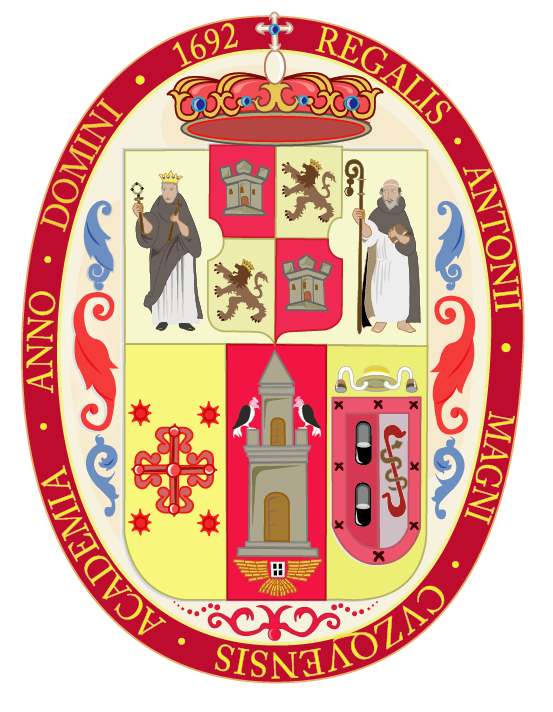
**UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO**

**ESCUELA DE POSGRADO**

**MAESTRÍA EN CIENCIAS**



**PROYECTO FINAL**

DNS Spoofing/DNS Cache Poisoning

**INTEGRANTES**

* Jhon Anderson Bellido Serrano
* Katherine Yessenia Serpa Condori
* David Roberth Falcón Aguilar
* Jhon Abraham Aguirre Carrazco

**DOCENTE**

Dr. Edwin Carrasco Poblete

**Cusco, enero 2025**

**Presentación**

El siguiente trabajo desarrolla la implementación de una red cliente-servidor para demostrar las vulnerabilidades y penetración (pentesting) aplicando el ataque DNS Spoofing, aplicando como contramedida y mitigación el uso de Iptables y Arptables, herramientas de comandos Linux para el filtrado y detección de paquetes en el firewall de un sistema. El procedimiento de ataque y contramedida están guiados y reglamentados por la normativa NIST SP 800-115, realizando la Evaluación de Seguridad de la Información (ESI) bajo las tres etapas primarias: planificación, ejecución y post ejecución

El objetivo es comprobar la mitigación frente a una amenaza sobre los clientes y servidores DNS en base a una guía estandarizada para la ESI, realizando pruebas simuladas fieles a ataques reales para realizar conclusiones consistentes y reales en entornos de trabajo institucionales.

En el capítulo I se desarrolla el marco teórico, con definiciones y conceptos teóricos que dan el sustento bibliográfico al trabajo.

El capítulo II toca el diseño experimental, donde con apoyo de figuras, tablas y otros recursos se hace la preparación del entorno de trabajo, la metodología proporcionada por el NIST y la preparación de los recursos virtualizados para las pruebas antes y después.

El capítulo III se enfoca en la experimentación, mostrando detalles técnicos que pueden guíar el procedimiento y la metodología para el ataque y mitigación sobre el DNS Spoofing.

Finalmente, los capítulos IV y V tratan sobre detalles finales como discusión de resultados y conclusiones respectivamente obtenidos de la experimentación, para finalizar y no dejando de ser importante el capítulo VI de las referencias y citas contenidas dentro del trabajo.

**Introducción**

Los ataques informáticos sobre centros de datos y servidores están a la orden del día, mencionando el último evento por parte del banco Interbank en nuestro país (RPP, 2024), son muchos los servicios que prestan los servidores dentro de un data center tales como DNS, HTTPS, Proxy, Firewall, FTP, etc., por lo que preocuparse por accesos y ataques informáticos es una tarea obligatoria.

En el presente informe daremos a conocer el ataque sobre los servidores DNS: DNS Spoofing y un caso de mitigación sobre comandos Linux: Iptables y Arptables. En base a las guías y recomendaciones NIST se reproducirá la intrusión al real en una red LAN con resultados evidentes sujetos a contramedida.

**Índice general**

[**I.**](#_heading=h.gjdgxs) **Marco Teórico** 7

[**a.**](#_heading=h.30j0zll) **NIST SP 800-115** 7

[**1.1.**](#_heading=h.1fob9te) **Estructura de la NIST SP 800-115** 7

[**1.1.1.**](#_heading=h.3znysh7) **Fase de Planeación** 7

[**1.1.2.**](#_heading=h.2et92p0) **Fase de Ejecución** 7

[**1.1.3.**](#_heading=h.tyjcwt) **Fase de Postevaluación** 8

[**1.2.**](#_heading=h.3dy6vkm) **Pruebas de Intrusión en la NIST SP 800-115** 8

[**b.**](#_heading=h.2s8eyo1) **Flujo de actividades de hacking ético** 11

[**c.**](#_heading=h.3rdcrjn) **Descripción del servicio de red objeto de evaluación** 13

[**d.**](#_heading=h.lnxbz9) **Descripción de la técnica de ataque seleccionada** 15

[**e.**](#_heading=h.44sinio) **Impacto de la técnica de ataque en el servicio de red** 18

[**f.**](#_heading=h.2jxsxqh) **Descripción de las técnicas de mitigación y contramedidas** 19

[**II.**](#_heading=h.4i7ojhp) **Diseño experimental** 25

[**a.**](#_heading=h.2xcytpi) **Descripción de escenarios de pruebas** 25

[**b.**](#_heading=h.3whwml4) **Descripción de equipos utilizados** 26

[**c.**](#_heading=h.qsh70q) **Descripción de aplicaciones utilizadas** 26

[**d.**](#_heading=h.1pxezwc) **Configuración de equipos y aplicaciones** 28

[**III.**](#_heading=h.49x2ik5) **Experimentación** 29

[**a.**](#_heading=h.2p2csry) **Conducción de las pruebas de ataque aplicando NIST SP 800 – 115** 29

[**b.**](#_heading=h.1rvwp1q) **Documentación de resultados de los ataques** 49

[**c.**](#_heading=h.3q5sasy) **Implementación de contramedidas** 51

[**d.**](#_heading=h.25b2l0r) **Conducción de las pruebas de ataque post mitigación** 51

[**e.**](#_heading=h.kgcv8k) **Documentación de resultados de los ataques** 51

[**IV.**](#_heading=h.34g0dwd) **Discusión de resultados** 52

[**V.**](#_heading=h.1jlao46) **Conclusión** 53

[**VI.**](#_heading=h.43ky6rz) **Referencias** 54

**Índice de figuras**

[**Figura 1** Metodología de prueba de penetración de cuatro etapas 9](#_heading=h.1t3h5sf)

[**Figura 2** Pasos de la fase de ataque con bucle invertido a la fase de descubrimiento 10](#_heading=h.4d34og8)

[**Figura 3** Se muestra las 5 etapas de hacking ético en un sistema informático 11](#_heading=h.17dp8vu)

[**Figura 4** Resolución de una solicitud DNS a la dirección IP correcta 14](#_heading=h.26in1rg)

[**Figura 5** Ataque de compromiso del servidor DNS 17](#_heading=h.35nkun2)

[**Figura 6** Proceso de validación. 23](#_heading=h.z337ya)

[**Figura 7** Proceso de consultas de DNS recursivas 23](#_heading=h.3j2qqm3)

[**Figura 8** Escenario DNS Spoofing 25](#_heading=h.1ci93xb)

[**Figura 9** Generación del certificado y firma digital por el servidor DNS 27](#_heading=h.3as4poj)

[**Figura 10** Windows 8.1 (víctima) IP: 192.168.1.9 29](#_heading=h.147n2zr)

[**Figura 11** Windows 10 (anfitriona) IP: 192.168.1.6 30](#_heading=h.3o7alnk)

[**Figura 12** DEBIAN (atacante) IP: 192.168.1.14 30](#_heading=h.23ckvvd)

[**Figura 13** Activación del modo routing 30](#_heading=h.ihv636)

[**Figura 14** Instalamos locate y ettercap en DEBIAN 31](#_heading=h.1hmsyys)

[**Figura 15** Instalación de ettercap modo gráfico y visualización de su versión 31](#_heading=h.41mghml)

[**Figura 16** Comando para editar el archivo etter.dns 32](#_heading=h.2grqrue)

[**Figura 17** Edición del archivo etter.dns y conocimiento del comando para realizar el ataque DNS Spoofing 33](#_heading=h.vx1227)

[**Figura 18** Comando de ataque DNS Spoofing 34](#_heading=h.3fwokq0)

[**Figura 19** Visualización de máquinas víctimas potenciales 35](#_heading=h.1v1yuxt)

[**Figura 20** Estado normal de windows 36](#_heading=h.4f1mdlm)

[**Figura 21** Apertura de una nueva pestaña de terminal 36](#_heading=h.2u6wntf)

[**Figura 22** Instalación de servicio DNSMASQ y edición del dnsmasq.conf 37](#_heading=h.19c6y18)

[**Figura 23** Edición del archivo dnsmasq.conf 38](#_heading=h.3tbugp1)

[**Figura 24** Reinicio del servicio dnsmasq 38](#_heading=h.28h4qwu)

[**Figura 25** Verificación del estado del servicio dnsmasq 39](#_heading=h.nmf14n)

[**Figura 26** Instalación de YERSINIA 40](#_heading=h.37m2jsg)

[**Figura 27** Interfaz gráfica de YERSINIA 40](#_heading=h.1mrcu09)

[**Figura 28** Configuración de servicio DHCP 41](#_heading=h.46r0co2)

[**Figura 29** Configuración de YERSINIA parte II 41](#_heading=h.2lwamvv)

[**Figura 30** Búsqueda y generación de máquinas a vulnerar 42](#_heading=h.111kx3o)

[**Figura 31** Generación DHCP Spoofing 42](#_heading=h.3l18frh)

[**Figura 32** Verificación del cambio de la puerta de enlace y servidor DHCP con la IP de la máquina atacante 43](#_heading=h.206ipza)

[**Figura 33** Interrupción de la búsqueda DHCP 44](#_heading=h.4k668n3)

[**Figura 34** Cambio de direcciones DHCP y DNS efectiva 44](#_heading=h.2zbgiuw)

[**Figura 35** Detección del harvesting y rastro de carga 45](#_heading=h.1egqt2p)

[**Figura 36** Ping con direcciones atacadas (intranetdiresamdd y google) 46](#_heading=h.3ygebqi)

[**Figura 37** Remplazo de la Gateway con la IP 192.168.1.14 conforme a la figura N37 47](#_heading=h.2dlolyb)

[**Figura 38** Páginas sin redireccionamiento a web falsa (sin índex listo) 47](#_heading=h.sqyw64)

[**Figura 39** Paquetes interceptados desde la máquina atacante 48](#_heading=h.3cqmetx)

[**Figura 40** Intercambio de direcionjes IP con la dirección del atacante 49](#_heading=h.4bvk7pj)

[**Figura 41** Páginas Web capturadas por el atacante 50](#_heading=h.2r0uhxc)

[**Figura 42** Tráfico de paquetes tomados por el atacante 51](#_heading=h.1664s55)

**Índice de tablas**

[**Tabla 1** Consulta y respuesta DNS con datos de DNSSEC 24](#_heading=h.1y810tw)

[**Tabla 2** Descripción de equipos utilizados para la experimentación 26](#_heading=h.2bn6wsx)

1. **Marco Teórico**
2. **NIST SP 800-115**

Según Scarfone et al. (2008) Basado en la guía NIST SP 800-115 National Institute of Standards and Tecnólogy, nos menciona que:

Es un documento técnico que proporciona un marco organizado para llevar a cabo evaluaciones de seguridad en sistemas informáticos. Desarrollada por el National Institute of Standards and Technology (NIST), su propósito es asistir a las organizaciones en la detección y reducción de vulnerabilidades, la valoración de la eficacia de sus medidas de seguridad y la protección de su infraestructura tecnológica.

La guía se enfoca en el análisis técnico de la seguridad, incluyendo pruebas de penetración (pentesting) y otros métodos para recopilar información e identificar posibles brechas. Su uso es común en sectores como el gubernamental, financiero y sanitario, así como en empresas tecnológicas que buscan resguardar sus datos, uno de sus activos más importantes.

Scarfone et al. (2008) también indica en su guía la siguiente estructura del NIST SP-800-115 que incluye 3 fases esenciales a desarrollar:

* 1. **Estructura de la NIST SP 800-115**

La NIST SP 800-115 se estructura en tres etapas fundamentales que cubren todo el proceso de evaluación de seguridad: Planificación, Implementación y Análisis Posterior. Cada una de estas fases es esencial para desarrollar un programa de pruebas de seguridad eficaz.

* + 1. **Fase de Planeación**

La etapa de planificación es crucial para definir los objetivos, el alcance y los criterios de éxito de las pruebas de seguridad. En esta fase, el equipo de pruebas colabora de manera cercana con los responsables del sistema para establecer:

* **Qué sistemas o componentes serán evaluados** (servidores, aplicaciones, redes, etc.).
* **Qué tipo de pruebas se realizarán**, ya sea pruebas de penetración, análisis de vulnerabilidades, revisiones de configuración, entre otros.
* **Cuáles son las reglas del ejercicio**, es decir, las condiciones bajo las cuales se llevarán a cabo las pruebas (por ejemplo, pruebas sin afectar la disponibilidad de los sistemas).
* **Establecimiento de responsabilidades**, se define quiénes estarán involucrados en las pruebas y los roles de cada participante.

Además, esta etapa permite coordinar y definir claramente los roles y responsabilidades de cada involucrado, minimizando posibles malentendidos. De esta manera, se garantiza que las pruebas de seguridad se lleven a cabo de forma controlada, sin afectar la operatividad de los sistemas esenciales de la organización.

