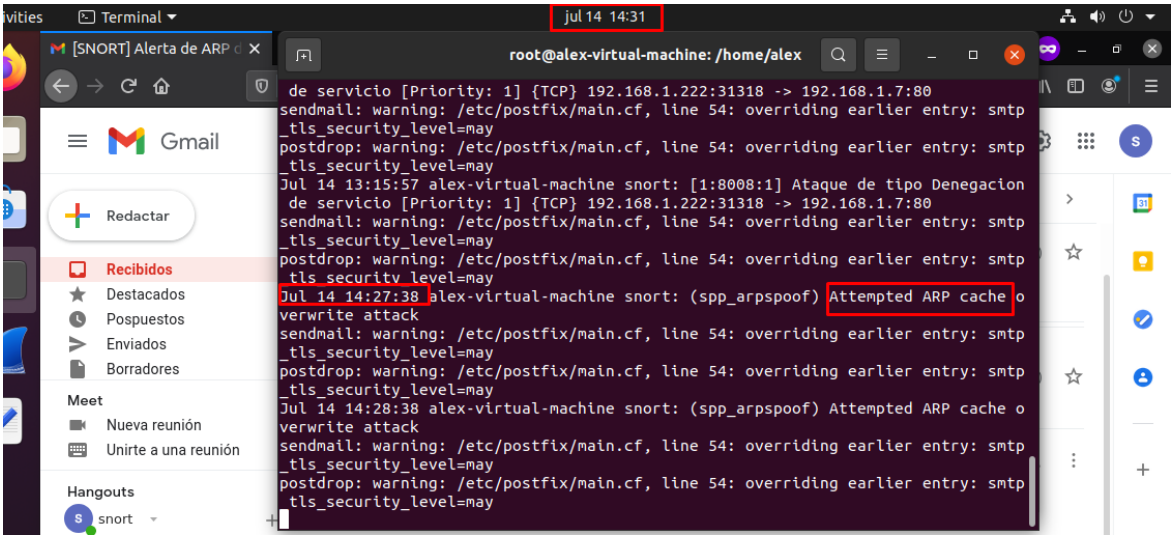
Figura 107 : Detección y alerta de ataque ARP



Fuente: Autor

Se procede a revisar los logs en swatchdog y allí se encuentran los logs correspondientes de la alerta ARP a las 14:27 pm (Figura 108).

