ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO DE UN TOOLKIT DE PRUEBAS DE INTRUSIÓN BASADO EN OSSTMM

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELÉCTRONICA Y REDES DE INFORMACIÓN

SHIRLA FRECY IZA SANHUEZA

shirla_16@hotmail.com

DIRECTOR: PhD. LUIS ENRIQUE MAFLA GALLEGOS

enrique.mafla@epn.edu.ec

Quito, Septiembre 2015

DECLARACIÓN

Yo, Shirla Frecy Iza Sanhueza, declaro bajo juramento que el trabajo aquí descrito es de mi auditoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

_____

Shirla Frecy Iza Sanhueza

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Shirla Frecy Iza Sanhueza, bajo mi supervisión.

_____

PhD. Enrique Mafla

DIRECTOR DEL PROYECTO

AGRADECIMIENTOS

Agradezco a mis padres quienes con su cariño, dedicación, afecto y confianza siempre me apoyaron en el camino emprendido, para alentarme en mi profesión y hacer de mí una persona de bien.

Gracias a mi hijito precioso Mati, quien con su risa, sus travesuras, sus ocurrencias llena mi vida de amor, cariño y me proporciona la fuerza suficiente para seguir con las metas en mi vida, y ver cada día como una nueva oportunidad de reír, amar y salir adelante. Te amo chiquito precioso.

Gracias a mis hermanos y familiares por su afecto, su paciencia, su amor y por siempre caminar junto a mí en este trayecto de mi vida, siendo un pilar fundamental para alcanzar todas mis metas.

De manera especial agradezco al Dr. Enrique Mafla quien con su guía, me ayudó a culminar esta etapa importante de mi vida.

DEDICATORIA

“Lo que la mente del hombre puede concebir y creer, es lo que la mente del hombre puede lograr”

Napoleon Hill

A mis papitos Dario y Fresia que con su amor, su paciencia, su constancia, me han enseñado a caminar en la vida, a luchar por los sueños y a no rendirse nunca. Gracias por su tenacidad y su lucha incansable y por todo lo que me han ayudado para que este sueño se convierta en realidad. Los amo.

A mi hijo Matí que con sus travesuras siempre me alegra la vida. Te amo gordito lindo, gracias por llenar mi vida de bendiciones y alegrías. Esto es por ti mi vida, y mucho más.

A mi hermano Dusannt, por tu cariño y amor incondicional. Gracias ñañito por todo tu apoyo, tu ejemplo, has sido una gran guía referencial en mi vida.

A mi hermana Leyli, por tu apoyo y cariño que siempre me has brindado. Te agradezco por guiarme en cada paso de mi vida y siempre velar por mi bienestar.

A mi hermano Arnolito, por tu amor, cariño y apoyo incondicional. Gracias por impulsarme a ser mejor cada día, tu presencia en mi vida es una bendición.

De manera especial, agradezco al Dr. Enrique Mafla, por sus consejos y su guía para culminar esta etapa importante de mi vida.

Shirla

CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

El presente proyecto de titulación propone un Toolkit de pruebas de intrusión sobre un perímetro de red a nivel de pequeñas empresas en nuestro país. Este proyecto permitirá entregar un documento público que detalle un procedimiento formal de pruebas de intrusión, de tal forma que cualquier profesional de la seguridad pueda tomar de referencia para evaluar la seguridad en las redes perimetrales de las pequeñas empresas. La documentación de este proyecto de titulación se encuentra dividida en 4 capítulos.

En el primer capítulo se define el planteamiento y solución del problema del proyecto actual y seleccionamos las metodologías y herramientas para el desarrollo del mismo. La metodología que se utiliza es Cisco Safe[1], sección "Módulo Internet de la Empresa", para el diseño e implementación del perímetro de red. Se escogió esta metodología porque se basa en principios de diseño de seguridad, con un enfoque modular para facilitar el crecimiento y escalabilidad de la red. Para la implementación del Toolkit de pruebas de intrusión se utiliza el manual formal de pruebas expuesto por la metodología OSSTMM V3 [2] (Manual de la Metodología abierta de testeo de seguridad) sección "Seguridad de las Comunicaciones en las Redes de Datos". Se escogió esta metodología por reunir procedimientos ordenados y estándar en auditorías de seguridad, que pueden ser aplicados por profesionales de la seguridad informática.

En el segundo capítulo se diseña e implementa el perímetro de la red de datos y el Toolkit de pruebas de intrusión, en base a un análisis previo de requerimientos. Se detallan los elementos y módulos necesarios, tanto de la metodología Cisco Safe

[1] Cisco Safe: Un modelo de seguridad para las redes de las empresas. Ofrece información sobre las mejores prácticas a las partes interesadas en el diseño e implementación de redes seguras. Fuente: (http: //www.cisco.com/web/ES/assets/docs/safe_wp1.pdf, Página. 1).

[2] OSSTMM: Manual de la Metodología Abierta de Testeo de Seguridad. Es un documento que reúne procedimientos ordenados y metodológicos de pruebas que debe realizar un profesional de la seguridad informática durante el desarrollo de auditorías y verificaciones de seguridad. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf).

como de la Metodología OSSTMM para cumplir con los requerimientos expuestos. Luego, se detalla el Toolkit formal de pruebas de intrusión exponiendo el procedimiento, la herramienta recomendada y el comando o módulo de cada herramienta que permitirá cumplir con la tarea solicitada.

En el tercer capítulo se documenta todas las pruebas y resultados obtenidos en el desarrollo del proyecto. Se incluye también el análisis de resultados que permitan al analista de seguridad determinar los falsos positivos, negativos y errores humanos en las pruebas descritas del presente proyecto. Finalmente en el cuarto capítulo se presenta las conclusiones y recomendaciones derivadas del desarrollo del proyecto de titulación.

PRESENTACIÓN

El presente proyecto de titulación tiene como objetivo desarrollar un Toolkit de pruebas de intrusión aplicado a una red perimetral de datos, a fin de obtener un documento público formal que esté a disposición de las pequeñas empresas para implementar auditorías de seguridad.

Para cumplir con este objetivo en primera instancia se diseña el perímetro de la red de datos basado en la metodología Safe de Cisco Módulo “Internet de la Empresa”, la cual plantea directrices de diseño para redes seguras. Posteriormente se aplica el Toolkit de pruebas de intrusión bajo los lineamientos de la metodología OSSTMM V3 sección “Canal de Redes de Datos”, quien recoge procedimientos ordenados y metódicos para el desarrollo de auditorías de seguridad.

Para el desarrollo del toolkit de pruebas de intrusión se utilizan las fases definidas por la propia metodología, comenzando por una etapa de inducción para definir los requisitos de la auditoría, el alcance y sus limitaciones. Posteriormente se desarrolla las fases de Interacción, Investigación e Intervención a fin de conocer la forma de trabajo de los sistemas e intervenir directamente con ellos.

Se desarrollarán los procedimientos formales, en algunos casos mediante mecanismos manuales y en la mayoría con el uso de herramientas/comandos para cumplir la sección de pruebas requerida. Posteriormente en el análisis de resultados se realizará un análisis detallado de los resultados esperados, en base a los resultados obtenidos a fin de constatar el alcance en las pruebas desarrolladas.

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS

1.1 INTRODUCCIÓN

En este capítulo se define el planteamiento del problema, se propone la solución al mismo, y se selecciona las metodologías y herramientas a ser utilizadas para el desarrollo del presente proyecto. Se define el problema en la sección 1.2 tomando en cuenta la integración de las funcionalidades para el diseño de redes seguras con la Metodología Cisco Safe [L1] y el documento formal de pruebas de intrusión con la Metodología OSSTMM [L2] (Manual de la Metodología Abierta de Testeo de Seguridad) versión 3. Posteriormente se propone una solución en la sección 1.3 que está acorde a los requerimientos planteados en el diseño del perímetro de red y del toolkit de pruebas de intrusión respectivamente. Finalmente, se detalla las herramientas que se utilizarán para el desarrollo del toolkit de pruebas de intrusión conforme a las diferentes fases definidas en la metodología OSSTMM.

1.2 PLANTEAMIENTO DEL PROBLEMA

En el país no existe un documento público de procedimientos formales de pruebas de intrusión aplicado a las redes perimetrales de las pequeñas empresas. Si bien es cierto, que existen metodologías de seguridad como Cisco Safe que basan sus lineamientos de diseño para implementar redes seguras. Por otro lado también existen metodologías para el desarrollo de auditorías estándar de seguridad como OSSTMM (Manual de la Metodología abierta de testeo de seguridad). Entonces, se hace necesario la combinación de ambas metodologías para desarrollar procedimientos públicos, que permitan a un analista de seguridad llevar a cabo pruebas de intrusión con procedimientos ordenados y estándar sobre redes perimetrales.

1.3 PROPUESTA DE SOLUCIÓN

La propuesta de solución del problema planteado, implica combinar ambas metodologías, Cisco Safe para el desarrollo de un perímetro de red seguro y OSSTMM para realizar el procedimiento formal de pruebas de intrusión sobre este perímetro y obtener un documento público formal que esté a disposición de las pequeñas empresas para implementar auditorías de seguridad.

Por tanto los objetivos del proyecto actual son:

- Entregar un documento formal de pruebas de intrusión en la red perimetral, de tal forma que un profesional de la seguridad pueda tomar de referencia para evaluar la seguridad de las redes perimetrales a nivel de pequeñas empresas.
- Determinar las vulnerabilidades y amenazas de la red perimetral de datos, para evaluar el nivel de seguridad y recomendar las medidas de corrección y precaución pertinentes.

1.4 METODOLOGÍA Y HERRAMIENTAS

Esta sección se divide en dos etapas, la selección de la metodología y de las herramientas a ser usadas para la implementación del proyecto actual.

1.4.1 METODOLOGÍA

Para la elaboración del presente proyecto se utilizará la metodología Cisco Safe descrita en la siguiente sección 1.4.1.1 para establecer el perímetro de red, y la metodología OSSTMM descrita en la sección 1.4.1.2 para la elaboración del procedimiento formal de pruebas de intrusión. La aplicación de ambas metodologías, dividen el proyecto en 2 etapas previas: análisis de requerimientos y diseño las cuales son descritas brevemente en las secciones 1.4.1.3 para el perímetro de red y 1.4.1.4 para el Toolkit de pruebas de intrusión.

1.4.1.1 Marco de trabajo de la Arquitectura Cisco Safe

Cisco Safe es una arquitectura de Seguridad utilizada como modelo para implementación de redes seguras. Esta metodología hace uso de su enfoque modular para distribuir el crecimiento y seguridad de la red en bloques funcionales. En la figura 1-1 se muestra el diagrama de funcionalidad de Cisco Safe, el cual define 3 secciones, cada uno con sus bloques funcionales: Campus de la Empresa, Contorno de la Empresa y el Módulo del ISP.