* + 1. **Fase de Ejecución**

Durante la fase de ejecución, se realizan las pruebas técnicas, las cuales pueden abarcar actividades como:

* **Recolección de información**: Consiste en obtener información sobre el sistema objetivo mediante técnicas como el escaneo de puertos, la identificación de servicios activos y la detección de versiones de software. Este proceso es fundamental para comprender las posibles vulnerabilidades y puntos débiles.
* **Identificación de vulnerabilidades**: En esta fase, los evaluadores identifican vulnerabilidades en la infraestructura de TI, como configuraciones deficientes, software desactualizado o ausencia de parches de seguridad. Para ello, emplean tanto herramientas automatizadas como técnicas manuales para detectar posibles fallos.
* **Pruebas de penetración**: En esta etapa, los especialistas intentan aprovechar las vulnerabilidades identificadas para determinar su nivel de riesgo. Se trata de una simulación controlada de un ataque real, diseñada para evaluar la resistencia de los sistemas ante posibles intrusiones.

A lo largo del proceso, los evaluadores deben actuar con ética y dentro del marco legal, garantizando que sus acciones no interrumpan el funcionamiento de los sistemas ni comprometan la seguridad de la información.

* + 1. **Fase de Postevaluación**

La fase de postevaluación es fundamental para examinar los resultados alcanzados y crear informes detallados que contengan:

* **Resumen de vulnerabilidades encontradas**: Se muestran las vulnerabilidades organizadas según su nivel de criticidad (bajo, medio, alto), acompañadas de su posible impacto en los sistemas.
* **Recomendaciones para mitigar las vulnerabilidades**: En este apartado se proponen acciones correctivas que las organizaciones pueden adoptar para reforzar sus defensas.
* **Evaluación del desempeño de los controles de seguridad**: Los resultados brindan a las organizaciones la oportunidad de evaluar si sus controles son efectivos o si requieren ajustes.

Esta fase también contempla la presentación de los resultados a las partes interesadas, garantizando que la alta dirección esté al tanto y lista para tomar decisiones fundamentadas en los hallazgos.

Tenemos más detalle sobre las metodologías de intrusión, Henry (2017), que se basó en la guía NIST SP 800-115, nos recalca que:

* 1. **Pruebas de Intrusión en la NIST SP 800-115**

Las pruebas de intrusión se definen como pruebas de seguridad en las que los evaluadores simulan ataques reales con el objetivo de identificar formas de eludir las medidas de seguridad de una aplicación, sistema o red de datos. Además, se señala que la mayoría de las pruebas de intrusión implican la búsqueda de la explotación combinada de vulnerabilidades en varios sistemas, de manera que puedan ser utilizadas para obtener privilegios de acceso más elevados que los que se podrían alcanzar explotando una sola vulnerabilidad.

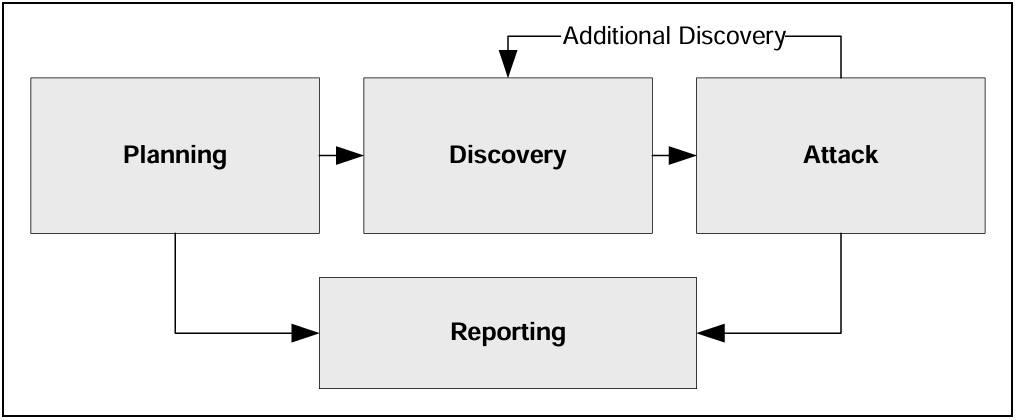
Según la NIST SP 800-115, las pruebas de intrusión pueden ser utilizadas para determinar:

* La manera en que el sistema tolera los patrones de ataques del mundo real.
* El nivel de sofisticación que un atacante necesita para comprometer con efectividad el sistema.
* Las medidas adicionales que se deben emplear para mitigar las amenazas contra el sistema.
* La habilidad de los defensores para detectar los ataques y responder apropiadamente a estos.

Henry (2017) nos menciona la importancia de la metodología de prueba de penetración basado en la NIST SP 800-115 el cual consta de 4 fases y resalta la importancia de ello citándolo en la siguiente figura:

Como se aprecia en la Figura 1, se muestra un esquema de las fases de una prueba de intrusión:

**Figura 1** *Metodología de prueba de penetración de cuatro etapas*

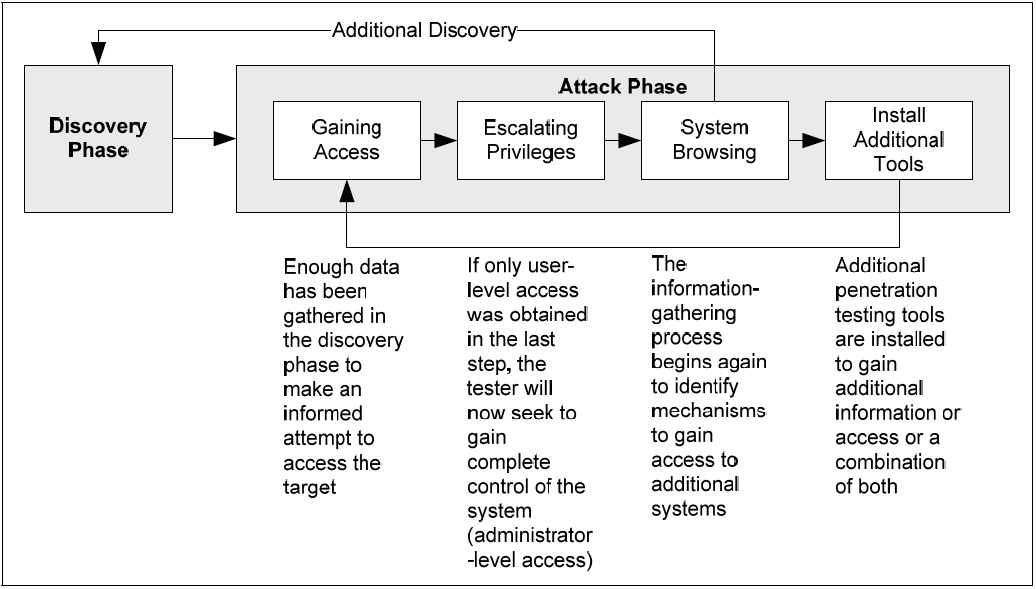


*Nota.* Fases de una Prueba de Intrusión. Tomado de NIST SP 800-115.

A continuación, se describen cada una de las fases:

* **Fase de Planificación**: Se establecen las normas que deben cumplirse durante la prueba de intrusión, se definen los objetivos a lograr y se gestionan las aprobaciones requeridas. Se preparan las condiciones técnicas y organizativas necesarias para garantizar el éxito de la prueba. En esta etapa, no se lleva a cabo ninguna prueba de seguridad.
* **Fase de Descubrimiento**:
  + Se lleva a cabo el escaneo y la recopilación de información sobre la infraestructura tecnológica de la organización.
  + Se lleva a cabo la identificación de vulnerabilidades utilizando la información obtenida sobre los servicios, la infraestructura tecnológica y otros datos que permitan realizar búsquedas en bases de datos de vulnerabilidades públicas o internas.
* **Fase de Ejecución**: En este proceso, se verifica la validez de las vulnerabilidades previamente identificadas, siendo esta la fase principal del proceso (figura 2). Si un ataque tiene éxito, se debe aislar y documentar detalladamente la vulnerabilidad, además de proponer medidas para mitigarla. Las actividades internas realizadas incluyen:
  + Obtener privilegios de acceso: Si la información recolectada en la fase anterior es adecuada, durante esta actividad se puede lograr obtener privilegios de acceso al sistema.
  + Escalada de privilegios: Si en la actividad anterior solo se obtuvieron privilegios de acceso de nivel bajo (por ejemplo, usuario básico), en esta etapa, los evaluadores intentan obtener el control total del sistema, similar al que tendrían los administradores.
  + Navegación dentro del sistema: Con el control del sistema obtenido en la etapa anterior, los evaluadores deben buscar información adicional que les permita entender mejor la existencia de métodos para obtener privilegios de acceso en sistemas secundarios. Si se descubren nuevos datos o información, estos se agregarían a los resultados de la fase de Descubrimiento y se planificaría la explotación de las nuevas vulnerabilidades encontradas.
  + Instalación de herramientas adicionales: Durante el proceso de explotación de vulnerabilidades, puede ser necesario instalar herramientas que faciliten la recopilación de información adicional, la obtención de otros privilegios de acceso, o ambas acciones simultáneamente.

**Figura 2** *Pasos de la fase de ataque con bucle invertido a la fase de descubrimiento*



*Nota.* Actividades específicas de la Fase de Ejecución. Tomado de NIST SP 800-115.

* **Fase de Documentación y Reporte**: Se lleva a cabo de manera simultánea con las demás fases de la siguiente forma:
  + En la fase de Planificación se documenta el Plan de Evaluación o las Reglas de Interacción.
  + En la fase de Descubrimiento se almacenan los reportes generados por los escaneadores de vulnerabilidades e informaciones útiles obtenidas a través de otros medios.
  + En la fase de Ejecución se almacenan los reportes generados por las herramientas de explotación de vulnerabilidades.
  + Al concluir la prueba de intrusión, se genera un reporte con la descripción de las vulnerabilidades encontradas, presenta una puntuación de riesgos y brinda una guía sobre cómo mitigar las debilidades descubiertas.

1. **Flujo de actividades de hacking ético**

Sánchez (2015) nos dice en su libro, El hacking es el acto de ingresar a un sistema informático privado desde el ciberespacio, aprovechando fallos en la seguridad, vulnerabilidades o mediante la obtención fraudulenta de contraseñas haciéndose pasar por usuarios autorizados. Según el científico en computación Peter G. Neuman (1984), esta práctica implica una intrusión malintencionada con el objetivo de explorar datos en busca de información valiosa. En resumen, se trata de un acceso no autorizado a sistemas informáticos de terceros.

En la metodología en hacking ético, siguiendo un procedimiento de 5 etapas fundamentales ayudará al usuario a alcanzar su objetivo:

**Figura 3** *Se muestra las 5 etapas de hacking ético en un sistema informático*



Nota. Las 5 etapas del ethical hacking. Tomado de <https://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/4919/00005096.pdf?sequence=1&isAllowed=y>

A continuación, se describen las 5 fases:

Sánchez, (2015) define cada una de estas 5 etapas que forman parte de la metodología de Ethical Hacking.

1. **Reconocimiento**

En la primera fase, conocida como reconocimiento, se reúne evidencia e información sobre los objetivos a atacar. Este proceso puede ser de tipo pasivo o activo.

El reconocimiento pasivo consiste en recopilar datos sobre el objetivo sin que este lo detecte, mientras que el reconocimiento activo emplea herramientas y técnicas que interactúan mínimamente con el objetivo, lo que aumenta el riesgo de ser descubierto. Sin embargo, la información obtenida en esta etapa puede no ser completamente fiable, ya que proviene de fuentes externas.

1. **Escaneo y Enumeración**

En la segunda fase, denominada escaneo y enumeración, se aprovecha la información recopilada previamente, como dominios, direcciones u otros datos, para aplicar activamente herramientas y técnicas que permitan obtener detalles más precisos sobre los objetivos. Esto puede ir desde realizar un simple barrido de ping o un escaneo de red para identificar sistemas y puertos activos, hasta ejecutar un análisis avanzado de vulnerabilidades.

1. **Ganar Acceso**

La tercera fase, considerada la más emocionante, es donde se vulneran los controles y medidas de seguridad para obtener acceso no autorizado. Durante esta etapa, se lleva a cabo el ataque, que puede variar en complejidad, desde ingresar a una red inalámbrica con una contraseña débil hasta ejecutar técnicas avanzadas, como un desbordamiento de búfer o una inyección SQL contra una aplicación web.

1. **Mantener Acceso**

En la cuarta fase, conocida como mantenimiento del acceso, los atacantes buscan establecer y conservar un punto de entrada para regresar al sistema o máquina ya comprometida. Estos dispositivos infectados pueden ser utilizados para escanear nuevas redes, capturar tráfico o lanzar nuevos ataques. Para mantener el acceso, se pueden emplear troyanos, rootkits u otros métodos de persistencia.

1. **Cubrir Huellas**

En la fase final, denominada cobertura de huellas, los atacantes buscan ocultar su actividad y evadir la detección por parte de los sistemas y equipos de seguridad. Las técnicas más comunes incluyen la eliminación o modificación de archivos de registro, el ocultamiento de archivos o directorios y el uso de conexiones cifradas con el sistema comprometido. Por esta razón, la ausencia de archivos de registro puede ser un indicio de que un sistema ha sido vulnerado.

En el caso de los ethical hackers, durante esta fase deben restringir sus acciones para evitar cualquier daño o pérdida de datos, asegurando que su intervención no afecte la integridad del sistema evaluado.

En conclusión, los atacantes malintencionados suelen identificar un solo punto de acceso y lo explotan para alcanzar sus fines ilícitos. En contraste, los ethical hackers analizan de manera exhaustiva todos los posibles puntos de entrada y vulnerabilidades dentro del alcance definido, considerando el tiempo y los objetivos autorizados para la evaluación.

Es fundamental que todo el proceso sea documentado detalladamente para la elaboración del informe final. Además, es importante destacar que la función de los ethical hackers no es corregir las vulnerabilidades, sino identificarlas y proponer soluciones viables.