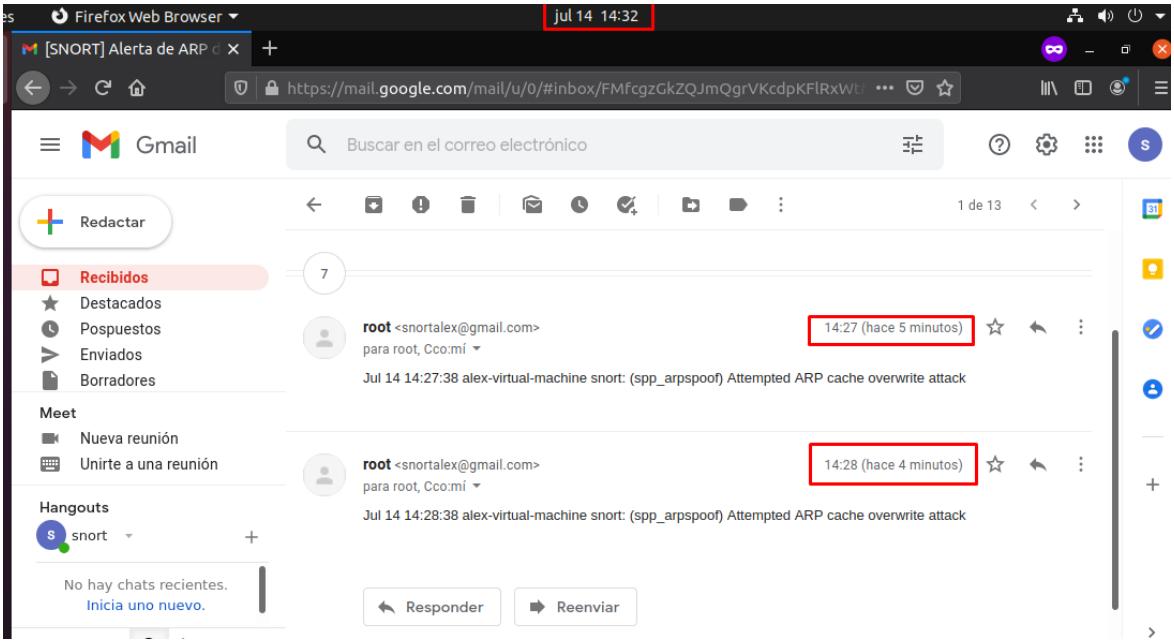
Figura 108 : Revisión logs en Swatchdog



Fuente: Autor

Se revisa nuevamente el buzón de correo, se encuentran las alertas en la bandeja de entrada a la misma hora 14:27 pm, con su respectivo 1 minuto entre correos como está configurado él envió (Figura 109).

Figura 109: Revisión de notificaciones en correo electrónico



Fuente: Autor

Al hacer el seguimiento del ataque y su respectiva detección, alerta y notificación al usuario se logra evidenciar que el sistema funciona correctamente y está en capacidad para funcionar en un hogar donde se tenga un dispositivo Smart switch que se quiera asegurar de una mejor manera ante los atacantes informáticos.

3. Conclusiones y recomendaciones

3.1 Conclusiones

- Ante la creciente demanda de dispositivos IoT en el mercado y el incremento de ataques informáticos a estos dispositivos es importante plantear mecanismos que ayuden a mitigarlos y detenerlos. Con la presente investigación se logró detectar los ataques dirigidos al dispositivo IoT, detener el ataque con la ayuda del detector de intrusos y notificar al usuario mediante correos electrónicos para que este pueda responder ante la situación como se puede evidenciar en el capítulo 2.6 Fase 5.

- La identificación de los ataques pertinentes al dispositivo fue fundamental para segmentar todas las otras categorías de ataques existentes que son aplicables al dispositivo IoT seleccionado como se evidencia en el apartado 2.2.2. Al enfocarnos en el pilar de la disponibilidad se logró establecer cuatro de los ataques informáticos más comunes a este tipo de dispositivos.

- Al clasificar los indicadores de compromiso mediante la herramienta brindada por la CVSS (capitulo 2.3.3) se logra tener un panorama más claro sobre el impacto y la severidad de los ataques investigados. El ataque con más puntuación por su impacto fue sin duda el de DoS, los otros ataques tienen una clasificación media, gracias a esta calculadora se pueden medir los ataques con respecto al otro según su impacto.

- La arquitectura implementada logró monitorear en tiempo real el tráfico de red hacia y saliente del dispositivo IoT, con esta arquitectura se logra detectar, alertar, rechazar paquetes y notificar al usuario final sobre su dispositivo IoT (capitulo 2.4).

- Con la investigación realizada y el montaje del laboratorio se lograron replicar los ataques informáticos en diferentes escenarios propuestos, los ataques fueron exitosos (capitulo 2.3.2), el dispositivo IoT fue siempre afectado en su disponibilidad y se evidenció que

efectivamente este tipo de dispositivos es muy vulnerable ante ataques de denegación de servicio y los que implican la manipulación de los registros ARP.

- Los ataques expuestos en la investigación son ataques de red (capitulo 2.2.1), por esto la importancia de tener un sistema de detección de intrusos en la red toda vez estos dispositivos no cuentan con ningún tipo de protección.

- Se realiza un plan de respuesta a incidentes para los ataques establecidos en la investigación (capitulo 2.5.1), con este plan el usuario final debe poder detener un ataque informático en tiempo real a su dispositivo y mejorar la seguridad en su red para evitar ataques futuros.