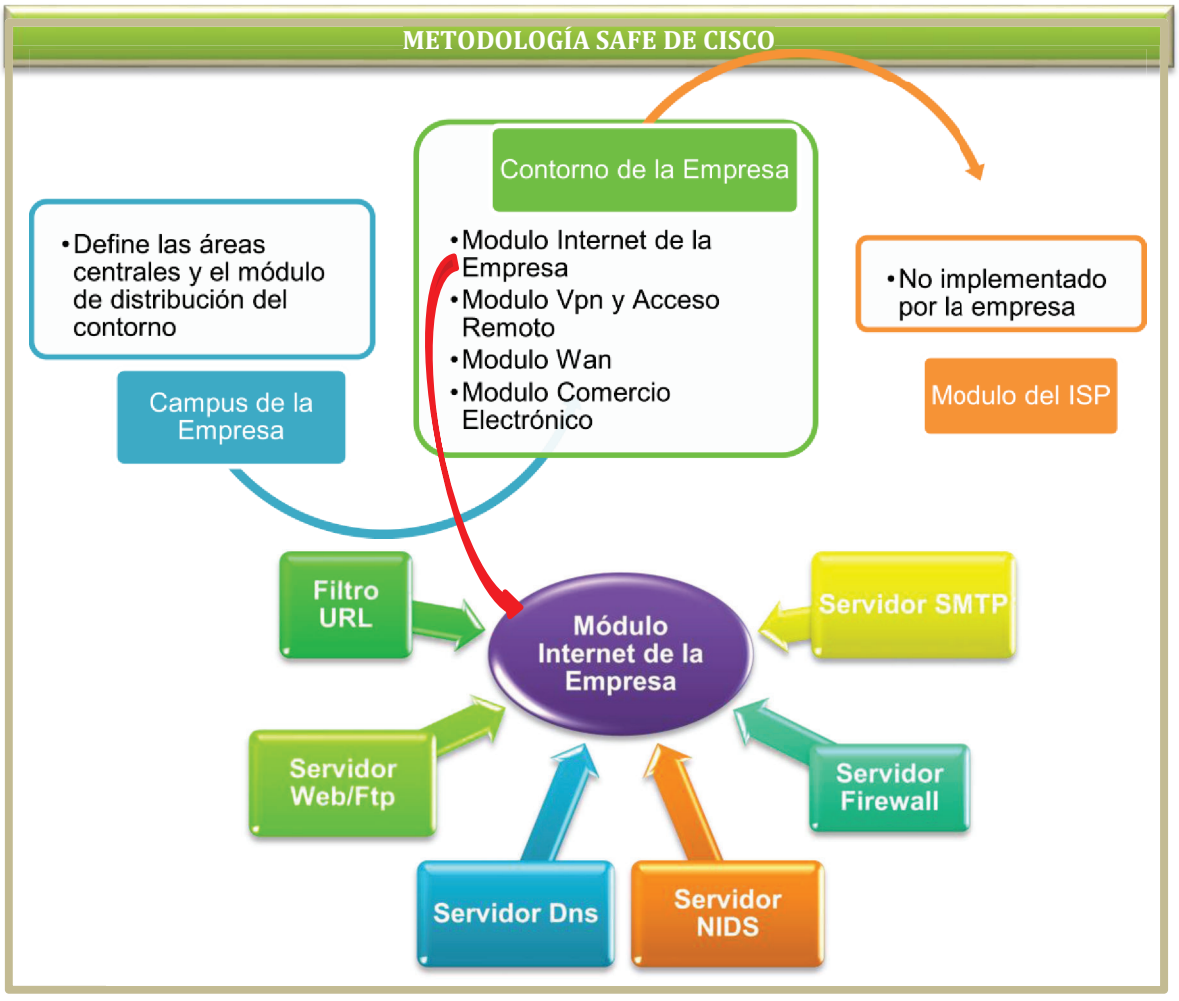


Figura 1-1 Marco de Trabajo de la Metodología Safe de Cisco

El proyecto actual centrará su análisis en la Sección “Contorno de la Empresa” con el Módulo “Internet de la Empresa””. Cisco Safe define a este Módulo para brindar a los

usuarios internos conexión a los servicios de Internet y acceso a los usuarios de internet a la información de los servidores públicos. Los elementos que conforman este módulo son: Filtro Url, Servidor Web/Ftp, Servidor Dns, Servidor NIDS, Servidor Firewall y Servidor SMTP.

1.4.1.2 Marco de trabajo de la Metodología OSSTMM (Manual de la Metodología Abierta de Testeo de Seguridad)

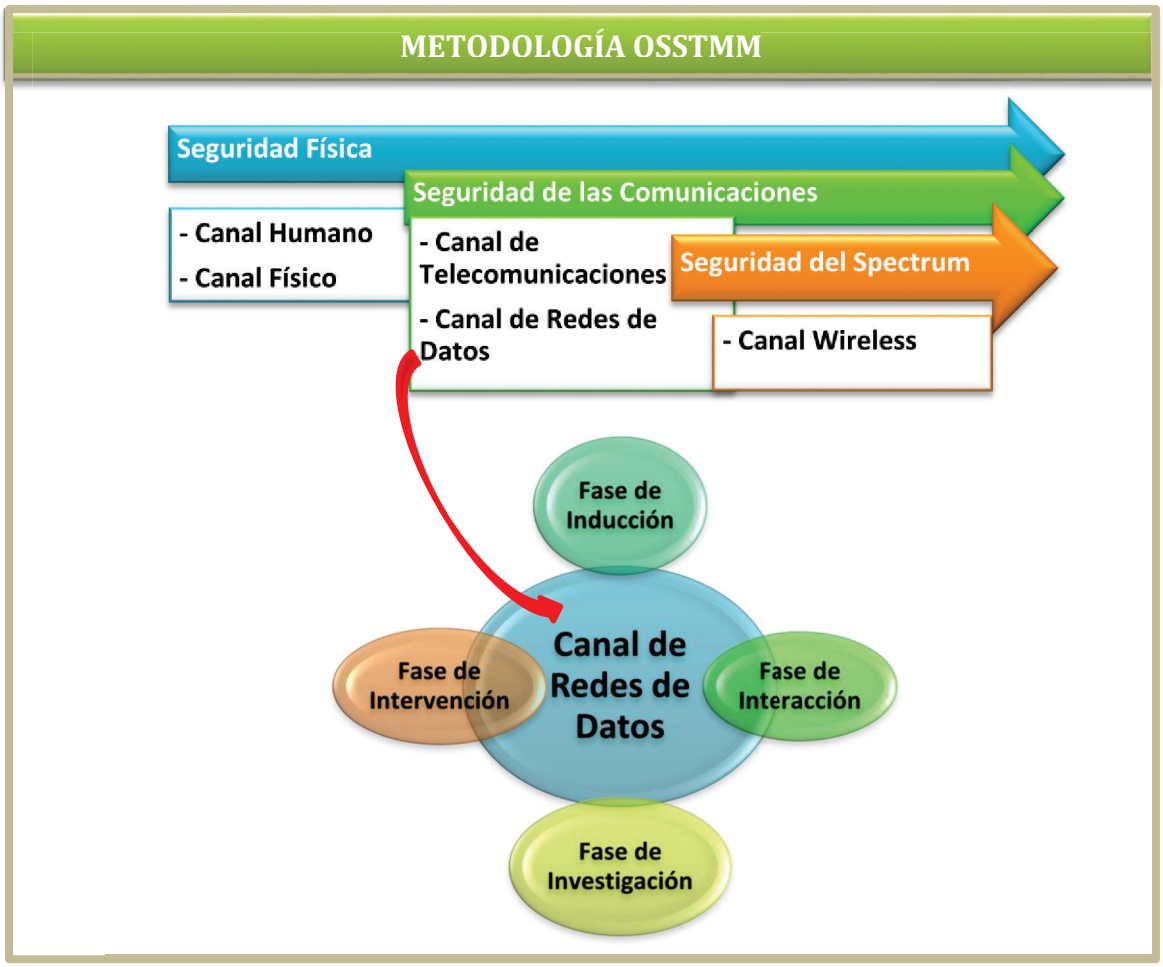


Figura 1-2 Marco de Trabajo de la Metodología OSSTMM

La metodología OSSTMM es un documento que reúne procedimientos estándar y ordenados de pruebas que debe realizar un profesional de la seguridad informática durante el desarrollo de auditorías de seguridad. En la figura 1-2 se muestra el diagrama de funcionalidad de OSSTMM, el cual trabaja sobre 3 ámbitos de

seguridad: Seguridad Física, Seguridad de las Comunicaciones y Seguridad del Spectrum. El proyecto actual centrará su análisis en el ámbito de "Seguridad de las Comunicaciones" en el "Canal de Redes de Datos". Este canal define las siguientes fases para una auditoría de seguridad: Fase de Inducción, Fase de Interacción, Fase de Investigación y Fase de Intervención, que posteriormente serán detalladas en la sección 1.4.1.4.

1.4.1.3 ANÁLISIS DE REQUERIMIENTOS Y DISEÑO DEL PERÍMETRO DE LA RED DE DATOS.

En esta sección se realizará el análisis de requerimientos y diseño del Perímetro de la red de datos.

a) Análisis de Requerimientos

Este análisis recopilará las necesidades descritas en la sección "Directrices del diseño" correspondiente al módulo "Internet de la Empresa" del documento "Cisco Safe: Un Modelo de seguridad para las redes de las empresas[3]. Los requerimientos expuestos por esta sección se detallan a continuación:

Requerimiento 1: Construir el perímetro de la red sobre una arquitectura de seguridad

Requerimiento 2: Evitar Accesos no autorizados y que la mayor parte de los ataques no afecten a los recursos de red más valiosos

Requerimiento 3: Filtrar las direcciones URL para permitir o denegar solicitudes Http.

[3] Directrices del diseño. Módulo Internet de la Empresa. Cisco Safe: Un modelo de seguridad para las redes de las empresas. (Fuente: http://www.cisco.com/web/ES/assets/docs/safe_wp1.pdf. Página 26)

Requerimiento 4: Bloquear las respuestas DNS para que responda únicamente a los comandos deseados e impida realizar transferencias de zona desde cualquier parte que no sean los dns internos.

Requerimiento 5. Proteger los servidores de direcciones públicas Web/Ftp contra ataques de desbordamiento SYN de TCP y filtrar para limitar las respuestas a los clientes.

El análisis detallado de requerimientos será realizado en la sección 2.1 del capítulo 2 del presente proyecto.

b) Diseño del Perímetro de la Red de datos

Los elementos dentro del Módulo "Internet de la Empresa" de Cisco Safe que permitirá cumplir los requerimientos detallados en la sección actual se detallan a continuación[4]:

Servidor Firewall

Este elemento implementado en base a políticas de filtrado de paquetes de entrada y salida en la herramienta Iptables del servidor, nos permitirá cumplir los requerimientos 1 y 2.

Servidor de Control de Contenido

Este servidor se implementará mediante Squid Proxy, declarando reglas de acceso y restricción en las peticiones Http que nos permitirá cumplir los requerimientos 1 y 3.

Servidor DNS

El servidor Dns local mediante las configuraciones apropiadas en los archivos de configuración de la zona Dns, nos permitirá cumplir los requerimientos 1 y 4.

[4] Dispositivos Principales y Directrices del Diseño. Cisco Safe: Un modelo de seguridad para las redes de las empresas. (Fuente: http://www.cisco.com/web/ES/assets/docs/safe_wp1.pdf. Página 24 y 26)

Servidores Públicos Web/Ftp

El servidor Web/Ftp con las configuraciones apropiadas en los filtros del Servidor firewall y el archivo de configuración de apache nos permitirá cumplir los requerimientos 1 y 5.

El diseño e implementación del perímetro de red será elaborado en la sección 2.2 del capítulo 2 del proyecto actual.

1.4.1.4 ANÁLISIS DE REQUERIMIENTOS Y DISEÑO DEL TOOLKIT DE PRUEBAS DE INTRUSIÓN.

En esta sección se realizará el análisis de requerimientos y diseño del Toolkit de pruebas de intrusión.

a) Análisis de Requerimientos

Este análisis recopilará las necesidades descritas en la sección "Proceso de 4 puntos" de la sección "Qué se necesita hacer" definido por la metodología OSSTMM[5]. Esta sección hace referencia a las fases que tiene una auditoría de seguridad. A continuación se detalla los requerimientos:

Requerimiento 1. El Toolkit de pruebas de intrusión debe asegurar una fase de inducción que permita una comprensión de los requisitos de la auditoría, el alcance y las limitaciones en el perímetro de la red.

Requerimiento 2. El Toolkit de pruebas de intrusión debe contar con una fase de interacción con los objetivos que permita determinar el ámbito de aplicación de la auditoría.

[5] Sección 2.6: Proceso de 4 puntos. Qué se necesita hacer. Manual de la metodología abierta de testeo OSSTMM. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf. Página 43).

Requerimiento 3. El Toolkit de pruebas de intrusión debe contar con una fase de investigación que permita verificar la forma de trabajo de los activos, así como las configuraciones de control y servicios en el perímetro de la red.

Requerimiento 4. El Toolkit de pruebas de intrusión debe contar con una fase de intervención que ponga a prueba los sistemas y servicios dentro del perímetro de la red.

El análisis detallado de requerimientos será elaborado en la sección 2.3 del capítulo 2 del proyecto actual

b) Diseño del Toolkit de Pruebas de Intrusión

Los módulos más relevantes de la sección "Seguridad de las Comunicaciones" en el "Canal de Redes de Datos" definidos por OSSTMM, que nos permitirá cumplir los requerimientos detallados en la sección actual se detallan a continuación[6]:

Modulo Logística y Verificación de la detección activa

El objetivo de estos módulos es preparar el entorno de prueba del canal para evitar falsos positivos y negativos y determinar los controles activos y pasivos. Estos módulos permiten el cumplimiento del Requerimiento 1.