1. **Descripción del servicio de red objeto de evaluación**

Sabemos que el DNS permite a los usuarios conectarse a sitios web mediante nombres de dominio en lugar de direcciones IP. Es por ello que (Quiroz & Goodwin, 2024) lo definen a continuación:

1. **¿Qué es un servidor DNS?**

Un servidor DNS (también llamado servidor de nombres) es un tipo especial de software de servidor que utiliza una base de datos DNS para responder a las consultas relacionadas con el Sistema de Nombres de Dominio. Como los servidores DNS suelen estar alojados en máquinas dedicadas, las computadoras que ejecutan estos programas también se conocen como servidores DNS.

Gracias al DNS, los usuarios de Internet pueden ingresar un dominio, es decir, un nombre fácil de recordar, en la barra de direcciones de su navegador. Cada dominio en Internet tiene al menos una dirección IP, que es necesaria para que las computadoras se comuniquen en la red. Un servidor DNS tiene conocimiento de las asociaciones entre dominios y direcciones IP, o bien sabe a qué otro servidor DNS debe enviar la solicitud. Así, al acceder a un sitio web, se realiza primero una solicitud a uno o más servidores DNS antes de conectarse al sitio. Esto hace que los servidores DNS sean componentes esenciales para el funcionamiento de Internet.

(Quiroz & Goodwin, 2024) mencionan 4 tipos de servidores DNS, entre los cuales tenemos:

1. **Diferentes tipos de servidores DNS**

* **Servidor DNS recursive**

El servidor DNS recursivo es el primer paso en una consulta DNS, encargándose de buscar y devolver una dirección IP. Utiliza datos almacenados en caché o consulta a otros servidores DNS si es necesario. Cada búsqueda optimiza el proceso mediante el almacenamiento en caché, lo que acelera el acceso a sitios web. Estos servidores son generalmente proporcionados por los proveedores de servicios de internet (ISP).

* **Servidor de nombres raíz**

Cuando un servidor DNS recursivo no cuenta con los datos en caché, envía una solicitud DNS al servidor de nombres raíz. Este servidor recibe la consulta y la redirige a un servidor de nombres de dominio de nivel superior (TLD), según la extensión del sitio solicitado, como .com, .org o .net. Existen 13 servidores raíz DNS principales, gestionados por la Corporación de Internet para la Asignación de Nombres y Números (ICANN).

* **Servidor de nombres de dominio de primer nivel (TLD)**

Los servidores TLD almacenan información sobre los dominios que comparten la misma extensión. Esto implica que existen servidores TLD específicos para dominios con extensiones como .com, .org y .net. Cuando la consulta alcanza el servidor TLD adecuado, este la redirige al servidor de nombres autoritativo.

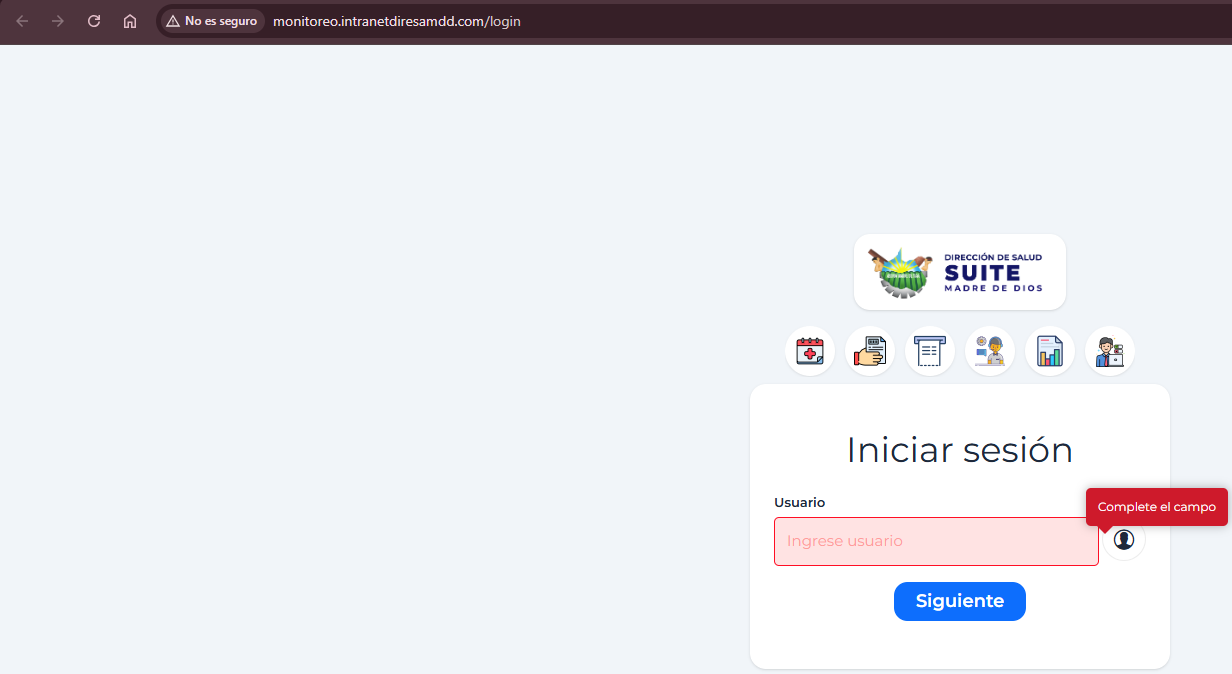
* **Servidor de nombres autoritativo**

El proceso de recuperación de una dirección IP se basa en servidores DNS autorizados que almacenan registros con información de dominio. Si se encuentra la dirección correcta, se envía al solucionador recursivo; de lo contrario, se genera un error. Un sistema DNS eficiente debe ser rápido e imperceptible para el usuario, y la distribución de carga entre cuatro tipos de servidores ayuda a evitar sobrecargas.

1. **Descripción del servicio de red**

Este enlace (<http://monitoreo.intranetdiresamdd.com/login>) nos conduce a una página de inicio de sesión la intranet perteneciente a la Dirección Regional de Salud de Madre de Dios (DIRESA MDD). Dado que se trata de una intranet, el acceso está restringido a usuarios autorizados, como empleados o colaboradores de la DIRESA MDD. Sin credenciales de acceso, no es posible obtener más detalles sobre el contenido o las funcionalidades específicas de la plataforma.

**Figura 4** *Resolución de una solicitud DNS a la dirección IP correcta*



Nota. Dirección del servicio web de evaluación. <http://monitoreo.intranetdiresamdd.com/login>

* Proceso de resolución de nombres en el sistema de Domain Name System (DNS)

1. El cliente que accede por primera vez a un nombre de dominio o URL envía una solicitud al solucionador de DNS.
2. El solucionador DNS dirige la solicitud a un servidor raíz.
3. El servidor raíz, que es un servidor de nombres autorizado, responde proporcionando la dirección de un servidor correspondiente al dominio de nivel superior.
4. Luego, el solucionador DNS consulta al servidor TLD, el cual contiene los registros DNS del dominio de nivel superior.
5. Como respuesta, el solucionador DNS recibe la dirección IP del servidor DNS autorizado para el dominio solicitado.
6. A continuación, el solucionador DNS solicita al servidor DNS autorizado la dirección IP del servidor de origen donde está alojado el sitio web.
7. El servidor DNS autorizado proporciona al solucionador DNS la dirección IP del servidor de origen.
8. El solucionador DNS transmite la dirección IP al cliente.
9. Con esta información, el cliente puede establecer comunicación con el servidor de origen del sitio web solicitado.
10. Finalmente, el servidor de origen envía los datos del sitio web al cliente.

En resumen, cuando un usuario ingresa una URL en su navegador, el sistema DNS traduce ese nombre en una dirección IP mediante una serie de consultas a servidores DNS, hasta encontrar la dirección del servidor donde está alojado el sitio web. Una vez obtenida la dirección IP, el cliente puede conectarse al servidor y cargar la página web solicitada.

1. **Descripción de la técnica de ataque seleccionada**

Tal como detalla (Masas, 2023) en el sitio web imperva a thales company, define 3 puntos importantes:

1. **Suplantación de DNS**

* **¿Qué es la suplantación de identidad (DNS)?**

El ataque conocido como suplantación o envenenamiento de caché DNS consiste en alterar los registros del sistema de nombres de dominio (DNS) con el objetivo de redirigir a los usuarios a un sitio web falso y malintencionado, el cual puede imitar al sitio legítimo que intentaban visitar.

Cuando los usuarios acceden a la página fraudulenta, pueden ser inducidos a ingresar sus credenciales de acceso, creyendo que están en su cuenta real. Esto permite a los atacantes obtener datos sensibles. Además, el sitio falso puede infectar el dispositivo del usuario con virus o gusanos sin su conocimiento, otorgando al atacante un acceso persistente a la información y al equipo comprometido.

* **¿Cómo funciona la suplantación de DNS?**

La suplantación de DNS se basa en la explotación de vulnerabilidades en el sistema de nombres de dominio (DNS) y sus protocolos asociados. Este tipo de ataque puede ejecutarse de diversas formas:

* Un atacante puede emplear el protocolo de resolución de direcciones (ARP) para interceptar el tráfico del enrutador y manipular los registros de resolución de nombres de dominio.
* También puede alterar los registros de un servidor DNS autorizado, redirigiendo a los usuarios hacia un sitio web fraudulento.
* Otra técnica consiste en atacar un servidor de nombres intermedio, aprovechando fallos en su sistema de almacenamiento en caché para llevar a cabo un ataque de intermediario (Man-in-the-Middle o MITM).
* **¿Cuáles son las posibles consecuencias de la suplantación de DNS?**

La manipulación de DNS posibilita que el atacante obtenga información confidencial de usuarios que no sospechan nada. Si el atacante logra infiltrarse en el dispositivo de un usuario mediante un gusano o virus, puede aprovechar la oportunidad para instalar ransomware. Al redirigir el tráfico de páginas web legítimas hacia un sitio falso, los atacantes pueden mejorar la autoridad y el posicionamiento en los motores de búsqueda de su sitio fraudulento, lo que facilita atraer más tráfico hacia ese dominio en el futuro.

* **¿Cómo funciona la comunicación DNS normal?**

En un proceso de comunicación DNS habitual, el dispositivo del usuario inicia una consulta al solicitar un sitio web. El solucionador DNS, que generalmente está gestionado por el proveedor de servicios de Internet (ISP) del usuario, recibe la consulta. Si el solucionador no tiene la dirección IP correspondiente guardada en su caché, reenvía la consulta a un servidor raíz. Este servidor raíz redirige al solucionador al servidor de dominio de nivel superior (TLD) adecuado (por ejemplo, .com, .org, .net). Luego, el servidor TLD guía al solucionador DNS hacia el servidor de nombres del dominio, que devuelve la dirección IP asociada. Finalmente, el solucionador proporciona esta dirección IP al dispositivo del usuario, permitiendo que se conecte al sitio web solicitado.

Este procedimiento es esencial para la navegación en Internet y facilita que los usuarios accedan a sitios web mediante nombres de dominio comprensibles en lugar de usar direcciones IP numéricas.

1. **Métodos de ataque de suplantación de DNS:**

* **El hombre en el medio (MITM):**

Este método implica interceptar las comunicaciones entre los usuarios y un servidor DNS para redirigirlas a una dirección IP distinta o maliciosa. Al colocarse entre el usuario y el servidor DNS, los atacantes pueden alterar las respuestas DNS y dirigir a los usuarios a destinos no deseados.

* **Compromiso del servidor DNS:**

En este método, los atacantes toman el control de un servidor DNS configurado para entregar una dirección IP maliciosa. Al comprometer dicho servidor, logran tomar el control del proceso de resolución de DNS y pueden modificar las respuestas para redirigir a los usuarios hacia sitios fraudulentos.

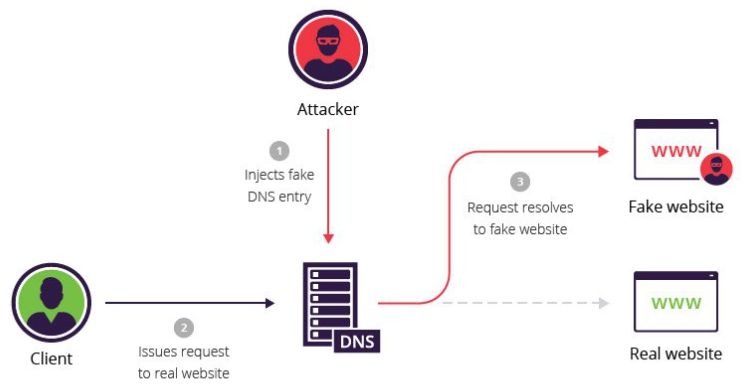
* **Aprovechamiento del tiempo de vida (TTL):**

Este método aprovecha los valores de tiempo de vida (TTL) en la caché del servidor DNS. Los atacantes alteran estos valores TTL para asegurar que las entradas DNS incorrectas permanezcan en la caché durante más tiempo. Esto aumenta el número de víctimas potenciales y extiende la duración del ataque. Incluso aquellos usuarios que no eran el objetivo inicial pueden ser redirigidos a sitios maliciosos si las respuestas DNS alteradas siguen presentes en la caché.

Cada uno de estos métodos representa una amenaza significativa para la integridad y seguridad de la resolución de DNS, lo que subraya la importancia de implementar medidas de defensa efectivas contra los ataques de suplantación de DNS.

(Masas, 2023) en la siguiente figura 5 nos muestra una representación de un ataque de envenenamiento de caché DNS (DNS poisoning), en el cual un atacante manipula las respuestas del servidor DNS para redirigir a las víctimas a un sitio web falso en lugar legítimo.

**Figura 5** *Ataque de compromiso del servidor DNS*



*Nota.* Aprovechamiento del tiempo de vida (TTL). <https://www.imperva.com/learn/application-security/dnsspoofing/#:~:text=Domain%20Name%20Server%20(DNS)%20spoofing,resemble%20the%20user's%20intended%20destination>.