3.2 Trabajo a futuro

Considerando los resultados positivos que se obtuvieron con la arquitectura propuesta y el desempeño del dispositivo Smart switch, en un futuro trabajo se podría considerar atacar y explotar la seguridad en dispositivos IoT que ya vengan con protocolos de seguridad red implementados. Esto con el objetivo de probar que tan bueno son los protocolos invertidos en estos dispositivos y proponer maneras de mejorar la tecnología de seguridad que los fabricantes disponen en los dispositivos IoT que si cuentan con ello.

A. Anexo: PlandeRespuestaIncidentes.docx

Este documento es anexado a la presente investigación. Allí se encuentra un plan de acción dirigido al usuario final dueño de un Smart switch IoT. Con este documento el usuario final tendrá un plan detallado con pasos a seguir para asegurar mejor su red ante ataques informáticos y tendrá un plan de acción para seguir en caso de algún incidente en su red.

Bibliografía

[1] A. Bera, “80 Mind-Blowing IoT Statistics (Infographic),” 2019.

[2] Knud Lasse Lueth, “State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating,” *iot analytics*, 2018.

[3] Á. O. Arenas, “Medellín estrena dos semáforos inteligentes: ¿para qué sirven?,” *El Colombiano*, 2019.

[4] V. Schmid, “How to Choose the Perfect Smart Home Assistant for You,” *Digitized*, 2019.

[5] J. C. Hu, “How one lightbulb could allow hackers to burgle your home,” 2018.

[6] Avast, “Avast Smart Home Security Report 2019,” p. 17, 2019, [Online]. Available: https://cdn2.hubspot.net/hubfs/486579/avast_smart_home_report_feb_2019.pdf.

[7] A. Marotti, “Smart devices hacked in digital home invasions,” 2019.

[8] U. 42, “2020 Unit 42 IoT Threat Report,” *paloalto*, 2020. https://unit42.paloaltonetworks.com/iot-threat-report-2020/.

[9] “Nokia: Threat Intelligence Report 2020,” *Comput. Fraud Secur.*, vol. 2020, no. 11, p. 4, 2020, doi: 10.1016/s1361-3723(20)30115-9.

[10] S. Basic, “Sonoff smart switch basic,” 2016. https://sonoff.itead.cc/en/products/sonoff/sonoff-basic.

[11] L. Shenzhen Cool House Technology Co., “eWeLink - Smart Home,” 2020. .

[12] C. A. Tejada Villalba, “Que es el Chip Wifi Esp8266,” 2019. http://iot.alltimetech.com.co/blog/blog5/.

[13] M. Harwood, “Network+ Exam Cram: Wireless Networking,” *Junio 9*, 2009. https://www.pearsonitcertification.com/articles/article.aspx?p=1329709&seqNum=4.

[14] M. Rouse, “CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance),” *Octubre*, 2008. https://searchnetworking.techtarget.com/definition/CSMA-CA.

[15] K. Aseem, “What is the Difference between WPA2, WPA, WEP, AES, and TKIP?,” *Octubre 3*, 2017. https://helpdeskgeek.com/networking/what-is-the-difference-between-wpa-and-wep-wireless-security-encryption/.

[16] Cybersecuritychallenge, “TOPIC : Fundamentals of computer networks : 4 Layer TCP / IP Model,” 2018, [Online]. Available:

https://www.cybersecuritychallenge.org.uk/app/uploads/2019/08/Lesson-Plan-4-Layer-TCP-IP-Model.pdf.

[17] N. Lord, “What are Indicators of Compromise?,” 2018. https://digitalguardian.com/blog/what-are-indicators-compromise.

[18] Trendmicro, “Indicators of Compromise.” https://www.trendmicro.com/vinfo/us/security/definition/indicators-of-compromise.

[19] M. Rouse, “Address Resolution Protocol (ARP),” 2019. .