Módulo Auditoría de Visibilidad, Verificación de Acceso y Verificación de Confianza

El propósito del módulo Auditoría de Visibilidad es enumerar los objetivos mediante la interacción directa e indirecta con los sistemas. El objetivo del módulo Verificación de Acceso es identificar los puntos de acceso, servicios y sistemas de autenticación. El objetivo del módulo Verificación de Confianza es validar la relación de confianza

[6] Sección 6.2: Módulos de Prueba. Manual de la metodología abierta de testeo OSSTMM. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf. Página 99).

entre los objetivos, donde exista acceso a la información sin necesidad de autenticación. El cumplimiento de estos 3 módulos nos permitirá cumplir con el Requerimiento 2.

Módulo Verificación de Configuración y Verificación de Exposición.

El objetivo del módulo Verificación de Configuración es probar la forma de trabajo de los activos y la posibilidad de eludir la seguridad funcional, explotando la configuración de controles de acceso y aplicaciones. El objetivo del módulo Verificación de Exposición es descubrir información pública que conduzca al acceso desde múltiples lugares con la misma autenticación. El cumplimiento de estos 2 módulos nos permitirá cumplir con el Requerimiento 3.

Módulo Auditoría de Privilegios y Validación de Supervivencia

El objetivo del módulo Auditoría de Privilegios es verificar el uso correcto de los sistemas de autenticación y autorización, comprobando la posibilidad de ingresos no autorizados y escalamiento de privilegios. El objetivo del módulo Validación de Supervivencia es verificar la resistencia y continuidad de los servicios ante cambios excesivos o degradación del servicio. El cumplimiento de estos 2 módulos permitirá cumplir con el requerimiento 4.

El diseño e implementación del Toolkit de pruebas de intrusión sobre el perímetro de red será elaborado en la sección 2.4 del capítulo 2 del proyecto actual.

1.4.2 SELECCIÓN DE LAS HERRAMIENTAS PARA EL PROCEDIMIENTO FORMAL DE PRUEBAS DE INTRUSIÓN

A continuación se detallan las herramientas que se utilizarán para el desarrollo del procedimiento formal de pruebas de intrusión, a fin de asegurar el cumplimiento de los requerimientos detallados en la sección actual 1.4.1.4.

a) FASE DE INDUCCIÓN[7]

En esta fase se comienza por un entendimiento de los requisitos de la auditoría, el alcance y sus limitaciones. A continuación se detalla el aporte de cada herramienta utilizada en esta sección en relación al concepto de la fase.

Herramienta Whois

Whois permite obtener información detallada acerca de las entidades en el internet como el propietario de un nombre de dominio, hosts de internet, servidores, cuyos datos son registrados a nivel mundial.

Herramienta Dig

Esta herramienta detalla información acerca de los host y direcciones IP correspondientes a los registros DNS. Para ello hace uso de la información del archivo resolv.conf del servidor DNS.

Herramienta Host

Este comando de Linux permite obtener información relativa a un dominio, como registros de correo, web, relación de servidores DNS, etc.

Herramienta Central Ops

Esta herramienta permite determinar información acerca del registro del dominio, registro de red, y registros DNS, entre otra información. El link está disponible a nivel de internet http://centralops.net/co/

Herramienta Apache Bench

Esta herramienta permite testear servidores web, verificando los tiempos de respuestas ante peticiones de consultas.

---

[7] Sección 6.2: Módulos de Prueba: Fase de Inducción, Sección A. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf. Página 99).

Herramienta ping

Esta herramienta permite probar los servicios utilizando el protocolo ICMP. Para ello hace uso de solicitudes ECHO request y las respuestas con ECHO reply.

Herramienta Squid Proxy Server

Este software muestra las funciones de control de contenido en cuanto a las peticiones http vía web. Para ello hace uso de reglas de control de acceso para limitar estas conexiones.

Herramienta Firewall Iptables

Este marco de trabajo disponible en el núcleo Linux permite interceptar y manipular paquetes de red. Maneja apertura, re direccionamiento, dropeo de puertos, así como filtrado de protocolos, servicios y aplicaciones.

Herramienta Nmap

Esta herramienta de exploración de redes permitirá realizar una exploración de red para determinar los puertos abiertos en los sistemas objetivos.

b) FASE DE INTERACCIÓN[8]

En esta fase se determinará el ámbito de aplicación de la auditoría en base a las interacciones con los objetivos. A continuación se detalla el aporte de cada herramienta utilizada en esta sección en relación al concepto de la fase.

Herramienta Wireshark

Esta herramienta de monitoreo nos permite capturar el tráfico de diferentes protocolos en la red, a fin de identificar los servicios utilizados en los host objetivos.

[8] Sección 6.2: Módulos de Prueba: Fase de Interacción, Sección B. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf. Página 100).

También permitirá verificar los números de secuencia en los paquetes intercambiados a los sistemas auditados.

Herramienta Maltego

Esta aplicación maneja una interfaz gráfica GUI que permite obtener información de los servicios y registros Dns asociados al dominio auditado.

Herramienta Nmap

Esta herramienta de exploración de redes y sondeo de seguridad, mediante escaneos específicos a los objetivos auditados, permitirá realizar exploración de protocolos IP, respuestas ante solicitudes ICMP y TCP específicas, búsqueda de puertos, servicios UDP/TCP abiertos, escaneo Fingerprinting para detección del Sistema Operativo, etc.

Herramienta ping

Esta herramienta nos permitirá verificar la conectividad hacia los sistemas/servicios objetivo estableciendo parámetros como el time to live TTL del protocolo IP.

Herramienta Traceroute:

Esta utilidad de red permitirá verificar los nodos intermedios hacia los sistemas auditados utilizando la característica del Time To Live (TTL) del protocolo IP.

Herramienta Firebug de Firefox[9]

Esta herramienta conocida como una extensión de Firefox maneja un paquete de utilidades para analizar el código fuente del sitio web local.

[9] Definición de Firebug. (Fuente: http://es.wikipedia.org/wiki/FireBug).

Herramienta: Tail

Este comando utilizado en sistemas Gnu/Linux permitirá ver la información de los archivo de logs en los sistemas auditados. Por defecto muestra las últimas 10 líneas del archivo.

Herramienta: hping2

Este analizador/ensamblador de paquetes TCP/IP verificará los IPID[10] de los datagramas IP en los sistemas objetivos para técnicas de Idle Scan sobre los sistemas auditados.

Herramienta What Web

Esta herramienta vía línea de comandos permitirá recopilar información acerca del sitio web local, como tecnología, CMS[11], plataforma de blogs, biblioteca JavaScript, Lenguaje PHP, números de versión, direcciones de correo, etc.

Herramienta CMS-Explorer:

Esta utilidad permitirá verificar los plugins del sitio web local.

Herramienta Netcat

Esta utilidad de red utiliza un intérprete de comandos para realizar pruebas de solicitud y acceso a los servicios auditados. También permitirá realizar pruebas de encaminamiento de paquetes a los host objetivos y verificar los banners asociados. Puede ser utilizado tanto en sistemas Windows como Unix Linux.

---

[10] El IPID es la identificación de un datagrama IP ubicado en la cabecera del paquete. Se usa con frecuencia en la fragmentación de datagramas IP para identificar los segmentos pertenecientes a un mismo datagrama.

[11] CMS Sistema de gestión de Contenidos, es un programa informático que permite crear una estructura de soporte (framework) para la creación y administración de contenidos, principalmente en páginas Web, por parte de los administradores, editores, participantes y demás usuarios. (Fuente: http://es.wikipedia.org/wiki/Sistema_de_gestión_de_contenidos).

Utilidad Uname

Permitirá verificar el nivel de parcheado, la arquitectura y procesador del sistema operativo auditado.

Herramienta Nikto

Esta herramienta permite realizar pruebas de vulnerabilidades en los servicios auditados, tales como detección de configuraciones erróneas, vulnerabilidades XSS, detección de ficheros en instalaciones por defecto, ataques de fuerza bruta, reportes en formato TXT, etc. En esta sección permitirá verificar pruebas de vulnerabilidad XSS contra el servidor web local. Viene incluida en el set de utilidades del Sistema Operativo Backtrack.

Herramienta XSSF

Esta herramienta utilizada como Framework permitirá explotar la vulnerabilidad XSS del sitio Web local. Permite administrar víctimas de ataques XSS genéricos y mantener dicha conexión a través de loops con Javascript. Viene incluida en el set de utilidades del Sistema Operativo Backtrack.

Herramienta Sql Map

Esta herramienta permitirá detectar y explotar las vulnerabilidades de tipo Inyección Sql, en el cual se obtiene el acceso total a la base de datos del servidor Web infectado con el código Sql malicioso. Esta utilidad viene incluida dentro del set de herramientas de Backtrack.

Herramienta Metasploit MSF

Esta herramienta permitirá verificar la vulnerabilidad Backdoor Vsftpd_234 en los sistemas auditados.

Herramienta: Jhon the Ripper

Es un programa de criptografía que mediante una combinación de ataques de fuerza bruta y diccionario, descifra contraseñas y permite determinar qué tan fuertes y robustas son las mismas. La forma de trabajo es mediante la generación de hashes por cada palabra o carácter utilizado para la prueba y compararlo con el archivo de contraseñas del sistema auditado. En esta sección permitirá determinar las credenciales de acceso en los sistemas objetivos.

Herramienta: Web Scarab

Es una utilidad que nos permite analizar aplicaciones web cuando se accede vía http como https. Permite reconocer y obtener URL's de una aplicación específica mediante la opción Proxy de intercepción, que permite al operador revisar y modificar las peticiones creadas por el navegador antes de que sean enviados al servidor, y para revisar y modificar respuestas enviadas por el servidor antes de que sean recibidas por el navegador[12]. En esta sección permitirá verificar y modificar las peticiones http en busca de errores lógicos en el sistema de autenticación. Existe información más detallada de los diferentes usos con esta herramienta en la guía de pruebas de OWASP [L3] Versión 3.0.

Herramienta: Cain y Abel

Cain y Abel es una herramienta usada principalmente por administradores de redes para comprobar el nivel de seguridad de una red. Permite recuperar contraseñas, usar técnicas de cracking como fuerza bruta, diccionario, criptoanálisis, sniffer de paquetes, etc. En esta sección permitirá realizar pruebas de "Main in the Midle[13]" y para suplantar la tabla Dns en los sistemas auditados.

[12] Definición de Web Scarab. (Fuente: https://www.owasp.org/index.php/Proyecto_WebScarab_OWASP).
[13] La técnica Main in the Midle permite interceptar el tráfico de una máquina víctima hacia un sistema objetivo, sin que ambos sistemas se den cuenta que esta comunicación ha sido intervenida.

Herramienta: nic.ec

La NIC es un operador de registro que se encarga de los registros de los nombres de dominio de un país. Se puede hacer uso de esta base de datos a nivel del Portal Web, para verificar y registrar los dominios creados, en este caso para Ecuador se puede acceder al sitio Web www.nic.ec. En esta sección permitirá examinar si existen registros de dominio de alto nivel GTLD similares al dominio local.

c) FASE DE INVESTIGACIÓN[14]

En esta fase se verifica la forma de trabajo de los activos, así como las configuraciones de control y servicios en los sistemas auditados. A continuación se detalla el aporte de cada herramienta utilizada en esta sección en relación al concepto de la fase.

Herramienta Google Hacking GH

Esta herramienta es un motor de búsqueda avanzado de Google que mediante el uso de operadores permite localizar cadenas específicas de texto en los resultados de búsquedas obtenidos. En esta sección permitirá validar la exposición de los sistemas, servicios y aplicaciones en esta motor de búsqueda público.

d) FASE DE INTERVENCIÓN

En esta fase se pone a prueba los sistemas y servicios auditados mediante interrupciones que puedan afectar las respuestas de las pruebas menos invasivas. A continuación se detalla el aporte de cada herramienta utilizada en esta sección en relación al concepto de la fase.

---

[14] Sección 6.2: Módulos de Prueba: Fase de Investigación, Sección C. (Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf. Página 101).

Herramienta: Web Scarab

Esta utilidad permitirá analizar las consultas http get y post a fin de verificar la posibilidad de autorizaciones fraudulentas y acceso a información sensible en el sitio web local.

Herramienta Mempodipper[15]

Este exploit basado en el CVE 2012-0056 detalla la vulnerabilidad basada en el acceso a /proc/pid/mem para pruebas de elevación de privilegios en los sistemas Linux auditados. Todos los kernels Linux 2.6.39 y superiores (incluyendo la serie 3.x) son vulnerables a este CVE. Actualmente la mayoría de las distribuciones ya han parcheado esta vulnerabilidad.