1. **Ejemplo de envenenamiento de caché DNS**

El siguiente caso ejemplifica un ataque de envenenamiento de caché DNS. En este tipo de ataque, un atacante (IP 192.168.3.30) intercepta la comunicación entre un cliente (IP 192.168.1.100) y el servidor de la página web www.estores.com (IP 192.168.2.200).

Para ejecutar el ataque, el atacante emplea una herramienta, como arpspoof, con la finalidad de engañar al cliente, haciéndo creer que la dirección IP del servidor es 192.168.3.30. De manera simultánea, también manipula al servidor para que asuma que la IP del cliente es 192.168.3.30.

* El atacante emplea arpspoof para alterar la tabla ARP del servidor, haciendo que asocie la dirección MAC del atacante con la del cliente.
* Luego, utiliza nuevamente arpspoof para engañar al cliente, haciéndole creer que la máquina del atacante es el servidor legítimo.
* Al ejecutar el comando de Linux echo 1 > /proc/sys/net/ipv4/ip\_forward, el atacante permite que los paquetes IP entre el cliente y el servidor pasen a través de su equipo.
* Se configura un archivo host en la computadora del atacante, vinculando el dominio www.estores.com a una dirección IP local falsa (192.168.3.30).
* En la IP local del atacante, se levanta un servidor web falso con la apariencia del sitio original.
* Usando una herramienta como dnsspoof, todas las consultas DNS son redirigidas al archivo de host del atacante, mostrando a las víctimas un sitio web fraudulento que facilita la instalación de malware en sus dispositivos.

1. **Impacto de la técnica de ataque en el servicio de red**

Asaduzzaman et al. (2023) la suplantación de DNS es uno de los ataques más peligrosos, aprovechando la vulnerabilidad en la detección de conflictos de direcciones IP en servidores DHCP. Aunque algunos sistemas operativos modernos protegen estos ataques, la suplantación de DNS mediante envenenamiento de caché es difícil, ya que los servidores DNS públicos están altamente protegidos. En este estudio, se propone un enfoque innovador de ataque ético dentro de una red local (LAN), en el que un atacante introduce un servidor DHCP no autorizado y redirige el tráfico del usuario a una puerta de enlace maliciosa. Luego, mediante un motor de suplantación de DNS basado en Python (scapy), el atacante altera las respuestas DNS para redirigir al usuario a un sitio web de phishing. Este ataque se valida usando Kali Linux y EVE-NG.

IONOS Digital Guide (2023), nos dice que el DNS es una tecnología fundamental en internet: en todos los establecimientos de conexión, se recurre a la resolución de nombres. Por este motivo, el **DNS *spoofing* puede afectar a todas y cada una de las conexiones del cliente**. Da igual si la víctima accede a una página web o envía un correo electrónico: si la dirección IP del servidor afectado ha sido manipulada, el atacante puede acceder a los datos.

Estos son los riesgos que entraña el *DNS* *spoofing*:

* **Robo de datos confidenciales:** A través de ataques como spear phishing y pharming, los ciberdelincuentes sustraen información sensible, como contraseñas, que suelen emplearse para acceder a sistemas informáticos o realizar fraudes.
* **Infección del sistema con *malware***: La víctima, sin darse cuenta, instala un software malicioso en su sistema, lo que facilita nuevos ataques y permite al atacante llevar a cabo actividades de espionaje.
* **Intercepción de un perfil de usuario completo:** Los datos personales interceptados pueden ser vendidos o utilizados para llevar a cabo nuevos ataques de spear phishing.
* **Riesgo de amenaza persistente:** Si un servidor DNS malicioso se instala en el sistema, la comunicación queda vulnerable. Además, las respuestas DNS falsificadas pueden permanecer en la caché temporalmente, causando daños durante un periodo prolongado.

IONOS Digital Guide (2023), durante la pandemia de COVID-19 en 2020, se registró una ola de ataques de DNS spoofing mediante hijacking de rúter. Los atacantes configuraban direcciones DNS maliciosas en rúters vulnerables y mostraban falsos avisos de la OMS, instando a las víctimas a instalar una supuesta aplicación informativa. En realidad, era un troyano que recopilaba datos sensibles del sistema para realizar futuros ataques de spear phishing. Entre los datos robados, figuraban los siguientes:

* *Cookies* (navegador)
* Historial del navegador
* Datos de pago (navegador)
* Datos de acceso guardados (navegador)
* Datos de formulario guardados (navegador)
* *Wallets* de criptomonedas
* Todo tipo de archivos de texto en el dispositivo
* Bases de datos para autenticación de dos factores (2FA)

En resumen, los ciberdelincuentes emplean el DNS spoofing para llevar a cabo ataques de phishing y pharming, con el objetivo de robar información sensible del usuario. En el DNS spoofing, se engaña a la víctima para que crea que está accediendo a un dominio legítimo. Aprovechando su confianza en el sitio falsificado, se le induce a instalar un software malicioso, lo que provoca que, sin querer, infecte su propio sistema.

1. **Descripción de las técnicas de mitigación y contramedidas**

(Barreto et al., 2023a) en su estudio, realizado en instituciones de educación superior (IES) en Brasil, analiza la seguridad en estas organizaciones en relación con protocolos web. Su objetivo es investigar y evaluar el estado de los servicios de seguridad web en las IES brasileñas, enfocándose en la implementación del Protocolo Seguro de Transferencia de Hipertexto (HTTPS) y las Extensiones de Seguridad del Sistema de Nombres de Dominio (DNSSEC), con el propósito de determinar su nivel de protección frente a ciberataques. Desde una metodología de enfoque cuantitativo y alcance descriptivo, se analizó la adopción de estas tecnologías de seguridad en servidores de instituciones educativas. Los resultados muestran que el 15% de las IES no utiliza certificados SSL/TLS, y entre las que sí lo admiten, un 14% no exige su uso. En cuanto a DNSSEC, solo el 2% de las instituciones han implementado este protocolo. Esto evidencia un bajo nivel de adopción de medidas de seguridad web, lo que podría exponer a estas entidades a riesgos significativos de ciberataques. Como conclusión, el estudio resalta la necesidad de que las IES desarrollen un plan de acción efectivo y continuo para mejorar su seguridad cibernética. La implementación de versiones actualizadas de estos protocolos es clave para mitigar riesgos y garantizar la protección de datos sensibles. Asimismo, se enfatiza la importancia de fortalecer la conciencia y el compromiso en la aplicación de estándares de seguridad web en el ámbito educativo.

(Hemanthindra et al., 2020) el sistema de nombres de dominio (DNS) es un componente esencial en la era tecnológica actual, ya que facilita las comunicaciones a través de Internet y redes. Sin embargo, su diseño original no ofrece suficiente seguridad para proteger la información que se maneja en la actualidad. Debido a su amplia aceptación y uso global, garantizar la seguridad del DNS sin afectar su funcionamiento se ha convertido en una necesidad. El protocolo DNSSEC es una solución viable para proteger la integridad de los datos y prevenir modificaciones malintencionadas. No obstante, la adopción de DNSSEC ha sido limitada, principalmente debido a la complejidad de su implementación, lo que ha desalentado a muchas organizaciones a utilizarlo. Para abordar este problema, el estudio propone un servicio de firma DNSSEC, donde los clientes se registran con un proveedor de servicios que implementa un entorno de firma que incluye un firmante DNSSEC, una base de datos y servicios web. Esto permite a los clientes gestionar sus zonas mediante una interfaz gráfica y automatizar el proceso de firma con un simple clic. La solución propuesta utiliza herramientas de código abierto y contenedores Linux para optimizar el uso de recursos. Además, las zonas firmadas se envían de forma segura a los servidores DNS del cliente mediante Transaction SIGnature (TSIG), asegurando la autenticidad de las solicitudes DNS. Con este enfoque, se elimina la complejidad técnica y el mantenimiento del sistema para el cliente, ya que el proveedor de servicios se encarga de la gestión, facilitando así la implementación de DNSSEC y mejorando la seguridad del DNS.

Nosyk et al. (2023) las extensiones de seguridad de DNS (DNSSEC) son una de las soluciones más eficaces para prevenir ataques de envenenamiento de caché de DNS. Sin embargo, la adopción de la validación DNSSEC es baja, ya que muy pocos solucionadores de DNS llevan a cabo este proceso. Identificar estos sistemas representa un desafío, ya que los métodos existentes no permiten realizar mediciones a escala global. Para abordar esta problemática, el estudio propone una nueva técnica remota que permite identificar solucionadores que validan DNSSEC. El método consta de dos pasos. Primero, se identifican solucionadores abiertos mediante el escaneo de 3.1 mil millones de hosts y se les solicita resolver un dominio correctamente configurado junto con siete dominios mal configurados intencionalmente. A partir de los patrones de consulta y códigos de respuesta, se construye un clasificador que distingue entre solucionadores que validan y aquellos que no lo hacen. Los resultados muestran que, aunque la mayoría de los solucionadores abiertos están habilitados para DNSSEC, solo el 18 % en IPv4 y el 38 % en IPv6 validan las respuestas que reciben. En el segundo paso, se identifican de forma remota solucionadores cerrados en redes sin validación de direcciones de origen (SAV) entrante. Utilizando el clasificador previamente creado, se detectó que el 37,4 % de los solucionadores cerrados de IPv4 y el 42,9 % de IPv6 validan DNSSEC. Además, los resultados fueron validados a través de sondas RIPE Atlas. Finalmente, el estudio demuestra que los solucionadores identificados como validadores y no validadores envían activamente consultas a los servidores raíz de DNS, lo que confirma que se trata de resolutores recursivos operativos y no de sistemas mal configurados.

Fukuda et al. (2020) DNSSEC es una técnica prometedora para mejorar la seguridad del DNS, ya que permite la autenticación de una cadena de confianza dentro de su jerarquía. Sin embargo, su implementación sigue siendo un desafío. Uno de los principales problemas es la falta de confiabilidad en la validación de DNSSEC en entornos operativos reales, especialmente debido a errores en su ejecución. Por ello, los operadores de DNS requieren métodos eficientes y confiables para detectar fallas en la validación de DNSSEC. La práctica actual consiste en validar periódicamente todas las zonas DNSSEC disponibles, pero este método presenta problemas de escalabilidad. Para solucionar esto, el estudio propone un enfoque alternativo basado en el análisis pasivo de consultas DNS recopiladas en servidores autoritativos a nivel de TLD, en lugar de realizar validaciones activas en todas las zonas. El método se enfoca en analizar los cambios en los patrones de consulta en servidores autoritativos antes y después de que ocurran fallas en la validación de DNSSEC. Para demostrar la validez de este enfoque, se realizaron mediciones activas y pasivas a gran escala utilizando sondas RIPE Atlas y un servidor autoritativo dedicado. Los resultados muestran que un aumento en las consultas DNSKEY es un indicador prometedor para detectar fallas de validación mediante el análisis de datos recopilados de forma pasiva. Además, se observó que un incremento en la cantidad de consultas es limitado incluso con valores TTL cortos en las consultas DNSSEC, lo que sugiere que una configuración con TTL reducido puede ayudar a mitigar el impacto del almacenamiento en caché de fallas de validación DNSSEC.

Abirami & Naresh (2024) el envenenamiento de la caché de DNS, también conocido como suplantación de DNS, es una amenaza significativa para la ciberseguridad, ya que permite a los atacantes manipular las cachés de DNS para redirigir a los usuarios a sitios web maliciosos. Este estudio analiza en profundidad los mecanismos de este tipo de ataque, identificando vulnerabilidades en la infraestructura de DNS y explorando los diferentes vectores utilizados por actores maliciosos. A partir de una revisión exhaustiva de la literatura sobre ataques de envenenamiento de caché de DNS y sus estrategias de mitigación, el estudio identifica brechas en la seguridad actual y oportunidades para futuras investigaciones. Como parte de la solución, se propone una estrategia basada en el uso de DNSSEC y DNS over TLS (DoT) para fortalecer la seguridad del DNS. La propuesta se centra en la implementación de estas tecnologías dentro del firmware del enrutador OpenWrt. A través de esta implementación, se demuestra cómo DNSSEC y DoT pueden mejorar la capacidad de los solucionadores DNS para resistir ataques de envenenamiento de caché, contribuyendo así a reforzar la seguridad general de Internet.

Khan & Goodwin (2024) DNSSEC es una función de seguridad del sistema DNS que utiliza autenticación criptográfica para garantizar que los registros DNS provengan de un servidor autorizado y no sean alterados. Aunque no protege la privacidad de las búsquedas, evita que atacantes manipulen las respuestas DNS y redirijan a los usuarios a sitios falsos.

La mitigación del **DNS Spoofing** se basa en la adopción de medidas de autenticación y validación en las respuestas DNS. La principal herramienta para lograrlo es **DNSSEC (DNS Security Extensions)**.

DNSSEC y la seguridad del DNS son conceptos relacionados, pero con enfoques diferentes. DNSSEC se centra en reforzar la seguridad del Sistema de Nombres de Dominio mediante extensiones criptográficas para asegurar la integridad y autenticidad de los registros DNS. Por otro lado, la seguridad del DNS es un concepto más amplio que incluye diversas medidas para proteger todo el entorno DNS, no solo los protocolos de DNSSEC, y abarca amenazas como los ataques DDoS, ofreciendo una estrategia integral de protección.

Barreto et al. (2023) los ciberataques afectan a todas las organizaciones, incluidas las instituciones de educación superior (IES), y pueden interrumpir su funcionamiento o provocar la fuga de datos sensibles. Por lo tanto, es crucial adoptar servicios de seguridad actualizados para los dispositivos y servicios expuestos a Internet, utilizando las versiones más recientes y seguras de los protocolos web y cumpliendo con las recomendaciones de seguridad. Este artículo analiza la implementación de protocolos de seguridad web, como HTTPS y DNSSEC, en las IES brasileñas. Los resultados muestran que el 15% de las IES no utilizan certificados SSL/TLS y el 14% de las que los tienen no requieren su uso. En cuanto a DNSSEC, solo el 2% de las IES lo implementan. Estos hallazgos subrayan la necesidad de diseñar un plan de acción continuo para mejorar la protección contra ciberataques. Por ello plantea algunas validaciones para proteger respuestas DNS mediante NDSSEC.