[20] Ccna, “The ARP table on a Cisco router,” 2020. [Online]. Available: https://study-ccna.com/the-arp-table-on-a-cisco-router/.

[21] Radware, “ARP Poisoning,” 2019. https://security.radware.com/ddos-knowledge-center/ddospedia/arp-poisoning/.

[22] R. Villarreal, “The Better Ettercap... Bettercap!,” *Julio 20*, 2019. https://bestestredteam.com/2019/07/20/the-better-ettercap-bettercap/.

[23] Cloudflare, “What is a man-in-the-middle attack?,” 2018. https://www.cloudflare.com/learning/security/threats/man-in-the-middle-attack/.

[24] Paloalto, “What is a denial of service attack (DoS) ?,” 2020. .

[25] Angelawood, “Mobile Deauthentication Attacks,” *Julio 13*, 2019. .

[26] J. Deogirikar and A. Vidhate, “Security attacks in IoT: A survey,” *Proc. Int. Conf. IoT Soc. Mobile, Anal. Cloud, I-SMAC 2017*, pp. 32–37, 2017, doi: 10.1109/I-SMAC.2017.8058363.

[27] X. H. Lulu Liang, Kai Zheng, Qiankun Sheng, “A Denial of Service Attack Method for an IoT System,” 2016, [Online]. Available: https://ieeexplore-ieee-org.itm.elogim.com:2443/document/7976501.

[28] B. Cong, S. Qiankun, G. Xingren, Z. Kai, and H. Xin, “A Tool for Denial of Service Attack Testing in IoT,” *Xi’an Jiaotong-Liverpool Univ.*, 2016, [Online]. Available: http://etisconf.com/2016/wp-content/uploads/2016/08/6_1_Bao_Guan_Sheng_Zheng_Huang.pdf.

[29] Anakod, “ESP8266 security vulnerability! Krack attack,” 2017.

[30] C. Pastorino, “Aclarando KRACK Attack, la vulnerabilidad descubierta en WPA2,” 2017.

[31] M. Taneja, “An Analytics Framework to Detect Compromised IoT Devices using Mobility Behavior,” pp. 38–43, 2013.

[32] L. Rudman and B. Irwin, “Dridex: Analysis of the traffic and automatic generation of IOCs,”

2016 Inf. Secur. South Africa - Proc. 2016 ISSA Conf., pp. 77–84, 2016, doi: 10.1109/ISSA.2016.7802932.

[33] H. Cho, S. Lee, N. Kim, B. Kim, and J. Park, "Method of Quantification of Cyber Threat based on Indicator of Compromise," *2018 Int. Conf. Platf. Technol. Serv.*, pp. 1–6, 2018.

[34] A. De Procesos, R. Rpa, and R. P. A. Cibernético, "Riesgo cibernético y RPA."

[35] A. G. Mayank Verma,Dr. Ponnurangam Kumarguru, Shuva Brata Deb, "Analysing Indicator of Compromises for Ransomware: Leveraging IOCs with Machine Learning Techniques," *2018 Int. Conf. Platf. Technol. Serv.*, 2018, [Online]. Available: https://ieeexplore-ieee-org.itm.elogim.com:2443/stamp/stamp.jsp?tp=&arnumber=8587409&tag=1.

[36] SONOFF, "SONOFF Basic R2 10A Smart WiFi Wireless Light Switch, Universal DIY Module for Smart Home Automation Solution, Works with Amazon Alexa & Google Home Assistant, Works with IFTTT, No Hub Required," 2019. https://www.amazon.com/Wireless-Universal-Automation-Solution-Assistant/dp/B07KP8THFG/ref=sr_1_3?dchild=1&m=A2QBZNDSRNO1IV&marketplaceID=ATVPDKIKX0DER&qid=1584732710&s=merchant-items&sr=1-3&th=1.

[37] Espressif, "Espressif Achieves the 100-Million Target for IoT Chip Shipments," Shanghai, China, 2018.