Herramienta Hping3

Es la versión actualizada de hping2. Permite analizar/ensamblar paquetes TCP/IP para realizar auditorías de seguridad, testeo de redes, firewalls, escaneo de puertos, etc. En esta sección permitirá realizar pruebas de Denegación de Servicio DoS al sitio web local.

[15] Escalación de privilegios vía SUID. Introducción a Mempodipper. (Fuente: http://blog.zx2c4.com/749. Página 1)

CAPÍTULO 2: IMPLEMENTACIÓN DEL PERÍMETRO DE LA RED Y DESARROLLO DEL TOOLKIT DE PRUEBAS DE INTRUSIÓN

En este capítulo se define el diseño e implementación del perímetro de la red, y del toolkit de pruebas de intrusión. La definición del perímetro de red se realiza en la sección 2.2 utilizando los lineamientos de la metodología Cisco Safe. Para cumplir con este objetivo, previamente se realiza el análisis de los requerimientos en la sección 2.1, utilizando los requerimientos descritos en la sección 1.4.1.3 del presente proyecto. La definición del toolkit de pruebas de intrusión se realiza en la sección 2.4, utilizando las tareas descritas por la metodología OSSTMM. Para cumplir con este objetivo, se realiza el análisis de los requerimientos en la sección 2.3, en base a los requerimientos descritos en la sección 1.4.1.4 del actual proyecto.

2.1 ANÁLISIS DE LOS REQUERIMIENTOS PARA EL DESARROLLO DEL PERÍMETRO DE LA RED DE DATOS.

El análisis de los requerimientos se basa en los lineamientos de diseño estipuladas en la sección “Directrices del diseño” correspondiente al módulo “Internet de la Empresa” del documento “Cisco Safe: Un Modelo de seguridad para las redes de las empresas”. Estos requerimientos permiten plantear recomendaciones de seguridad para el diseño de infraestructura, redes, servicios y aplicaciones en los ambientes relativos al Contorno de las Empresas.

2.1.1 REQUERIMIENTO 1. *“Construir el perímetro de la red sobre una arquitectura de seguridad”*

Para cumplir este requerimiento se diseña e implementa el perímetro de red bajo los lineamientos del módulo “Internet de la Empresa” descritos por la metodología Cisco Safe[16]. En la figura 2-1 se esquematiza los dispositivos definidos por este módulo que permitirán cumplir el requerimiento 1.

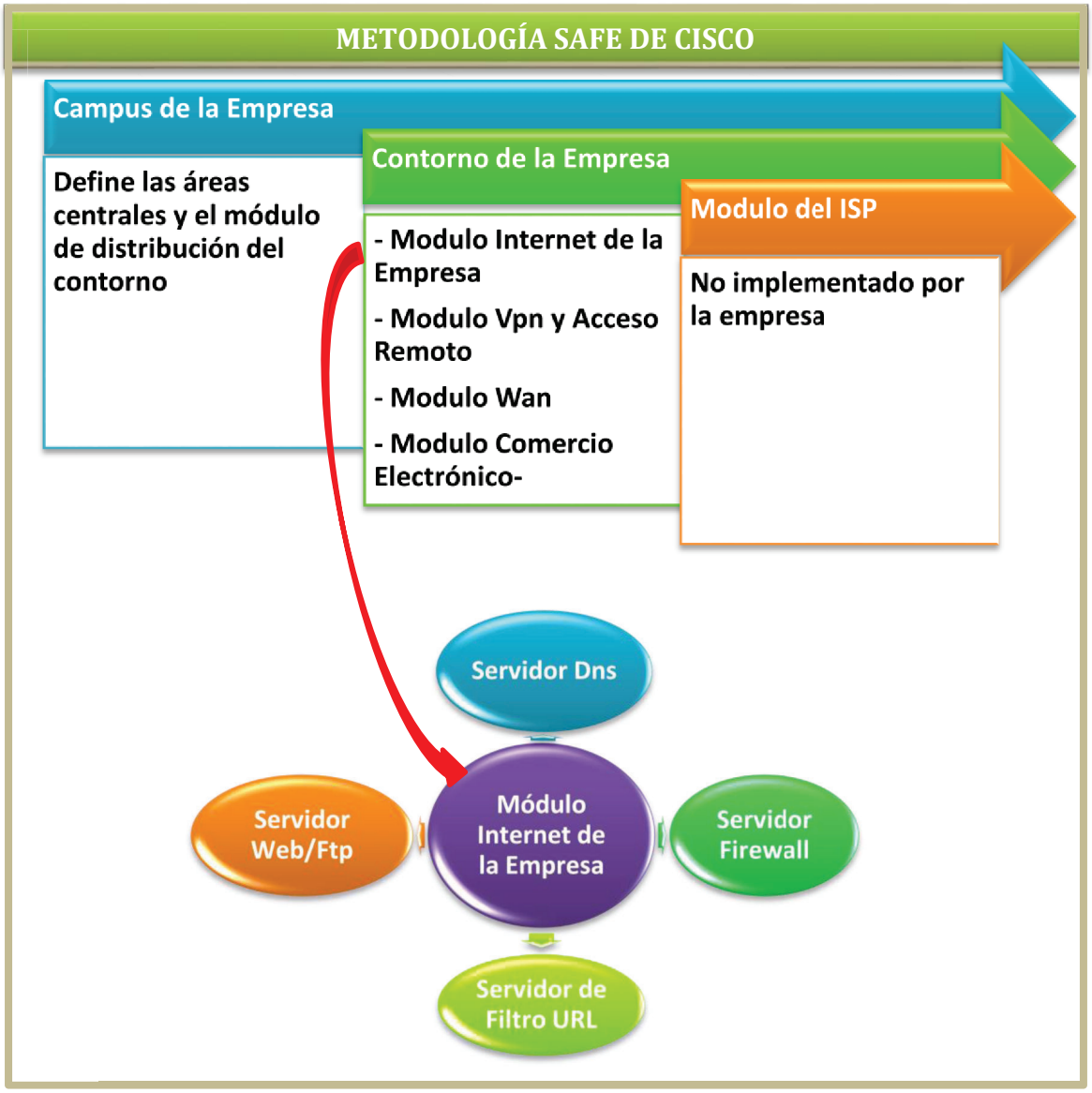


Figura 2-1 Dispositivos del Módulo “Internet de la Empresa” de Cisco Safe

[16] Dispositivos Principales. Módulo de Internet de la Empresa. Cisco Safe: Un modelo de seguridad para las redes de las empresas. (Fuente: http://www.cisco.com/web/ES/assets/docs/safe_wp1.pdf. Página 24)

Para el proyecto actual se utiliza los siguientes elementos: Servidor Web Http, Servidor Ftp, Servidor Dns, Servidor Firewall y Servidor de filtro de direcciones URL. La función de cada uno de estos servicios dentro del perímetro de red, se detalla a continuación en las subsecciones 2.1.2 a la 2.1.5 respectivamente.

2.1.2 REQUERIMIENTO 2: *"Evitar accesos no autorizados y que la mayor parte de los ataques no afecten a los recursos de red más valiosos"*

Para cumplir este requerimiento, se utiliza el elemento "Firewall" del módulo "Internet de la Empresa" descrito por la metodología Cisco Safe. El Servidor Firewall bloquea el acceso no autorizado mediante reglas de filtrado de paquetes entrantes y salientes en el iptables del servidor. En la figura 2-2 se describe esta relación de los controles, con el modelo de referencia de redes TCP/IP[17], el elemento "Firewall" y la política de seguridad para garantizar el cumplimiento del requerimiento 2.

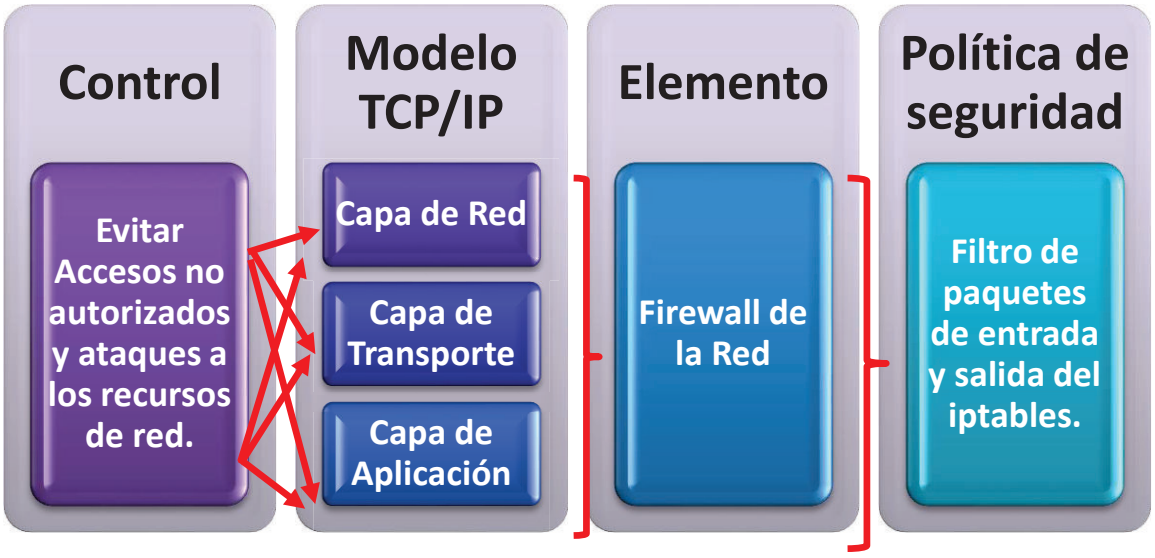


Figura 2-2 Control para evitar accesos no autorizados y ataques a los recursos de red

[17] Modelo de Referencia TCP/IP. El modelo TCP/IP describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que un equipo pueda comunicarse en una red. (Fuente: http://es.wikipedia.org/wiki/Modelo_TCP/IP)

2.1.3 REQUERIMIENTO 3:"*Filtrar las direcciones URL para permitir o denegar solicitudes http"*

Para cumplir este requerimiento, se utiliza el elemento "Servidor de filtro de direcciones URL" del módulo "Internet de la Empresa" descrito por la metodología Cisco Safe. El Servidor Squid Proxy lo utilizamos para filtrar las direcciones URL ante solicitudes del protocolo Http. En la figura 2-3 describimos esta relación del control, con el modelo de referencia de redes TCP/IP, el elemento "Control de Contenido" y la política de seguridad para garantizar el cumplimiento del requerimiento 3.

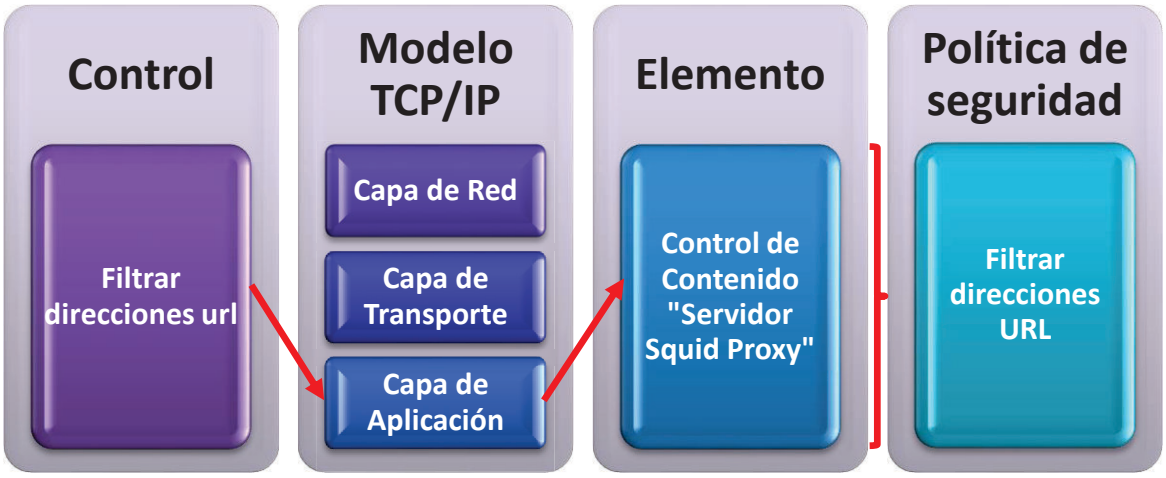


Figura 2-3 Control para filtro de direcciones URL

2.1.4 REQUERIMIENTO 4:"*Bloquear las respuestas dns para que responda únicamente a los comandos deseados e impida realizar transferencias de zona desde cualquier parte que no sean los dns internos"*

Para cumplir este requerimiento, se utiliza el elemento "Servidor Dns" del módulo "Internet de la Empresa" descrito por la metodología Cisco Safe. El Servidor DNS permite el envío y respuestas de consultas Dns y permitirá bloquear las respuestas para responder únicamente a los comandos deseados. También impedirá realizar

transferencias de zonas que no sean los Dns internos, mediante configuraciones de seguridad en la zona directa, inversa, y en el archivo de configuración “named.conf”. En la figura 2-4 se describe esta relación del control, con el modelo de referencia de redes TCP/IP, el elemento “Servidor Dns” y la política de seguridad para garantizar el cumplimiento del requerimiento 4.