En el presente artículo de Microsoft Learn (2013), aborda la validación y protección de respuestas DNS mediante DNSSEC en Windows Server:

1. **Validar y proteger respuestas DNS mediante DNSSEC**

Los registros de recursos DNSSEC se utilizan para verificar y asegurar las respuestas DNS. La protección de las zonas DNS con DNSSEC se logra mediante el firmado de zona. Este proceso de firmado añade capacidad de validación sin alterar el mecanismo básico de consulta y respuesta DNS. Para obtener una introducción sobre DNSSEC en el servidor DNS de Windows Server, consulte la sección sobre Información general de DNSSEC.

Las firmas digitales se incorporan a las respuestas DNS para ofrecer validación. Estas firmas se encuentran en los registros de recursos asociados con DNSSEC, los cuales se generan y se añaden a la zona durante el proceso de firmado de zona.

* **DNS recursivo**

Cuando un servidor DNS recursivo o de reenvío compatible con DNSSEC recibe una consulta de un cliente para una zona firmada con DNSSEC, solicita al servidor DNS autoritativo que envíe también los registros DNSSEC para validar la respuesta. Un servidor DNS recursivo o de reenvío identifica que la zona es compatible con DNSSEC si contiene un registro DNSKEY, también conocido como anclaje de veracidad, para esa zona.

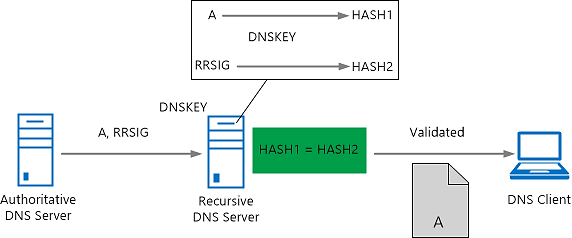
Consideremos que un servidor DNS no autoritativo puede utilizar la recursividad o el reenvío para resolver una consulta DNS. En este contexto, se hace referencia al servidor no autoritativo como un servidor DNS recursivo. Si el servidor opta por hacer un reenvío, el proceso de validación de DNSSEC para las respuestas DNS sigue siendo el mismo.

* **DNSKEY**

Un servidor DNS recursivo utiliza el registro DNSKEY para validar las respuestas de un servidor autoritativo. Para ello, descifra las firmas digitales en los registros relacionados con DNSSEC y compara los valores hash. Si los valores coinciden, proporciona la respuesta al cliente DNS; si no, envía un mensaje de error (SERVFAIL). Así, un servidor DNS con capacidad DNSSEC y un anclaje de veracidad válido protege contra ataques de suplantación de DNS, independientemente de que los clientes sean compatibles con DNSSEC. Si el cliente es compatible, puede configurarse para exigir que el servidor realice la validación DNSSEC.

Microsoft Learn (2013) también ilustra el proceso de validación de respuestas DNSSEC en un entorno de resolución DNS como se ve se muestra en la figura 6

**Figura 6** *Proceso de validación.*



*Nota.* Proceso de validación de respuestas DNS utilizando DNSSEC (Domain Name System Security Extensions). <https://learn.microsoft.com/es-es/windows-server/networking/dns/validate-dnssec-responses>

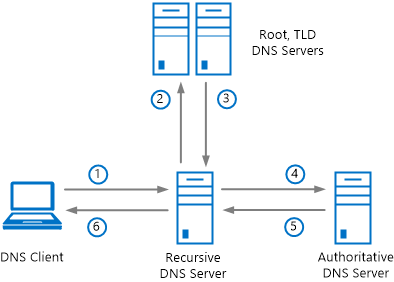
Los elementos DNSKEY se utilizan para calcular valores hash y descifrar registros RRSIG. En el proceso de validación no se muestran todos los pasos, ya que también se valida que los elementos DNSKEY sean válidos y, si existen, que los registros DS (firmante por delegación) también lo sean.

* **Proceso de consulta y respuesta de DNS**

En este ejemplo, se muestra cómo DNSSEC se integra en el proceso de consulta y respuesta de DNS para validar las respuestas. Un cliente DNS consulta un servidor recursivo, que a su vez consulta servidores autoritativos antes de devolver la respuesta. Se asume que los datos DNS no están en caché en el cliente ni en el servidor. La validación de las respuestas DNS a través de DNSSEC solo ocurre cuando la zona está firmada y se utilizan clientes y servidores compatibles con DNSSEC.

(Microsoft Learn, 2013) tambien ilustra una representación del proceso de resolución de nombres de dominio en DNS a través de un servidor recursivo, como se muestra en la figura 7.

**Figura 7** *Proceso de consultas de DNS recursivas*



*Nota.* proceso de resolución de nombres en el Sistema de Nombres de Dominio (DNS) utilizando un servidor recursivo. <https://learn.microsoft.com/es-es/windows-server/networking/dns/validate-dnssec-responses>

En la tabla 1 se muestran los pasos en una consulta y respuesta DNS con datos de DNSSEC opcionales.

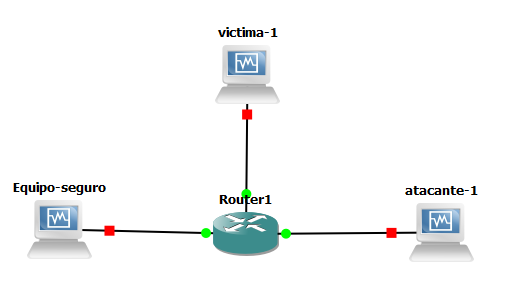
**Tabla 1** *Consulta y respuesta DNS con datos de DNSSEC*

| **Step** | **Consulta-respuesta** | **Datos de DNSSEC opcionales** |
| --- | --- | --- |
| 1 | Un cliente DNS envía una consulta DNS a un servidor DNS recursivo. | El cliente DNS puede indicar que es compatible con DNSSEC (DO=1). |
| 2 | El servidor DNS recursivo envía una consulta DNS a los servidores DNS raíz y de dominio de primer nivel (TLD). | El servidor DNS recursivo puede indicar que es compatible con DNSSEC (DO=1). |
| 3 | Los servidores raíz y de TLD devuelven una respuesta DNS al servidor DNS recursivo y proporcionan la dirección IP del servidor DNS autoritativo de la zona. | Los servidores autoritativos de la zona primaria pueden indicar que la zona secundaria está firmada mediante DNSSEC e incluye una delegación de protección (registro DS). |
| 4 | El servidor DNS recursivo envía una consulta DNS al servidor DNS autoritativo de la zona. | El servidor DNS recursivo puede indicar que es compatible con DNSSEC (DO=1) y es capaz de validar los registros de recursos firmados (CD=1) que se van a enviar en la respuesta. |
| 5 | El servidor DNS autoritativo devuelve una respuesta DNS al servidor DNS recursivo y proporciona los datos de registro de recursos. | El servidor DNS autoritativo puede incluir firmas DNSSEC en forma de registros RRSIG en la respuesta DNS para su uso en la validación. |
| 6 | El servidor DNS recursivo devuelve una respuesta DNS al cliente DNS y proporciona los datos de registro de recursos. | El servidor DNS recursivo puede indicar si se validó la respuesta DNS (AD=1) mediante DNSSEC o no. |

1. **Diseño experimental**
2. **Descripción de escenarios de pruebas**

Los escenarios de prueba se consideran a partir de dos fases de la guía NIST SP 800-115, como son la planificación, ejecución y post-ejecución, aplicando pruebas de vulnerabilidades y penetración (pentesting) y el análisis aplicando técnicas o métodos de mitigación a vulnerabilidades para actuar sobre el DNS Spoofing. Para ambos contextos se maneja una estructura definida como se aprecia en la siguiente figura:

**Figura 8** *Escenario DNS Spoofing*



*Nota.* Elaboración propia

Para el presente escenario se tiene una red LAN, mediante una máquina de ataque (virtualizada y con servicio de Apache2 para servicio web) sobre dos máquinas cliente (también virtualizadas) sobre las cuales se aplica el ataque que consiste en primer lugar la captura de la dirección física y dirección IP del router mediante el ARP Spoofing y en segundo lugar redireccionar la puerta de enlace o gateway hacia la IP del equipo atacante que mediante DNS Spoofing, obteniendo y pudiendo manipular las peticiones y autenticación de los clientes. El procedimiento está abierto a múltiples riesgos como el phishing, distribución de malware, corrupción de servicios, robo de identidad o seguimiento y espionaje.

Por otro lado, se propone como alternativa de mitigación mediante comandos Linux: Iptable y Arptable. Con el procedimiento realizado se somete nuevamente a la evaluación y postevaluación (pentesting) con resultados positivos para la mitigación y protegiendo los datos confidenciales del o los usuarios que acceden a una página web bajo un servidor DNS de direcciones web.

Ambos procedimientos se llevan a cabo de manera controlada y supervisada por el equipo de trabajo, en un entorno virtualizado mediante el programa VirtualBox en una red LAN.

1. **Descripción de equipos utilizados**

Para lograr realizar la el ataque y mitigación, se proponen 2 escenarios:

* **Escenario de ataque al equipo victima**

Los equipos involucrados para la simulación del ataque se instalaron bajo sistemas operativos Linux, distros como Debian y Lubuntu, dicha especificación se aprecia en la tabla 2, en este contexto :

**Tabla 2***: Descripción de equipos utilizados para la experimentación*

| **N** | **Sistema Operativo** | **Ip** | **Rol** |
| --- | --- | --- | --- |
| 1 | Lubuntu | 192.168.1.15 | Equipo víctima |
| 2 | 192.168.1.8 | Equipo seguro |
| 3 | Debian | 192.168.1.14 | Equipo atacante |

*Nota.* Elaboración propia

* **Escenario de ataque al equipo seguro**

Para la simulación de este escenario en el presente informe se usarán sistemas operativos Linux, distros como Kali y Linux Mint, dicha especificación se aprecia en la tabla 3, en este contexto :

**Tabla 3***: Descripción de equipos utilizados para la experimentación*

| **N** | **Sistema Operativo** | **Ip** | **Rol** |
| --- | --- | --- | --- |
| 1 | Linux Mint | 192.168.1.65 | Equipo víctima |
| 2 | 192.168.1.66 | Equipo seguro |
| 3 | Kali | 192.168.1.59 | Equipo atacante |

*Nota.* Elaboración propia

Los equipos cliente (víctimas) consumen los servicios del servidor Web cuya dirección IP se traduce a un dominio mediante un servidor DNS que direcciona mediante la dirección web asociada, esta descripción es la acción cotidiana, normal y común para un entendido básico en redes de computadores. Por otro lado un servidor DNS se encarga de traducir los nombres de dominio (*Ejemplo:* [*www.google.com*](http://www.google.com)) en direcciones IP (*Ejemplo: 142.250.190.46*) (Carrasco, 2025).

El servidor web (para fines del ataque, será instalado en el equipo atacante) recibe las peticiones de los clientes que acceden mediante un navegador web mediante el protocolo HTTPS o HTTP y responde devolviendo recursos que van desde archivos HTML hasta enriquecidos con datos dinámicos (Berenguel, 2016).

Por otro lado, un cliente atacante resulta ser un equipo configurado que ejecuta instrucciones de comando que permiten realizar ataques o pruebas de vulnerabilidad (Stuttard & Pinto, 2011).

1. **Descripción de aplicaciones utilizadas**

**Bettercap:** esuna herramienta especializada para redes de computadoras dedicada a la interceptación, inspección y manipulación del tráfico en redes LAN. Es utilizada en ataques Man-In-The-Middle con carácter ético; donde su potencia y bondad radica en una mayor velocidad, estabilidad y soporte para múltiples protocolos y equipos (Solivan, 2023). Será esta opción en el presente trabajo para la realización del ataque DNS Spoofing y ARP Spoofing.

Como detalle adicional se indica que, también es posible utilizar Ettercap el cual es utilizado para ataques Man-In-The-Middle con carácter ético sobre las redes locales. con diferencias en la velocidad, estabilidad y bajo soporte sobre protocolos y equipos (Majidha & Santhiyakumari, 2021). Sin embargo, Bettercarp es una mejora del Ettercap ya que puede interceptar y redirigir solicitudes DNS, incluso en redes cifradas. Su capacidad de trabajar con HTTPS y entornos inalámbricos lo hace superior a herramientas clásicas.

**Apache2:** Para levantar el servidor web en la máquina atacante, se hace uso del Apache2, que es un servidor multiplataforma de código abierto muy popular a nivel mundial. Su uso va desde alojar sitios web, aplicaciones y servicios (Apache Software Foundation, 2025).

Para la mitigación del DNS Spoofing se ha decidido realizar la configuración de comandos Linux Iptable y Arptable, de la cual hablaremos a continuación.