[38] nmap, "Nmap," *Actualización Agosto 10*, 2019. .

[39] "Real Academia Española." 2020, [Online]. Available: https://dle.rae.es/relevante.

[40] B. Bhushan and A. K. Rai, "Security vulnerabilities , attacks and countermeasures in wireless sensor networks at various layers of OSI reference model : A Survey," no. July, pp. 288–293, 2017.

[41] Andrés Méndez Barco, "GUÍA DE SEGURIDAD(CCN-STIC-423)INDICADORES DE COMPROMISO (IOC)," *Ed. y Cent. Criptológico Nac.*, 2015.

[42] N. Kamarinakis and S. David, "Kick them out," 2018. https://github.com/k4m4/kickthemout.

[43] A. Hackerkitty, "Ettercap," 2014. https://hackerkitty.wordpress.com/tag/ettercap/.

[44] "Bettercap," 2019. https://www.bettercap.org/intro/.

[45] F. Inc, "CVSS score," *1995-2020*. .

[46] S. Q. Tascon and J. Z. Jiménez, "Modelo para la gestión de incidentes de seguridad en redes industriales SCADA a través del algoritmo de predicción Filtro Kalman," 2019.

[47] C. and/or its Affiliates, “Snort,” 2021. https://www.snort.org/.

[48] S. A. R. Shah and B. Issac, “Performance comparison of intrusion detection systems and application of machine learning to Snort system,” *Futur. Gener. Comput. Syst.*, vol. 80, pp. 157–170, 2018, doi: 10.1016/j.future.2017.10.016.

[49] T. snort Project, “Snort User Manual,” 2020. http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node17.html#SECTION00321500000000000000.

[50] E. Acar, “What is Snort?,” 2018. https://medium.com/@acaremrullahkku/what-is-snort-547916bece5f.

[51] C. and/or its Affiliates, “Snort FAQ,” 2021. https://www.snort.org/faq/readme-thresholding.

[52] J. Esler, “Flow matters,” 2011. https://blog.snort.org/2011/09/flow-matters.html.

[53] M. Franco, “Snort: Aviso de alertas por correo,” *2017*. https://manuelfrancoblog.wordpress.com/2017/10/27/snort-aviso-de-alertas-por-correo/.

[54] M. Franco, “Configurar servidor de correo Postfix,” *2017*. https://manuelfrancoblog.wordpress.com/2017/10/09/configurar-servidor-de-correo-postfix/.

[55] C. Ltd, “Swatch,” *2019*. http://manpages.ubuntu.com/manpages/trusty/man1/swatch.1p.html.

[56] MinTic, “Guía para la Gestión y Clasificación de Incidentes de Seguridad de la Información,” 2016, [Online]. Available: https://mintic.gov.co/gestionti/615/articles-5482_G21_Gestion_Incidentes.pdf.

[57] Incibe, “Seguridad en la instalación y uso de dispositivos IoT :,” p. 27, 2020, [Online]. Available: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia-de-seguridad-iot.pdf.

[58] E. Raphaely, “A Complete Guide To Wireless (Wi-Fi) Security,” *Secure w2*, 2020. https://www.securew2.com/blog/complete-guide-wi-fi-security.

[59] Tp-link, “router tp-link,” 2021. https://www.tp-link.com/latam/home-networking/wifi-router/tl-wr850n/.

[60] Tenda, “Router tenda,” 2021, [Online]. Available: https://www.tendacn.com/en/product/AC6 v5.html.

[61] Ubiquiti, “Router Ubiquiti,” 2021. [Online]. Available: https://store.ui.com/products/unifi-ap-6-lite.

[62] CIS, “CIS Password Policy Guide,” 2021, [Online]. Available: https://www.cisecurity.org/white-papers/cis-password-policy-guide/.

[63] P. Nacional, “Policia Nacional de Colombia,” 2021. https://adenunciar.policia.gov.co/adenunciar/default.aspx.