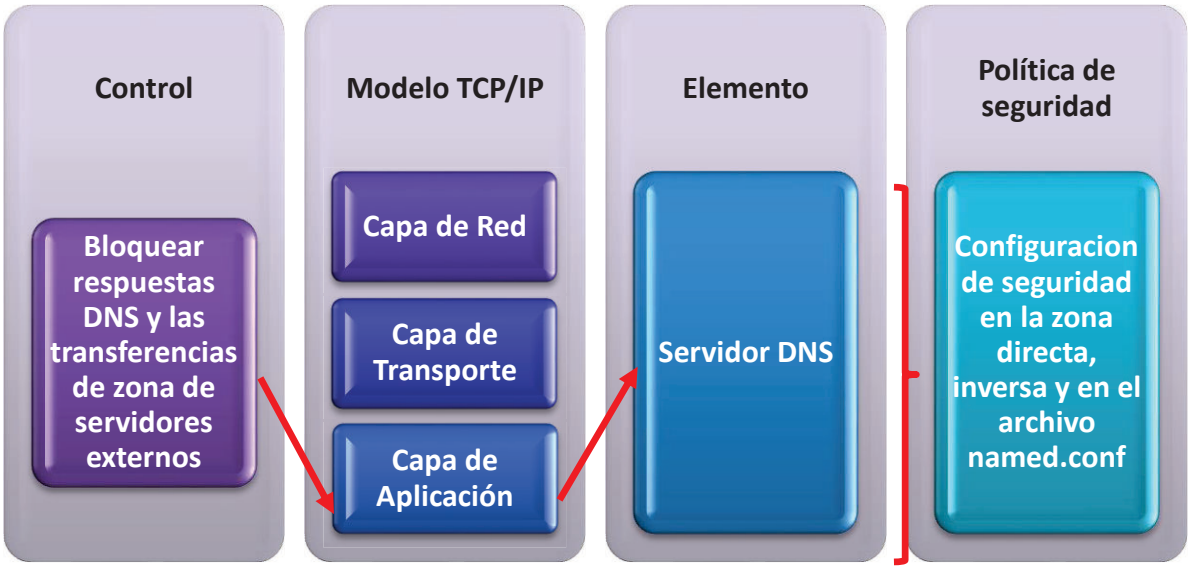


Figura 2-4 Control de seguridad en el Servidor DNS

2.1.5 REQUERIMIENTO 5. *”Proteger los servidores de direcciones públicas (web/ftp) contra ataques de desbordamiento syn de tcp y filtrar para limitar las respuestas a los clientes”*

Para cumplir este requerimiento, se utiliza el elemento “Servidor Ftp/Http” del módulo “Internet de la Empresa” descrito por la metodología Cisco Safe. Se desarrollará una página Web local para el proyecto actual denominada www.iza.sanhueza.ec, y también un Servidor Ftp local para el proceso de subida o descarga de archivos. Este servidor Web/Ftp se protegerá contra ataques de desbordamiento SYN de TCP mediante configuraciones apropiadas en el Kernel del Servidor Firewall.

Además se filtrará para limitar las respuestas a los clientes a través de configuraciones de seguridad en los archivos de configuración “httpd.conf” y “vsftpd.conf” de los servidores Web y Ftp respectivamente. En la figura 2-5 se describe esta relación del control, con el modelo de referencia de redes TCP/IP, el elemento “Servidor Ftp/Http” y la política de seguridad para garantizar el cumplimiento del requerimiento 5.

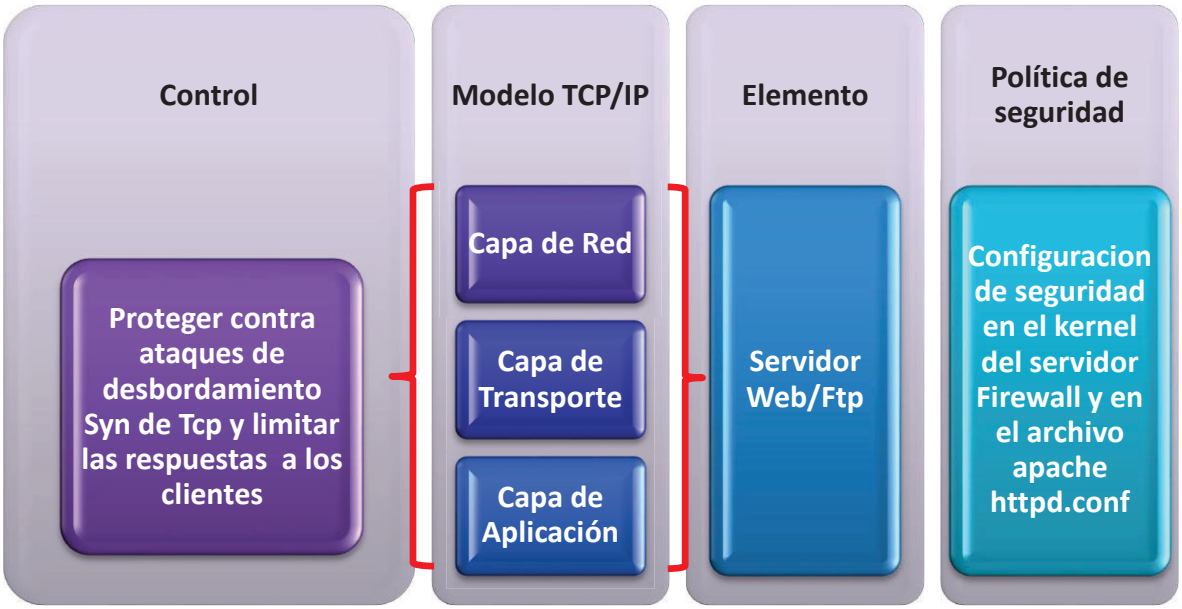


Figura 2-5 Control de seguridad en los servidores públicos Web/Ftp

2.2 DISEÑO E IMPLEMENTACIÓN DEL PERÍMETRO DE LA RED DE DATOS

Esta sección contiene el diseño e implementación del perímetro de la red de datos en base a los 5 requerimientos descritos en la sección anterior 2.1. Para ello se ha dividido en las subsecciones 2.2.1 para definir el área del perímetro de la red, seguidas de las subsecciones 2.2.2 hasta la 2.2.6 para detallar el diseño e implementación de los elementos del perímetro de la red de datos.

2.2.1 DEFINICIÓN DEL ÁREA DEL PERÍMETRO DE LA RED DE DATOS

Se diseña el perímetro de la red de datos bajo los lineamientos de la arquitectura de seguridad Cisco Safe. Esta arquitectura está basada en un enfoque modular para implementar la seguridad en bloques funcionales y relacionarlos entre sí. En la figura 2-6 se esquematiza el diagrama del perímetro de la red para el proyecto actual. Por un lado se define el módulo del ISP, que no es implementado por la empresa, el módulo "Internet de la Empresa" del bloque "Contorno de la Empresa" para definir el perímetro de la Red, y el módulo Campus de la Red que no será implementado por el proyecto actual, ya que no es el objeto de análisis del mismo. En el módulo "Internet de la Empresa" se implementa los siguientes elementos:

- Servidor Web/Ftp/Dns
- Servidor Firewall y Control de Contenido

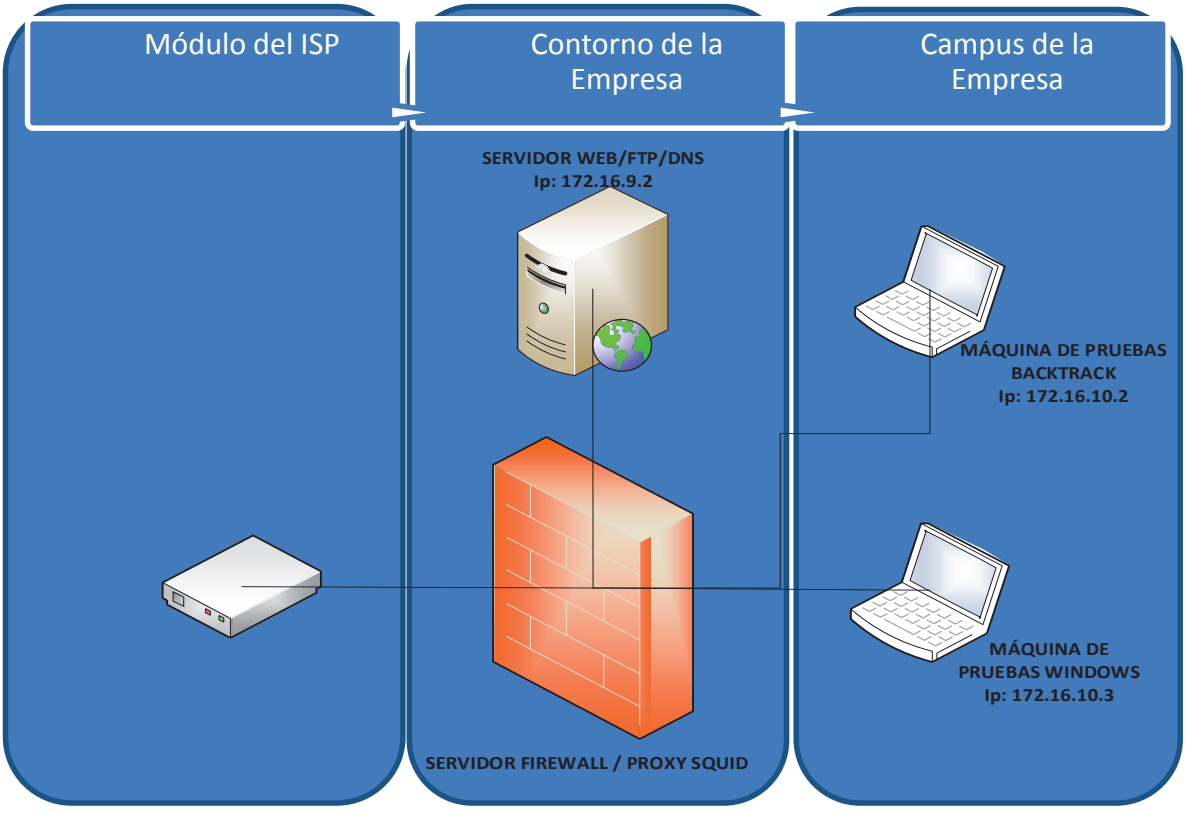


Figura 2-6 Diagrama del Perímetro de Red

La infraestructura lógica del perímetro de red se implementa bajo una plataforma virtualizada tanto para el Servidor Dns/Ftp/Web, el servidor Firewall/Control de Contenido, y para las máquinas de pruebas Backtrack y Windows para realizar las pruebas de intrusión sobre los objetivos del Perímetro de Red.

2.2.2 IMPLEMENTACIÓN DEL SERVIDOR FIREWALL

Se Implementa el Servidor Firewall para evitar los accesos no autorizados mediante políticas de seguridad. Para ello el Firewall utiliza el sistema Iptables, el cual maneja las secciones Filter, Nat y Mangle para el manejo de reglas en el Servidor. En la siguiente tabla 2-1 se detalla las políticas de seguridad y su relación con las secciones del Firewall y las cadenas que utiliza para ejecutar estas políticas. Las políticas de seguridad se plantearon en base a los servicios que se utilizarán en el perímetro de red para que puedan interactuar ante las consultas internas y externas.