Iptable y Arptable son herramientas de líneas de comando Linux que en el caso de Iptable permite configurar reglas de filtrado y manipulación de paquetes en el firewall del sistema operativo; para ello utiliza el framework Netfilter dentro del kernel de Linux para administrar el tráfico de red. por otro lado Arptable es análogo a Iptable, con la diferencia que el filtrado y manipulación se da en las tramas ARP, mapeando direcciones IP a direcciones MAC en redes LAN. (Payne et al., 2019)

**Tabla N4** *Características de Iptable y Arptable*

| **Características** | **Iptables** | **Arptables** |
| --- | --- | --- |
| **Nivel de red** | capa 3 | capa 2 |
| **Propósito** | filtrado de paquetes, NAT, firewall | control de ARP, prevención de spoofing |
| **Protocolos** | TCP, UDP, ICMP | ARP |
| **Usos comunes** | seguridad de red y filtrado de tráfico | protección contra ataques ARP |

*Nota. Fuente:* [*www.Netfilter.org*](http://www.netfilter.org)

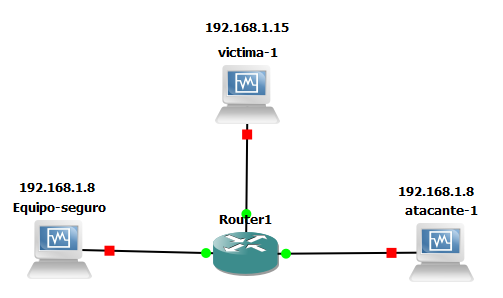
1. **Configuración de equipos y aplicaciones**

A continuación, se muestran las configuraciones a las máquinas detalladas en las tablas 2 y 3:

Equipos involucrados en la simulación de ataque sin mitigación

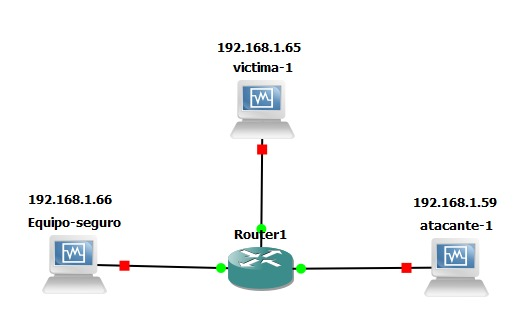
* Clientes en simulación del ataque: son dos máquinas virtualizadas con el sistema operativo Lubuntu y configurada con las IP estática 192.168.1.15 y 192.168.1.16 (para simular el ataque), máscara de red 255.255.255.0 y puerta de acceso 192.168.1.1.
* Atacante: es una máquina virtualizada con el sistema operativo Debian y configurada con una IP estática 192.168.1.14, máscara de red 255.255.255.0 y puerta de acceso 192.168.1.1, los servicios para realizar el ataque basado en la herramienta Bettercap para simular un escenario de pentesting (Man-in-the-Middle) para realizar el ataque ARP Spoofing y posterior DNS Spoofing y rastrear y ver paquetes y credenciales de los usuarios. Se precisa que esta máquina de ataque tiene el servicio Apache2, que permitirá la redirección de paquetes y credenciales para sí mismo desde las máquinas cliente.

Topología utilizada

****

*Nota.* Elaboración propia

* Clientes en simulación de la mitigación: son dos máquinas virtualizadas con el sistema operativo Linux Mint y configurada con las IP estática 192.168.1.65 y 192.168.1.66 (para simular el ataque), máscara de red 255.255.255.0 y puerta de acceso 192.168.1.1. Adicionalmente se instaló Arptables e Iptables en el equipo con ip 192.168.1.666
* Atacante: es una máquina virtualizada con el sistema operativo Kali y configurada con una IP estática 192.168.1.59, máscara de red 255.255.255.0 y puerta de acceso 192.168.1.1, también se usó la herramienta Bettercap para realizar el ataque ARP Spoofing y posterior DNS Spoofing que cuenta con el servicio Apache2, que permitirá la redirección de paquetes y credenciales para sí mismo desde las máquinas cliente (el cual no tendrá éxito en este escenario).

****

*Nota.* Elaboración propia

1. **Experimentación**
2. **Conducción de las pruebas de ataque aplicando NIST SP 800 – 115**

Para las pruebas de ataque o pentesting se toman las fases básicas para ataques DNS Spoofing, teniendo en cuenta la guía NIST SP 800 – 115, que sugiere: planificación, evaluación y post-evaluación, desarrollado a continuación:

* **Planificación**

**Objetivo**:

Realizar un ataque de **DNS Spoofing** previo ataque **ARP Spoofing**, con el propósito de interceptar y manipular las consultas DNS realizadas por un cliente al intentar acceder a un dominio web específico. Para ello, se llevó a cabo un ataque **Man-In-The-Middle (MITM)** utilizando la herramienta **Bettercap**, lo que permitió redirigir las solicitudes del dominio legítimo hacia una dirección IP controlada por el atacante.

La evaluación está sujeta a caracteres éticos puesto que no debe ser con fines maliciosos sino con fines académicos y de investigación, donde una vez concluido el procedimiento, se tiene el funcionamiento normal de los servicios web

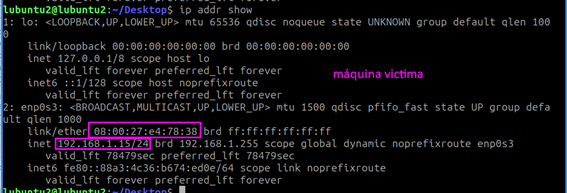
* **Ejecución**

Se revisan los documentos referentes a la configuración de IP en los equipos: atacante (Debian virtualizado) y clientes (Lubuntus virtualizados), configurando la conectividad a nivel de una LAN y el acceso a internet.

Se toma la vulnerabilidad en páginas conocidas ([www.facebook.com](http://www.facebook.com), [www.intranetdiresadmm.com](http://www.intranetdiresadmm.com))

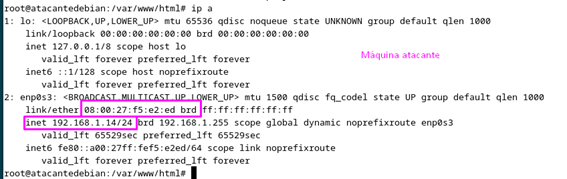
Se presenta a continuación los procedimientos para la configuración del ataque al DNS de la web definida:

**Figura 10** *Lubuntu maquina cliente 1 IP 192.168.1.15*



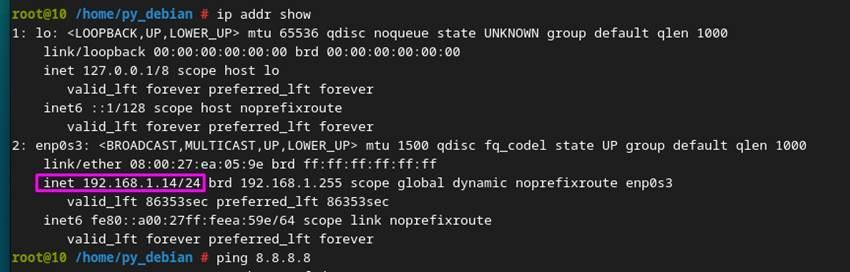
*Nota.* Elaboración propia

**Figura 11** *Debian máquina atacante y maquina s¿con servicio web mediante Apache2, IP 192.168.1.14*



*Nota.* Elaboración propia

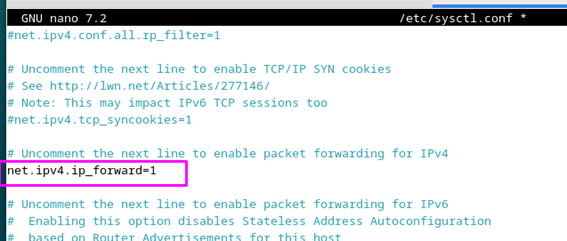
**Figura 12** *DEBIAN (atacante) IP: 192.168.1.14*



*Nota.* Elaboración propia

1. ***Configuraciones maquina atacante***
   * Activar modo routing mediante el comando sudo nano /etc/sysctl.conf
   * Habilitar el reenvío de IP: Para permitir que Debian reenvíe paquetes entre las máquinas, usamos este comando:

**Figura 13** *Activación del modo routing*



*Nota.* Elaboración propia

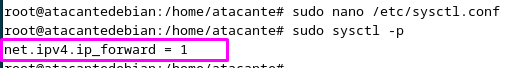
**Explicación**:

* **sysctl**: Es una herramienta para modificar parámetros del sistema.
* **net.IPv4.IP\_forward=1**: Habilita el reenvío de paquetes IP, permitiendo que DEBIAN actúe como un intermediario.

Guardamos y aplicamos los cambios, por lo que ejecutamos el comando sudo sysctl -p

**Figura 14**

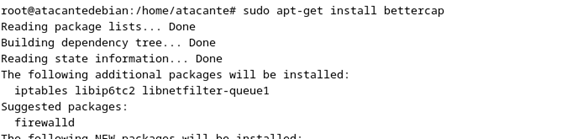
*Guardado y aplicación de los cambios*

**

*Nota.* Elaboración propia

En la máquina atacante instalamos la herramienta Bettercap

**Figura 15** *Instalación de Bettercap modo consola*



*Nota.* Elaboración propia

verificamos la versión y operatividad de la herramienta Bettercap

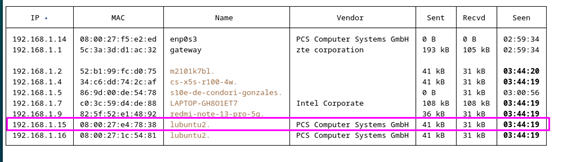
**Figura 16** *Comando para verificar el estado y versión de Bettercap*



*Nota.* Elaboración propia

Procedemos a realizar un escaneo de la red local, donde encontramos los datos de las dos máquinas cliente las cuales nos interesa para poder realizar en primer lugar un ARP Spoofing para tomar su MAC y gateway con la IP del equipo atacante.

**Figura 17** *Escaneo de la red*



*Nota.* Elaboración propia

Definimos como primera víctima al equipo de IP 192.168.1.15 y desde la máquina atacante realizamos la selección de IP objetivo y la activación del ARP Spoofing.

**Figura 18** *Objetivo IP y activación del ARP Spoofing*



*Nota.* Elaboración propia

Seguido definimos los dominios que interceptamos como parte del tráfico, en nuestro caso manejamos el dominio *monitoreo.intranetdiresamdd.com* para a continuación dirigir o asignar la IP de redirección a nuestro equipo atacante (*IP 192.168.1.14*) y posterior activación del DNS Spoofing.

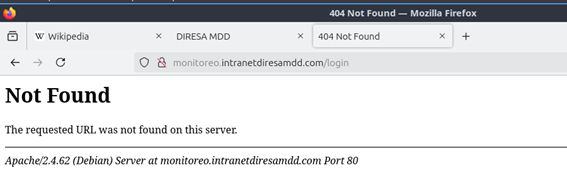
**Figura 19** *Definición de dominio, redirección de IP y activación del DNS Spoofing*



*Nota.* Elaboración propia

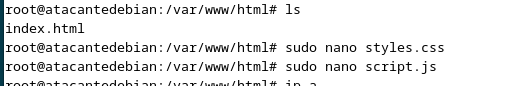
Notemos que al redireccionar la IP del sitio web original hacia la IP del atacante, no hay ninguna página definida que servirá como pantalla para que la víctima ingrese sus datos. Por lo que procedemos a su creación apoyándonos en el servicio web provisto por Apache2 en la máquina atacante.

**Figura 20** *Sitio web sin algún index.html*



Previamente instalamos el servicio Apache2 para servidores web, configurando y codificando sobre los archivos *index.html, styles.css* y *script.js*

**Figura 21** *Directorio de index.html*

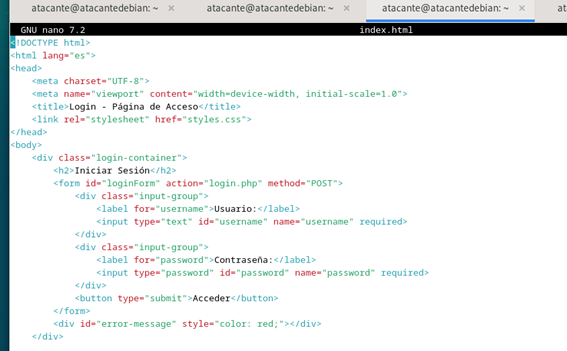


*Nota.* Elaboración propia

Codificamos sobre el archivo index.html como una web falsa que el cliente verá “similar”, no dándose cuenta que está siendo atacado.

**Figura 22**

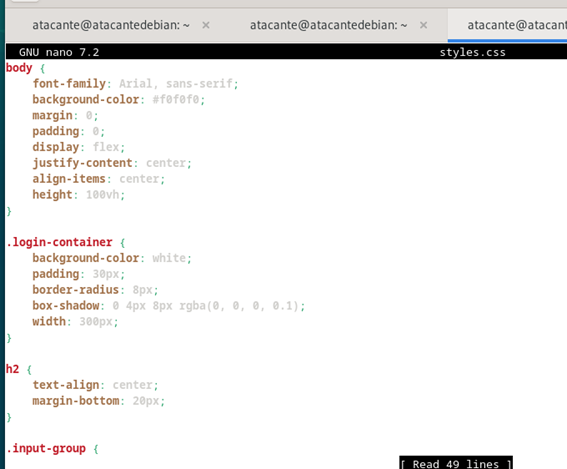
*Codificación de index.html*



*Nota.* Elaboración propia

Complementamos la edición del archivo styles.css para darle estilo a nuestra plantilla

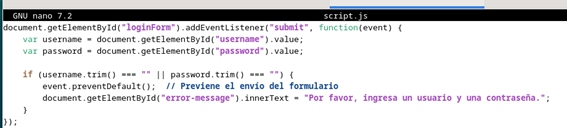
**Figura 23** *Edición y codificación del archivo styles.css*



*Nota.* Elaboración propia

Configuramos un archivo javascript para el manejo de credenciales de acceso mediante el método get.

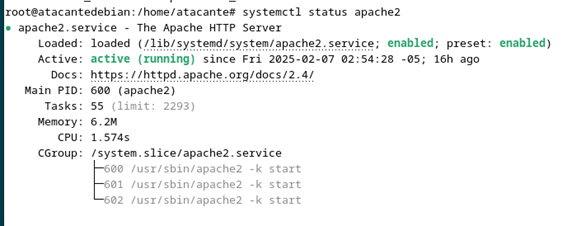
**Figura 24** *Edición y codificación sobre el archivo script.js*



*Nota.* Elaboración propia

Verificamos el estado del servicio Apache2

**Figura 25** *Verificación del estado del servicio Apache2*



*Nota.* Elaboración propia

Continuando el procedimiento, realizamos la definición de dominios objetivo, redirección hacia la IP atacante para posteriormente activar el DNS Spoofing.

**Figura 26** *Configuraciones de definiciones, objetivo y activación del DNS Spoofing*

*Nota.* Elaboración propia

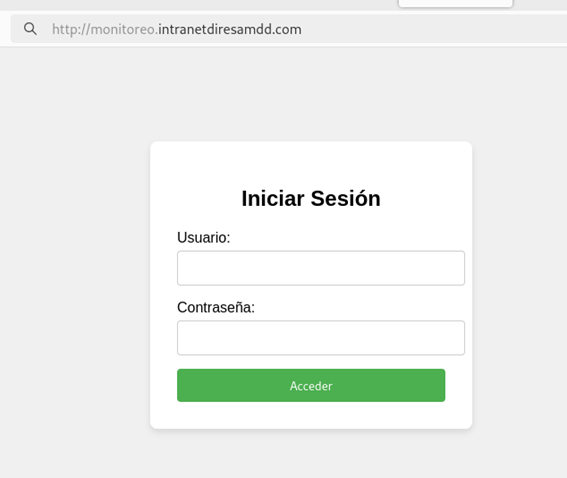
En un navegador del equipo cliente, al acceder al comunio pre definido como objetivo o target, tendrá por delante la plantilla preparada, que en un caso real, deberá ser similar a la interfaz original.

**Figura 27** *Dominio* [*monitoreo.diresamdd.com/login*](http://monitoreo.diresamdd.com/login) *antes del ataque*

**

*Nota.* Elaboración propia

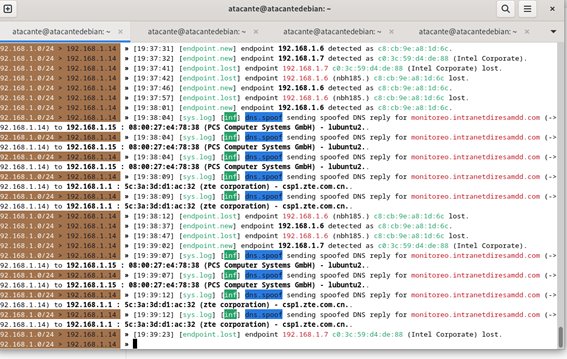
**Figura 28** *Dominio* [*monitoreo.diresamdd.com/login*](http://monitoreo.diresamdd.com/login) *después del ataque*

**

*Nota.* Elaboración propia

Verificamos dentro de la red la redirección de la puerta de enlace o gateway

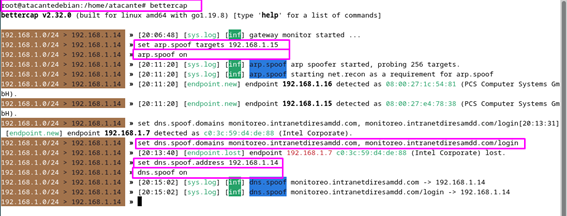
**Figura 29** *Captura del tráfico sobre la víctima*

**

*Nota.* Elaboración propia

Mostramos a continuación el resumen de comandos.

**Figura 30** *Resumen de comandos*

**

*Nota.* Elaboración propia

* **Post-ejecución**

Es evidente el resultado ‘positivo’ al ataque del servidor DNS, su secuestro y redireccionamiento, comprobando de esta manera la vulnerabilidad de seguridad en el servidor DNS mediante el ataque con la herramienta Bettercap *(Skoudis & Liston, 2005)*, modo de operación Man-In-The-Middle del DNS Spoofing (Radware, 2019).

Entre los hallazgos del ataque perpetrado, se presentan las figuras 28 y 29 de la fase de ejecución, que corresponde a la suplantación de las dominio y la captura del tráfico en la máquina cliente de IP 192.168.1.15, el redireccionamiento del sitio web hacia la IP de la máquina atacante de IP 192.168.1.14.

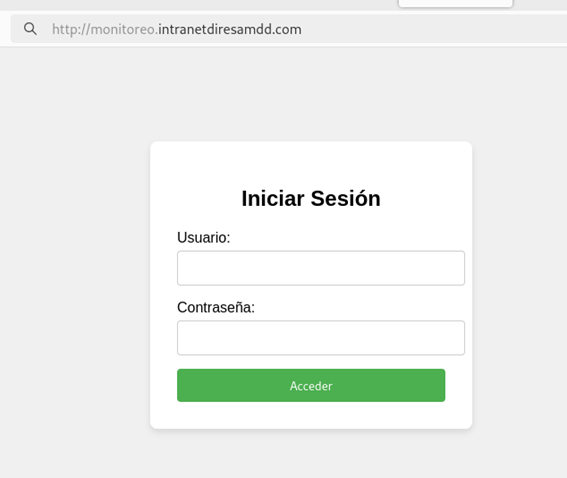
El equipo de trabajo al ver los resultados, propone en base a la documentación la implementación de la herramienta SNORT , desarrollada para un enfoque global de detección e intrusión en los diferentes recursos de una red computacional. (Fortinet, 2019)

1. **Documentación de resultados de los ataques**

Se presenta a continuación los resultados presentados a continuación:

* + Redireccionamiento del dominio

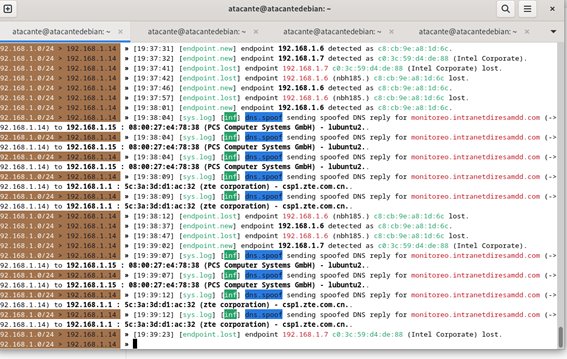
**Figura 40** *Redirección de dominio hacia la máquina atacante*

**

*Nota.* Elaboración propia

Redirección del tráfico en la máquina cliente

**Figura 41** *Redirección de tráfico de la máquina cliente*

**

*Nota.* Elaboración propia

1. **Implementación de contramedidas**

Como contramedida, se implementó la aplicación de comandos de iptables y arptables en un entorno Linux para mejorar la seguridad de la red y mitigar ataques de ARP Spoofing y DNS Spoofing (véase Figura 8).

La instalación de las herramientas necesarias se realizó mediante los siguientes comandos:

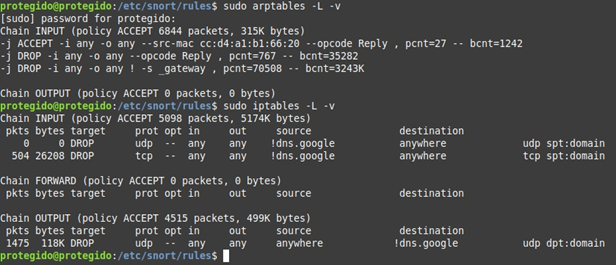
*apt install arptables*

*apt install iptables*

Posteriormente, se estableció las reglas de mitigación para ataques arp spoofing y ataques :

**Figura 42**

*Vista de configuración inicial de Iptables y Arptables*

**

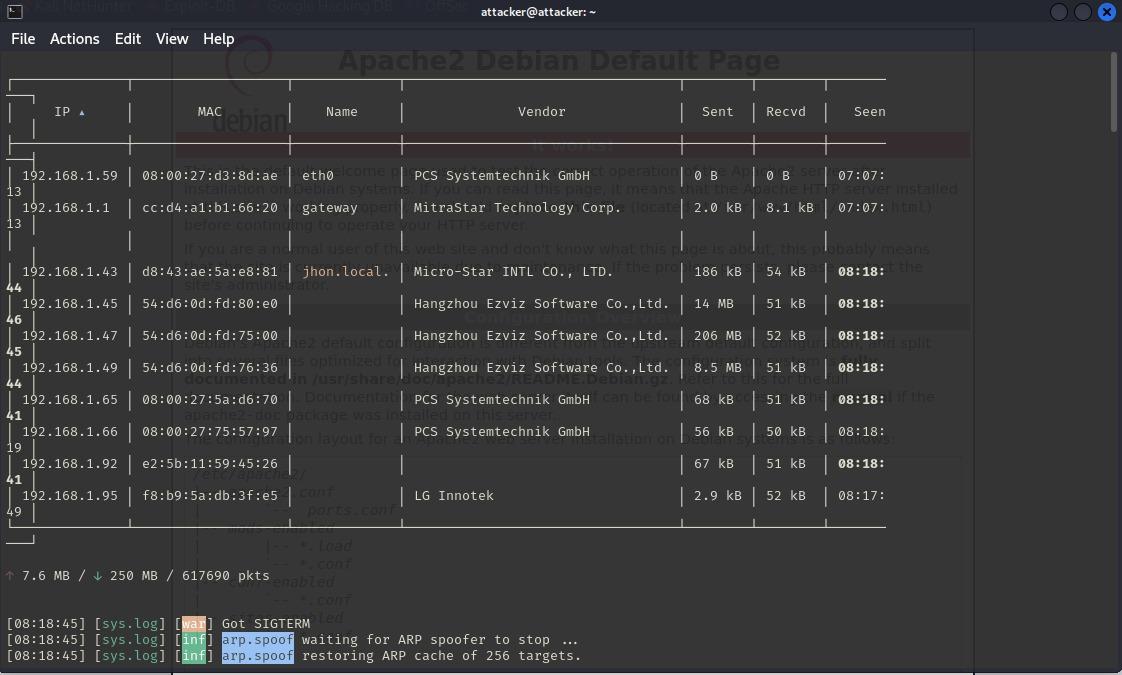
Nota. Elaboración propia

1. **Conducción de las pruebas de ataque post mitigación**

Se inicia nuevamente el ataque haciendo la búsqueda de equipos a ‘atacar’.

**Figura 43**

*Búsqueda de equipos para realizar ataque DNS Spoofing*



Nota. Elaboración propia

Se señala que las direcciones de equipos cliente se dan de la siguiente manera:

Tabla 00

Equipos cliente y sus IP asignadas

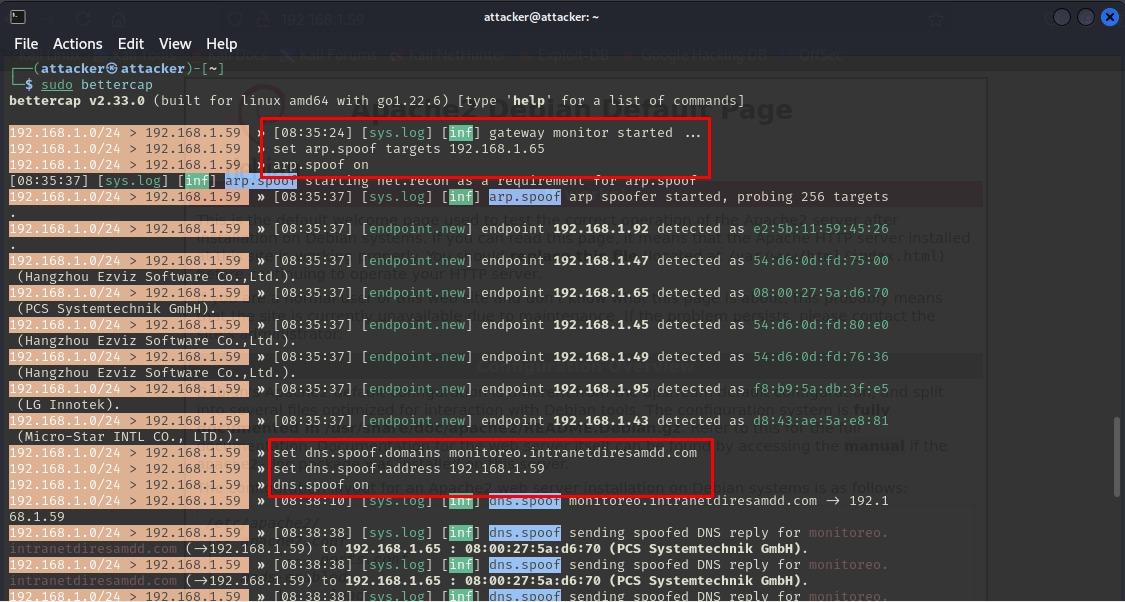
| **Cliente (con mitigación)** | **cliente (sin mitigación)** |
| --- | --- |
| 192.168.1.66 | 192.168.1.65 |

Nota. Elaboración propia

Procedemos al ataque sobre los equipos antes mencionados.