Política de Seguridad	Sección Firewall	Cadena de la Sección
a) Permitir las conexiones web, web seguro, correo, ftp, telnet, webmin[18] y dns.	Filter	Input y Forward
b) Permitir el acceso hacia el Internet desde las redes Lan (Segmento del Servidor Web/Ftp/Dns y Segmento de las máquinas de prueba).	Filter	Input y Forward
c) Enmascarar las conexiones Ftp, Http seguro, Ssh, Smtp, Webmin, Dns y Telnet hacia servidores externos.	Nat	Postrouting

[18] Webmin: Es una herramienta de configuración de sistemas accesible vía web para sistemas Unix. Con él se pueden configurar aspectos internos de muchos sistemas operativos, como usuarios, cuotas de espacio, servicios, archivos de configuración, apagado del equipo. Fuente: (http://es.wikipedia.org/wiki/Webmin).

d) Enmascarar el protocolo Icmp para permitir pruebas de conectividad ping hacia servidores externos.	Nat	Postrouting
e) Enmascarar las redes Lan (Segmento del Servidor Web/Ftp/Dns y Segmento de las máquinas de prueba) para la conexión a Internet.	Nat	Postrouting
f) Re direccionar las peticiones http de la Red Lan al puerto 3128, a fin de re direccionar las peticiones Web al Servidor Squid Proxy para filtrar direcciones URL.	Nat	Prerouting

Tabla 2-1 Definición de Políticas de Seguridad en el Servidor Firewall

La Sección Filter utiliza las cadenas INPUT para las peticiones de entrada hacia el firewall, FORWARD para los paquetes que pasan a través del firewall, y OUTPUT para los paquetes originados por el firewall. En la figura 2-7 se visualiza esta sección conforme a las Políticas de seguridad a y b definidas en la Tabla 2-1. Para cumplir con esto se definió una cadena propia llamada FILTER_FIREWALL que la relacionamos con las cadenas INPUT Y FORWARD.

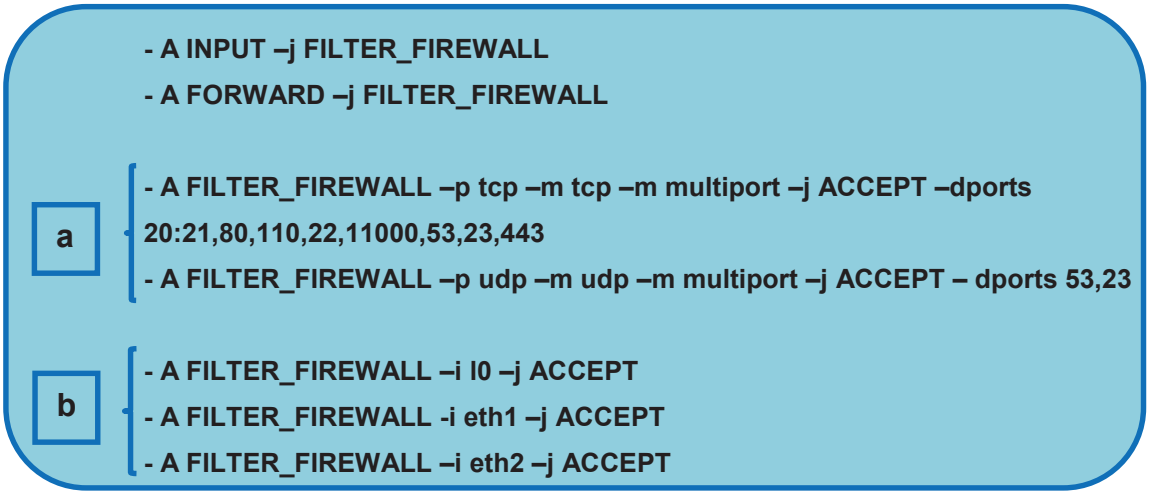


Figura 2-7 Sección Filter del Firewall

La Sección Nat usa las cadenas PREROUTING para manipular los paquetes de entrada antes de ser procesados por la tabla de enrutamiento, POSTROUTING para los paquetes salientes ya procesados por la tabla de enrutamiento y OUTPUT para los paquetes originados por el firewall. En la figura 2-8 se visualiza esta sección conforme a las Políticas de seguridad c, d, e y f definidas en la tabla 2.1. El archivo de configuración completo del Iptables del Firewall se detalla en el Anexo N. 1.

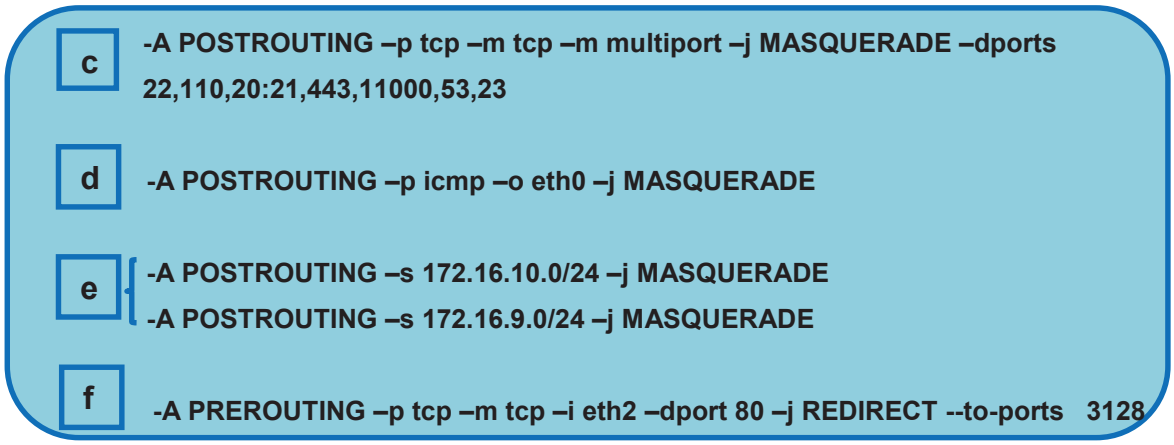


Figura 2-8 Sección Nat del Firewall

2.2.3 IMPLEMENTACIÓN DEL SERVIDOR FILTRO DE DIRECCIONES URL (PROXY SQUID)

El servidor Proxy Squid se utiliza para el control de contenido Web. Las configuraciones de acceso se definen en el archivo de configuración “squid.conf” del servidor. En la siguiente tabla 2-2 se detallan las políticas de seguridad y el módulo a implementar dentro del archivo de configuración. Estas políticas de seguridad se plantearon en base a los servicios que se utilizarán en el perímetro de red para permitir su acceso y evitar el uso de tráfico no recomendable en empresas corporativas.

Política de Seguridad	Módulo
a) Declarar los puertos seguros para permitir el acceso.	Existe una lista de puertos predefinidos como seguros en la sección Safe_ports del archivo de configuración.
b) Definir los segmentos de red Lan que podrán navegar vía Http.	Creación de la acl llamada “permitido”
c) Denegar el tráfico pornográfico, y redes sociales.	Creación de regla llamada “Redes_Sociales” en sección url_regex. El archivo asociado a la regla se localiza en /etc/squid/Redes_Sociales y contiene las palabras que definen el control de acceso.

Tabla 2-2 Definición de Políticas de seguridad en el Servidor Filtro URL

En la figura 2-9 se visualiza la configuración de los módulos que permiten cumplir las Políticas de seguridad a, b y c definidas previamente en la tabla 2-2. En la figura 2-10 se visualiza el contenido del archivo Redes_Sociales que hace referencia a las palabras que definen el control de acceso. El archivo de configuración completo del Servidor Squid Proxy se encuentra detallado en el Anexo N. 2

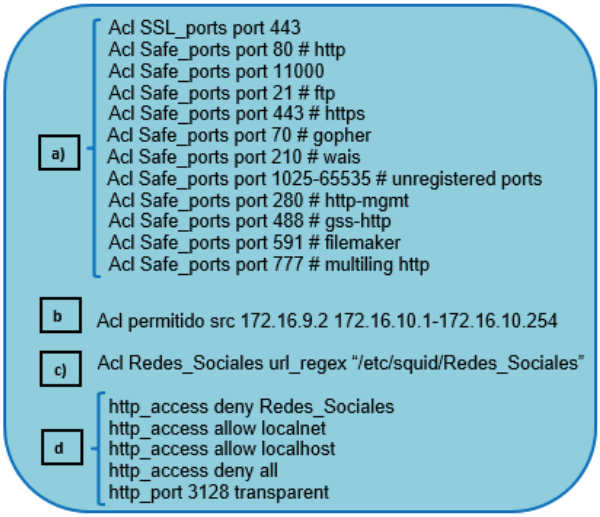


Figura 2-9 Configuración de módulos en el archivo squid.conf

Figura 2-10 Definición del Archivo Redes_Sociales

2.2.4 IMPLEMENTACIÓN DEL SERVIDOR DNS [W1]

El Servidor Dns permite la traducción de nombres de dominio a direcciones IP y viceversa. Se configura un dominio local definido como "iza.sanhueza.ec" para el proyecto actual, que no se publicará al internet. Las configuraciones de seguridad se establecen en el archivo de configuración "named.conf" y en los archivos de la zona directa[19] "iza.sanhueza.ec.zone ", y de la zona inversa[20] "9.16.172.in-addr.arpa.zone" del servidor Dns. En la siguiente tabla 2-3 se detalla las políticas de seguridad y el módulo a implementar. Estas políticas de seguridad se plantearon en base a las recomendaciones de seguridad definidas por la metodología Cisco Safe en cuanto a las directrices de diseño para el Servidor Dns[21].

Política de Seguridad	Módulo
a) Permitir la transferencia de zona solo a redes autorizadas, a fin de limitar los ataques externos	1. Configuraciones de seguridad en el archivo de la zona directa, "iza.sanhueza.ec.zone" para definir la transferencia de zona únicamente

[19] Zona Directa: Los clientes DNS pueden usar esta zona para obtener información, como las direcciones IP correspondientes a los servicios o nombres de dominio DNS, que está almacenada en la zona. (Fuente: https://technet.microsoft.com/es-es/library/cc816891 (v=ws.10).aspx).

[20] Zona Inversa: Restablece la asignación de direcciones IP a los nombres de dominio DNS. (Fuente: https://technet.microsoft.com/es-es/library/cc816891 (v=ws.10).aspx).

[21] Directrices de diseño. Módulo de Internet de la Empresa. Cisco Safe Un modelo de seguridad para las redes de las empresas. (Fuente: http: //www.cisco.com/web/ES/assets/docs/safe_wp1.pdf, Página. 26)

	al servidor Dns. 2. Configuraciones de seguridad en el archivo de la zona inversa, “9.16.172.in-addr.arpa.zone”.
b) Configurar el Dns como Servidor Cache para la red local y delegar las consultas recursivas[22] hacia los Dns del ISP para zonas distintas al dominio local “iza.sanhueza.ec”.	Configuración en el archivo “named.conf” en la sección Forwarders para el reenvío de solicitudes a los Dns del ISP.
c) Permitir las consultas Dns solo desde los segmentos de red interna (Segmento del Servidor Web/Ftp/Dns y Segmento de las máquinas de prueba) y bloquear el forward de consultas desde el Internet, a fin de evitar ataques de envenenamiento al caché DNS que puedan re direccionar las peticiones hacia servidores falsos.	Configuración en el archivo “named.conf” en la sección allow–query para limitar las consultas Dns a los segmentos de Lan interna.

Tabla 2-3 Definición de Políticas de seguridad en el Servidor DNS.

En las figuras 2-11, 2-12 y 2-13 se muestran las configuraciones de seguridad para cumplir las Políticas de seguridad a, b y c respectivamente definidas previamente en la tabla 2.3. El archivo de configuración completo del Servidor Dns se detalla en el Anexo N. 3.

[22] Consultas recursivas: Establece el reenvío de solicitudes Dns a una zona autoritativa no conocida, es decir a la zona Dns del ISP.

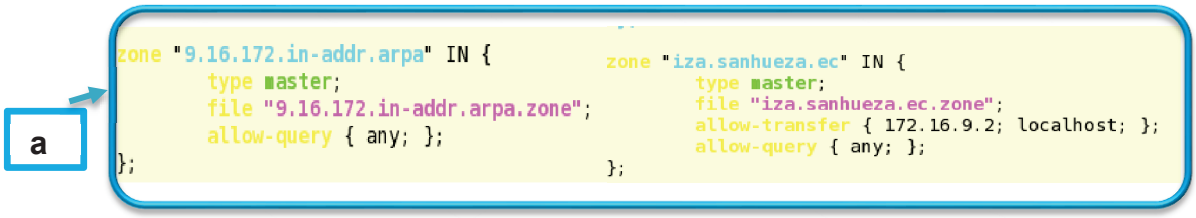


Figura 2-11 Zona directa e inversa del Servidor Dns

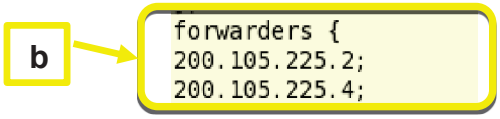


Figura 2-12 Sección Forwarders del archivo "named.conf" para el reenvío de consultas

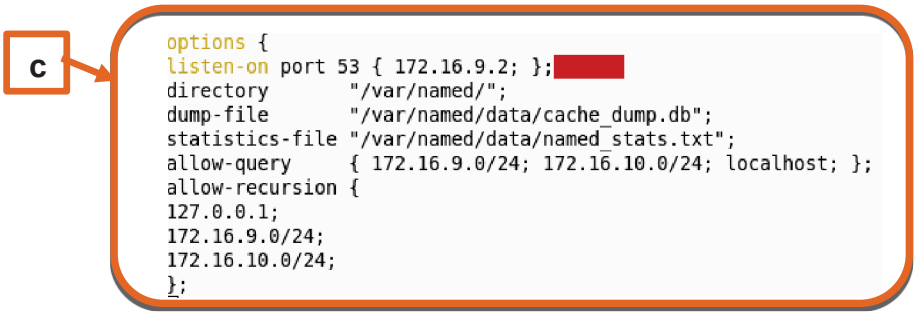


Figura 2-13 Sección Query del archivo "named.conf" para limitar las consultas

2.2.5 IMPLEMENTACIÓN DEL SERVIDOR WEB/FTP [W2]

El nombre del sitio web es "www.iza.sanhueza.ec" el cual proporciona información general del proyecto actual. La página Web funciona a nivel local con el servidor Apache de Linux y el gestor de contenido "Joomla". La instalación y configuración del servidor Web se lo detalla en el Anexo N. 4. El servicio Ftp lo implementamos para el proceso de subida o descarga de archivos a nivel local. La instalación y configuración del servidor Ftp se lo detalla en el Anexo N. 5. En la tabla 2-4 se detalla las políticas de seguridad y el módulo a implementar. Estas políticas de seguridad se plantearon

en base a las recomendaciones de seguridad definidas por la metodología Cisco Safe en cuanto a las directrices de diseño para los servidores públicos Web/Ftp[23].

Política de Seguridad	Módulo
a) Proteger contra ataques de desbordamiento SYN de TCP[24].	Configuraciones de seguridad en el parámetro net.ipv4.tcp_syncookies del sistema Kernel[25] al valor de 1.
b) Filtrar para limitar las respuestas a los clientes.	Configuraciones de seguridad en el archivo de configuración "httpd.conf" y "vsftpd.conf" del Servidor Web y Ftp respectivamente.

Tabla 2-4 Políticas de seguridad del Servidor Web/Ftp

En la figura 2-14 se verifica la configuración de seguridad que permite cumplir la Política de seguridad a definida previamente en la tabla 2-4.

```
Sysctl –a | grep net.ipv4.tcp_syncookies
Net.ipv4.tcp_syncookies = 1
```

Figura 2-14 Variable Syncookies del Kernel del Firewall

En la figura 2-15 se muestra las configuraciones, para cumplir la política de seguridad b) limitando las respuestas del servidor tales como información de versión y tipo de servidor, cuando se devuelve mensajes de error. Se configura la directriz Server Signature en Off en el archivo de configuración "httpd.conf" del Servidor Web y la directiva ftp_banner del archivo de configuración "vsftpd.conf" para indicar un mensaje de saludo, que reemplaza la salida por defecto (Nombre y versión) del Servidor Ftp.

---

[23] Directrices de diseño. Módulo de Internet de la Empresa. Cisco Safe Un modelo de seguridad para las redes de las empresas. (Fuente: http: //www.cisco.com/web/ES/assets/docs/safe_wp1.pdf, Página. 26)

[24] Ataques de desbordamiento Syn de TCP: Estos ataques usan técnicas de denegación de servicio que mediante un flujo de paquetes TCP/SYN con la Dirección Ip origen falsificada, saturan al host objetivo de múltiples peticiones Tcp/Syn, hasta que el equipo deje de responder otras solicitudes.

[25] El Kernel o núcleo de linux se define como el corazón del sistema operativo. Es el encargado de que el software y el hardware del ordenador puedan trabajar juntos. (Fuente: http://www.linux-es.org/kernel).

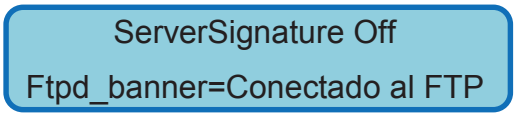


Figura 2-15 Parámetro Server Signature y ftpd_banner

2.3 ANÁLISIS DE LOS REQUERIMIENTOS PARA EL DESARROLLO DEL TOOLKIT DE PRUEBAS DE INTRUSIÓN

El análisis de los requerimientos se basa en las fases de la auditoría de seguridad recomendada por los lineamientos de la metodología OSSTMM V3. A este proceso de pruebas desde su inicio hasta su conclusión, OSSTMM lo denomina 4pp[26] y comprende 4 fases detalladas a continuación:

2.3.1 REQUERIMIENTO 1.*"El Toolkit de pruebas de intrusión debe asegurar una fase de inducción que permita una comprensión de los requisitos de la auditoría, el alcance y las limitaciones en el perímetro de la red"*

El objetivo de la Fase de Inducción es comprender los requisitos de la auditoría, el alcance y las limitaciones en el perímetro de la red. Para cumplir esta fase utilizamos los módulos Logística y Verificación de la detección Activa descritos por la metodología OSSTMM V3. El módulo Logística nos permite preparar el entorno de prueba del canal, y evitar falsos positivos/negativos que puedan conducir a resultados imprecisos en la prueba. El módulo Verificación de la Detección Activa nos permite determinar los controles activos y pasivos. En la figura 2-16 se describe esta relación de la Fase de Inducción, con el modelo de referencia TCP/IP, los

[26] Fases de la metodología OSSTMM basada en el 4PP (puntos de proceso) que desglosa el proceso de las pruebas desde su inicio hasta su conclusión. Fuente (http://www.isecom.org/mirror/OSSTMM.3.pd Sección 2.6 Four Point Process Página. 43)

módulos y pruebas necesarias descritos por OSSTMM V3, para garantizar el cumplimiento del Requerimiento 1.

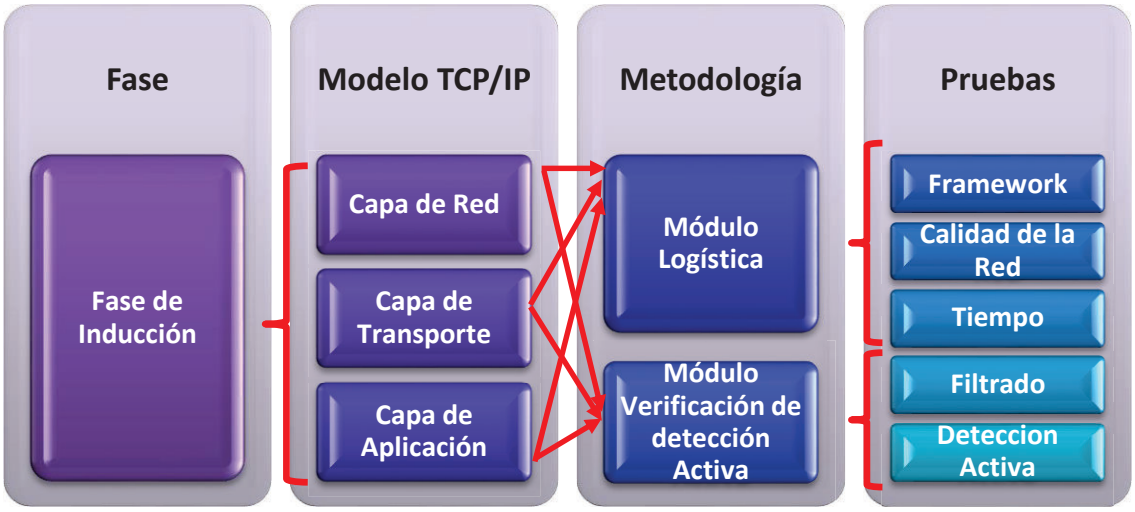


Figura 2-16 Fase de Inducción del Toolkit de pruebas de intrusión

2.3.2 REQUERIMIENTO 2.*"El Toolkit de pruebas de intrusión debe contar con una fase de interacción con los objetivos para determinar el ámbito de aplicación de la auditoría"*

El objetivo de esta Fase de Interacción es definir el ámbito de aplicación de la auditoría. Para cumplir esta fase se utiliza los módulos Auditoria de Visibilidad, Verificación de Acceso y Verificación de confianza descritos por la metodología OSSTMM V3. El módulo Auditoría de Visibilidad permite enumerar los elementos del perímetro de la red mediante la interacción directa e indirecta con los sistemas. El módulo Verificación de Acceso permite identificar los principales puntos de acceso. El módulo Verificación de Confianza permite realizar pruebas de confianza entre los sistemas del perímetro de red sin la necesidad de identificación o autenticación. En la Figura 2-17 se describe esta relación de la Fase de Interacción con el modelo de referencia TCP/IP, los módulos y pruebas necesarias descritos por OSSTMM V3, para garantizar el cumplimiento del Requerimiento 2.

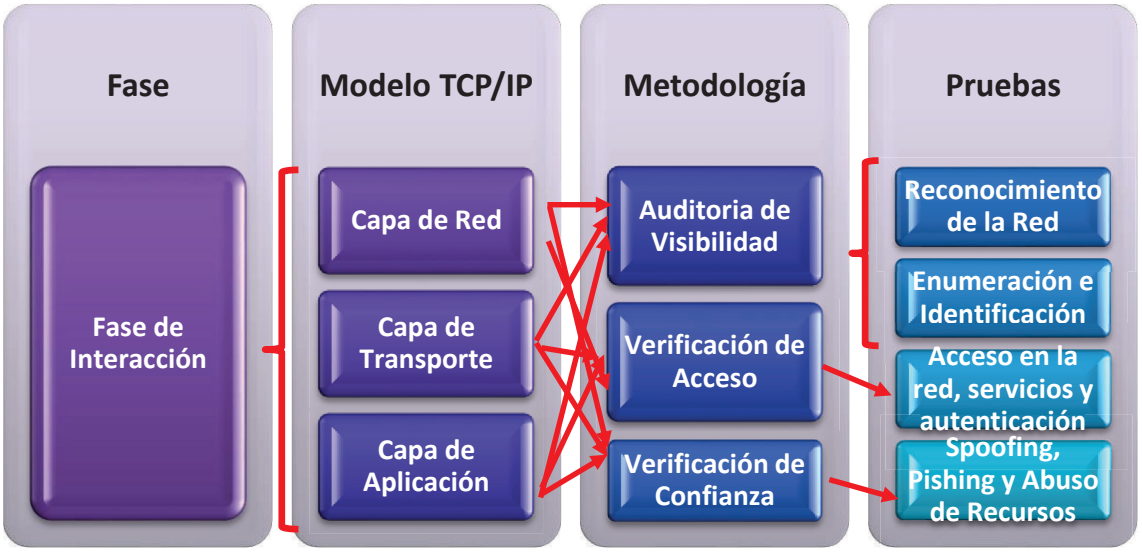


Figura 2-17 Fase de Interacción del Toolkit de pruebas de intrusión

2.3.3 REQUERIMIENTO 3. *"El Toolkit de pruebas de intrusión debe contar con una fase de investigación que permita verificar la forma de trabajo de los activos, así como las configuraciones de control y servicios en el perímetro de la red"*

El objetivo de esta fase de investigación es verificar la forma en que trabajan los activos, sus configuraciones de control y servicios. Para cumplir con esta fase, se utiliza los módulos Verificación de Configuración y Verificación de Exposición descritos por la metodología OSSTMM V3. El módulo Verificación de Configuración probará la forma de trabajo de los activos y la posibilidad de eludir la seguridad funcional, explotando la configuración de controles de acceso y aplicaciones. El módulo Verificación de Exposición permitirá descubrir información que conduzca al acceso desde múltiples lugares con la misma autenticación. En la figura 2-18 se describe esta relación de la Fase de Investigación con el modelo de referencia TCP/IP, los módulos y pruebas necesarias descritos por OSSTMM V3, para garantizar el cumplimiento del Requerimiento 3.