**Figura 44**

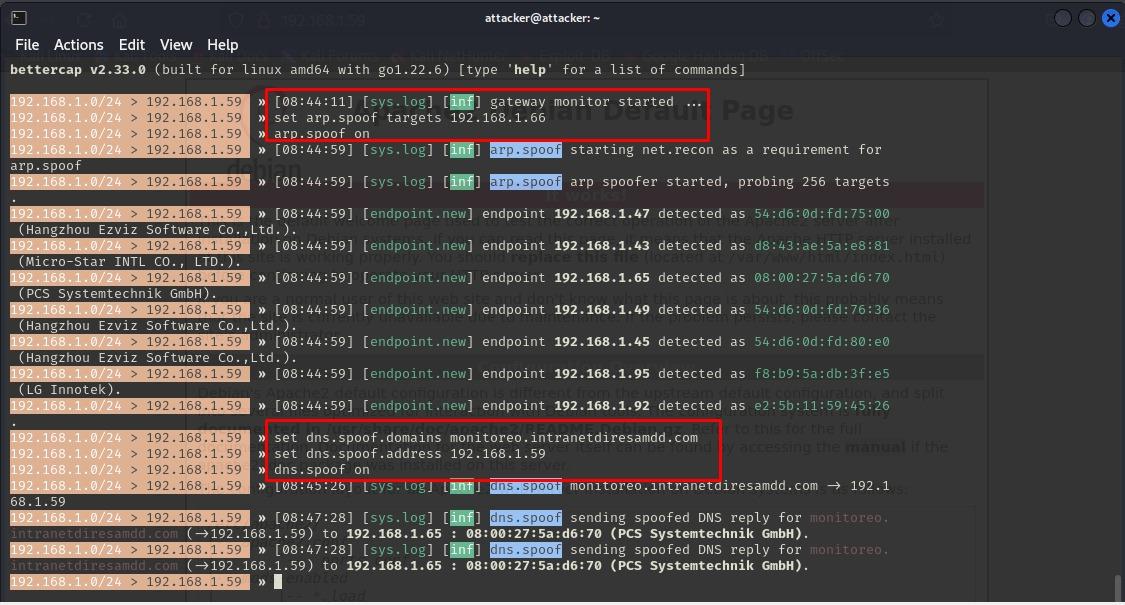
*Ataque DNS Spoofing sobre la máquina sin mitigación*



Nota. Elaboración propia

**Figura 45**

*Ataque DNS Spoofing sobre la máquina con mitigación*



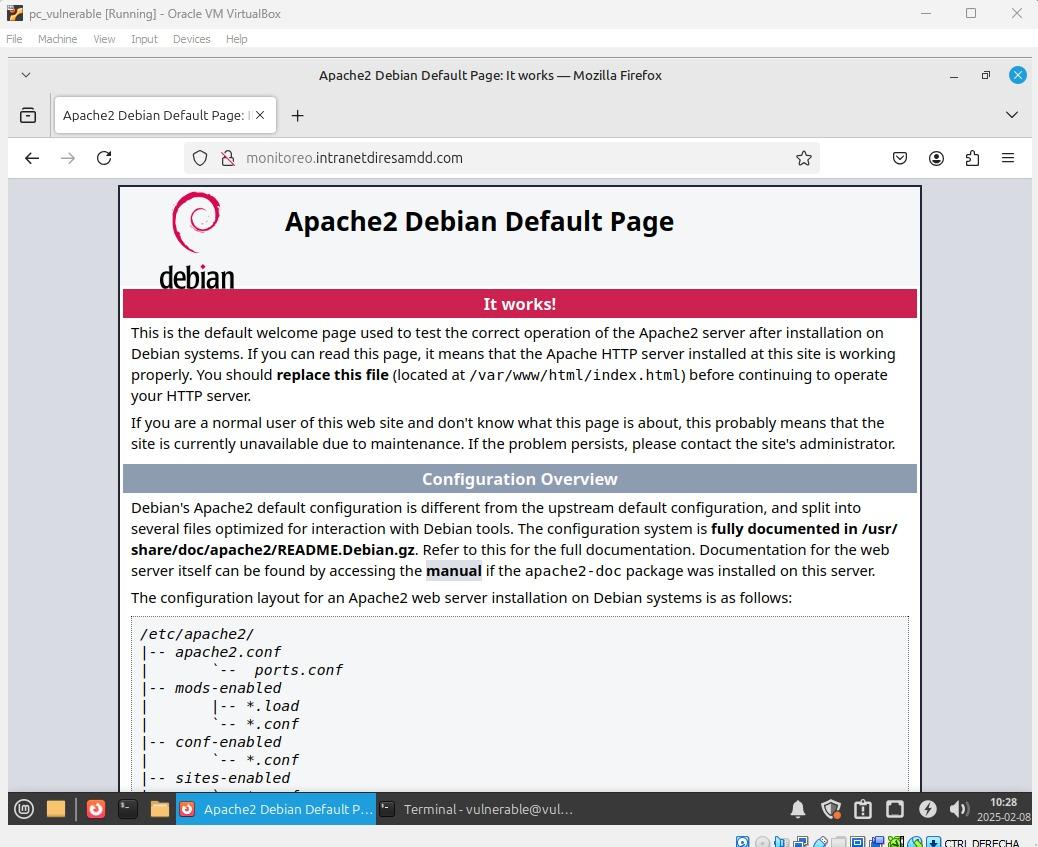
Nota. Elaboración propia

1. **Documentación de resultados de los ataques**

Los resultados pasada la experimentación son evidentes desde los navegadores de los equipos experimentales, teniendo los siguientes resultados:

**Figura 46**

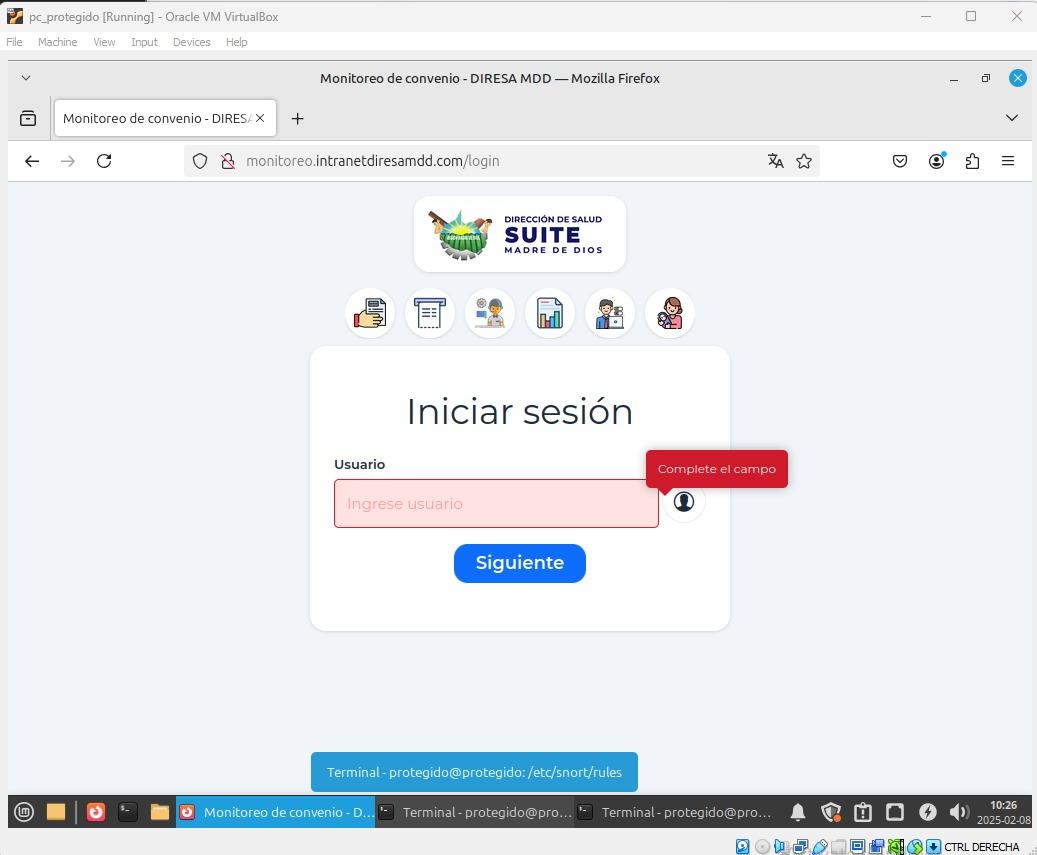
*Equipo sin mitigación*

**

Nota. Elaboración propia

**Figura 47**

*Equipo con mitigación*

**

Nota. Elaboración propia

1. **Discusión de resultados**

Los resultados obtenidos tras la aplicación de reglas en arptables e iptables demostraron una notable mejora en la seguridad del sistema de red, específicamente en la prevención de ataques de ARP Spoofing y DNS Spoofing. La comparación entre un entorno sin protección y otro con las reglas aplicadas permitió evaluar su efectividad en la prevención de suplantación de direcciones MAC (MITM) y en la protección de la resolución de nombres de dominio (DNS) contra respuestas falsificadas.

Las pruebas realizadas evidenciaron que la implementación de arptables bloqueó eficazmente respuestas ARP no autorizadas, impidiendo que un atacante manipulara la tabla ARP de la víctima y redirigiera su tráfico. Asimismo, la configuración de iptables restringe las respuestas DNS a únicamente aquellas provenientes de servidores DNS legítimos, mitigando ataques en los que se intentaba inyectar respuestas DNS falsificadas.

En conclusión, la implementación de reglas en arptables e iptables resultó efectiva para restringir la manipulación de direcciones ARP y evitar consultas DNS a servidores no confiables, fortaleciendo así la seguridad de la red contra ataques de suplantación de identidad y falsificación de registros DNS.

1. **Conclusión**

La operación conjunta de iptables y arptables demuestra ser una estrategia efectiva para prevenir ataques de DNS Spoofing, reduciendo la posibilidad de que un atacante suplante respuestas DNS falsas dentro de la red LAN; sin embargo, para una defensa integral, se recomienda complementarla con otras estrategias de seguridad, como el uso de DNS seguros y la implementación de autenticación en la infraestructura de red local y la protección global de todos los equipos de la red LAN, de tal modo que se pueda masificar esta mitigación.

1. **Referencias**

Abirami, S., & Naresh, R. (2024). DNS Enhancement with DNSSEC and DoT for Enhanced Online Security. *Proceedings of the 2nd IEEE International Conference on Networking and Communications 2024, ICNWC 2024*. https://doi.org/10.1109/ICNWC60771.2024.10537516

Apache Software Foundation. (2025). *Welcome to The Apache Software Foundation*. https://www.apache.org/

Arsys. (2024). *¿Qué es DNSSEC?* https://www.youtube.com/watch?v=gT8XfmnXqWw

Asaduzzaman, J., Muhammad Nazrul, I., & Sarker, I. H. (2023). Unveiling DNS Spoofing Vulnerabilities: An Ethical Examination Within Local Area Networks. *2023 26th International Conference on Computer and Information Technology, ICCIT 2023*. https://doi.org/10.1109/ICCIT60459.2023.10441649

Barreto, J., Almeida, H., & Pinto, P. (2023a). An Overview of HTTPS and DNSSEC Services Adoption in Higher Education Institutions in Brazil. *International Conference on Advanced Communication Technology, ICACT*, *2023-February*, 180–185. https://doi.org/10.23919/ICACT56868.2023.10079449

Barreto, J., Almeida, H., & Pinto, P. (2023b). An Overview of HTTPS and DNSSEC Services Adoption in Higher Education Institutions in Brazil. *International Conference on Advanced Communication Technology, ICACT*, *2023-February*, 180–185. https://doi.org/10.23919/ICACT56868.2023.10079449

Berenguel, J. (2016). *DESARROLLO DE APLICACIONES WEB EN EL ENTORNO SERVIDOR*. https://www.sancristoballibros.com/libro/desarrollo-de-aplicaciones-web-en-el-entorno-servidor\_57515

Carrasco, E. (2025). *Guía de laboratorio: configuración de un servidor DNS en Linux*. https://classroom.google.com/c/Njg3NjQ1MjE5Mzcx/m/NzQ3MDAwMzY3OTA5/details

Fukuda, K., Yoneya, Y., & Mitamura, T. (2020). Towards detecting DNSSEC validation failure with passive measurements. *Proceedings of IEEE/IFIP Network Operations and Management Symposium 2020: Management in the Age of Softwarization and Artificial Intelligence, NOMS 2020*. https://doi.org/10.1109/NOMS47738.2020.9110466

Hemanthindra, A., Phokeer, A., Ramsurrun, V., Katsina, P., Anantwar, S., & Seeam, A. K. (2020). DNSSEC as a service-A prototype implementation. *Proceedings of the International Conference on Microelectronics, ICM*, *2020-December*. https://doi.org/10.1109/ICM50269.2020.9331810

Henry, R. (2017). *Metodología de Pruebas de Intrusión en la NIST SP 800-115 | Behique Digital*. https://henryraul.wordpress.com/2017/05/10/metodologia-de-pruebas-de-intrusion-en-la-nist-sp-800-115/

IONOS Digital Guide. (2023). *¿Qué es DNS spoofing? - IONOS*. https://www.ionos.com/es-us/digitalguide/servidores/seguridad/dns-spoofing/

Khan, T., & Goodwin, M. (2024). *¿Qué es DNSSEC (extensiones de seguridad DNS)?* https://www.ibm.com/es-es/topics/dnssec

Majidha, K., & Santhiyakumari, N. (2021). *Sci-Hub | A Survey On Network Packet Inspection And ARP Poisoning Using Wireshark And Ettercap*. https://sci-hub.ru/10.1109/ICAIS50930.2021.9395852

Masas, R. (2023). *Suplantación de DNS*. https://www.imperva.com/learn/application-security/dns-spoofing/

Microsoft Learn. (2013). *Validar y proteger respuestas DNS mediante DNSSEC en el servidor DNS en Windows Server*. https://learn.microsoft.com/es-es/windows-server/networking/dns/validate-dnssec-responses

NIST. (2023). *National Institute of Standards and Technology*. https://www.nist.gov/

Nosyk, Y., Korczynski, M., & Duda, A. (2023). Guardians of DNS Integrity: A Remote Method for Identifying DNSSEC Validators Across the Internet. *Proceedings - 2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom/BigDataSE/CSE/EUC/ISCI 2023*, 1470–1479. https://doi.org/10.1109/TRUSTCOM60117.2023.00201

Quiroz, C., & Goodwin, M. (2024). *¿Qué es un servidor DNS?* https://www.ibm.com/es-es/topics/dns-server

Radware. (2019). *Servicios contra DDoS: productos y soluciones de seguridad en la nube*. https://es.radware.com/

Sánchez, M. (2015). *HACKING ETICO IMPACTO EN LA SOCIEDAD*.

Scarfone, K., Souppaya, M., Cody, A., & Orebaugh, A. (2008). *Special Publication 800-115 Technical Guide to Information Security Testing and Assessment Recommendations of the National Institute of Standards and Technology*. https://doi.org/10.6028/NIST.SP.800-115

Skoudis, E., & Liston, T. (2005). *Counter hack reloaded, second edition: a step-by-step guide to computer attacks and effective defenses, Second edition*. 784. https://books.google.com/books/about/Counter\_Hack\_Reloaded.html?id=nulSAAAAMAAJ

Solivan, R. (2023). *Network Security Assessment using Bettercap: DNS Spoofing and How to Mitigate*.

Stuttard, D., & Pinto, M. (2011). *The Web Application Hacker’s Handbook: Finding and Exploiting Security Flaws, 2nd Edition*. https://www.wiley.com/en-us/The+Web+Application+Hacker’s+Handbook%3A+Finding+and+Exploiting+Security+Flaws%2C+2nd+Edition-p-9781118026472