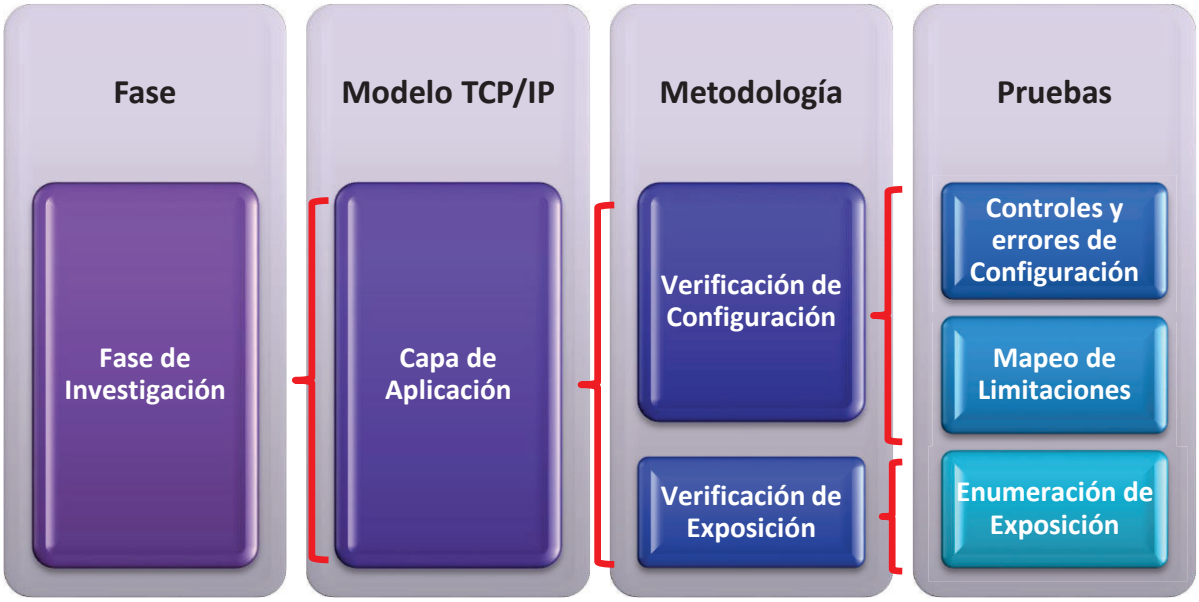


Figura 2-18 Fase de Investigación del Toolkit de pruebas de intrusión

2.3.4 REQUERIMIENTO 4. *"El Toolkit de pruebas de intrusión debe contar con una fase de intervención que ponga a prueba los sistemas y servicios dentro del perímetro de la red"*

El objetivo de esta fase de intervención es verificar si las interrupciones afectan a las respuestas de las pruebas menos invasivas a los sistemas y servicios. Para cumplir con esta fase se utiliza los módulos Auditoría de Privilegios y Validación de Supervivencia descritos por la metodología OSSTMM V3. El módulo Auditoría de Privilegios permite validar el uso correcto de los sistemas de autenticación y autorización, comprobando la posibilidad de ingresos no autorizados y escalamiento de privilegios. El módulo Validación de Supervivencia permite verificar la resistencia y continuidad de los servicios ante cambios excesivos o degradación del servicio. En la figura 2-19 se describe esta relación de la Fase de Intervención con el modelo de referencia TCP/IP, los módulos y pruebas necesarias descritos por OSSTMM V3, para garantizar el cumplimiento del Requerimiento 4.

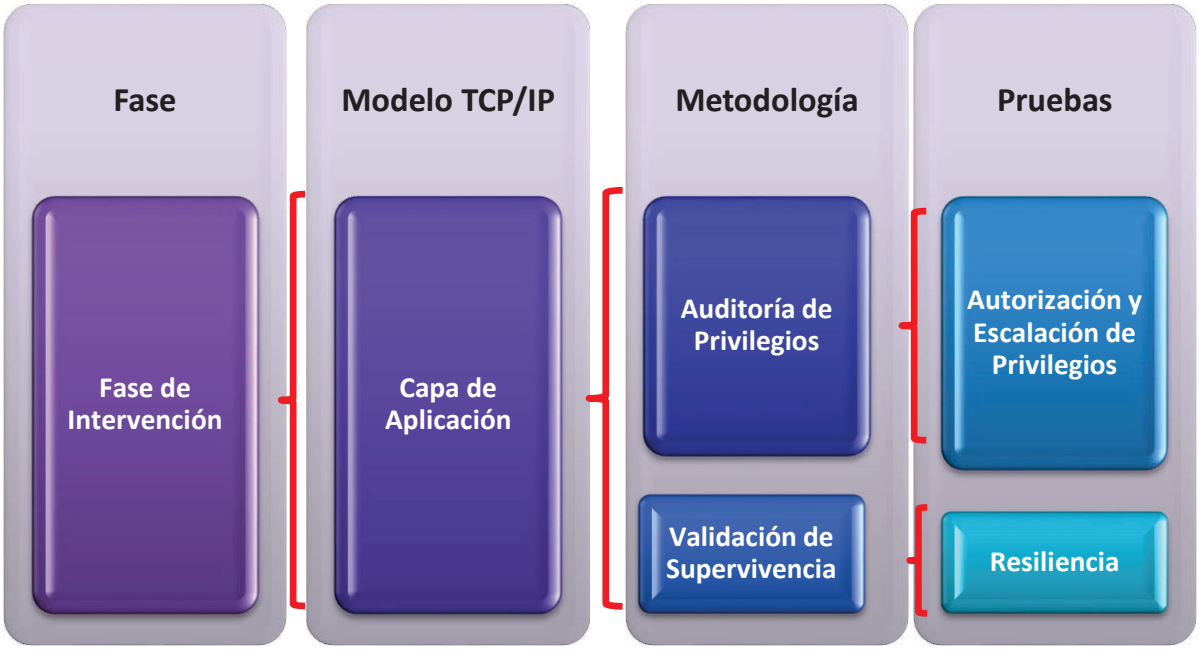


Figura 2-19 Fase de Intervención del Toolkit de pruebas de intrusión

2.4 DISEÑO E IMPLEMENTACIÓN DEL TOOLKIT DE PRUEBAS DE INTRUSIÓN.

Esta sección del proyecto contiene el diseño e implementación del toolkit de pruebas de intrusión en base al procedimiento formal descrito por OSSTMM V3 según los requerimientos detallados previamente en la sección 2.3.

Las pruebas de intrusión se encuentran debidamente ordenadas y detalladas según la fase descrita por la metodología OSSTMM V3. Estas pruebas comprenden la descripción teórica, la herramienta recomendada y el comando o módulo de cada herramienta que permite cumplir con la tarea solicitada.

2.4.1. PROCEDIMIENTO FORMAL DE PRUEBAS DE INTRUSIÓN PARA LA FASE DE INDUCCIÓN

La fase de inducción permite comprender los requisitos de la auditoría, el alcance y las limitaciones en el perímetro de la red. Para ello se desarrollarán las tareas más relevantes de los módulos: Logística y Verificación de la detección activa descritos por la metodología OSSTMM V3.

2.4.1.1 Módulo Logística

Este módulo permite preparar el entorno de prueba de canal, para evitar falsos positivos y negativos que conduzcan a resultados imprecisos en la prueba. En esta sección se desarrollan las siguientes pruebas: Framework, Calidad de la Red y Tiempo

Pruebas: Framework de la red

a) Determinar la ubicación de la propiedad y el propietario de los objetivos del housing[27]. Verificar el dueño de los objetivos de la información de registro de dominio.

Estas pruebas coleccionan información de la propiedad y registro de las direcciones IP del dominio auditado. El titular es la entidad a la que pertenece el dominio. El contacto administrativo contiene a la persona que se le permite administrar (cambiar) la información de dominio. El campo titular contiene el propietario y la ubicación del dominio.

[27] Housing: Es una modalidad de alojamiento web destinado principalmente a grandes empresas y a empresas de servicios web. (Fuente: http://es.wikipedia.org/wiki/Housing).

1. Herramienta: Whois

Esta herramienta permite obtener información detallada del dominio auditado. La sintaxis del comando es:

***Comando: Whois (dominio*)**

2. Herramienta host:

Esta utilidad permite verificar información de los registros Dns relacionados al dominio objetivo. La sintaxis del comando es:

Host –a4 (Dominio)

La opción –a4 muestra información detallada.

3. Herramienta: Dig

Esta utilidad permite verificar información global del dominio auditado, como el host o direcciones IP correspondiente a los registros Dns. La sintaxis del comando es:

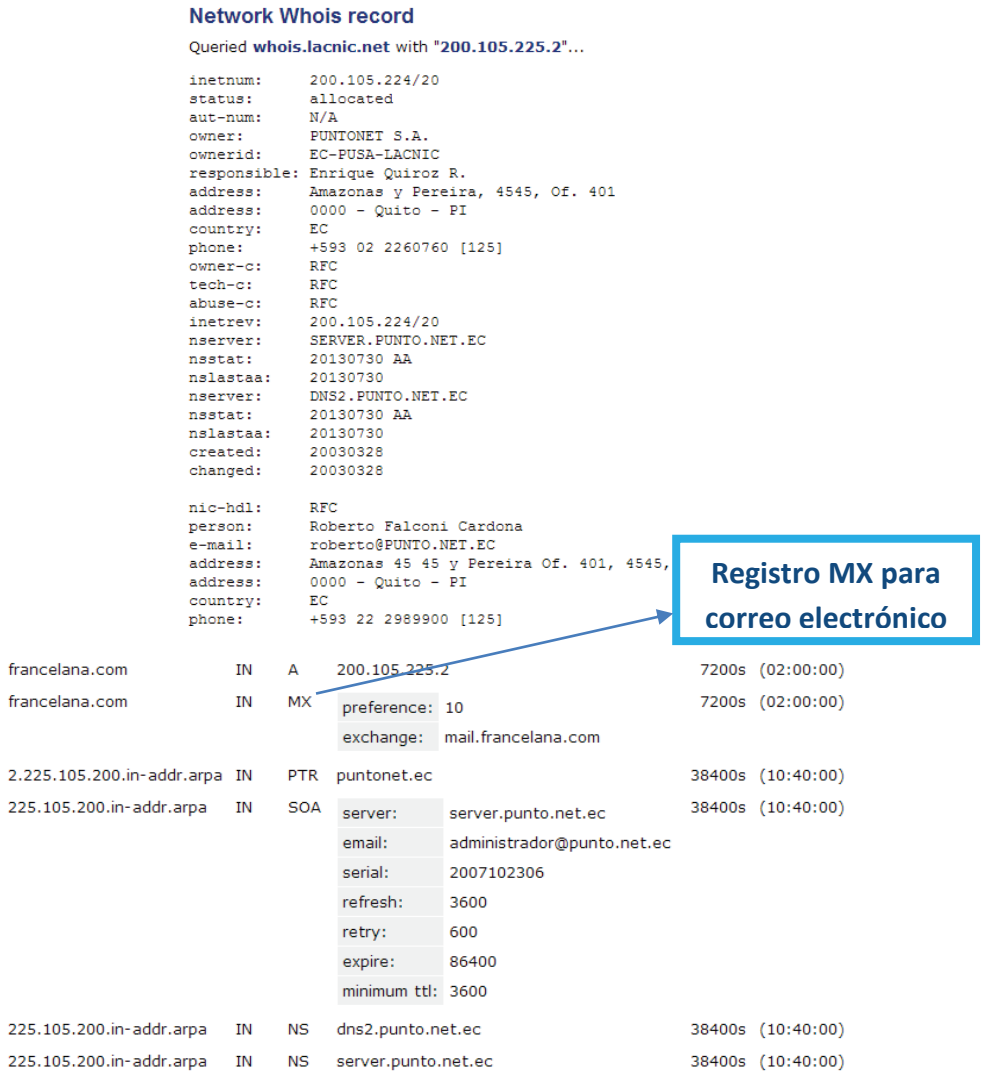
Dig (dominio)

b) Verificar el ISP (s) que proporcione acceso a la red.

El ISP es el proveedor de servicios de internet, que puede proporcionar servicios como correo, servicios de hosting, telefonía IP, etc.

Herramienta: Central Ops

Se utiliza esta herramienta para verificar el Registro de Red asociado al dominio auditado. En la figura 2-20 se verifica el resultado de esta prueba sobre un dominio de ejemplo “francelana.com”, determinando el ISP que proporciona el acceso a la red e información del registro MX usado para el servicio de correos.



Network Whois record

Queried **whois.lacnic.net** with "**200.105.225.2**"...

```
inetnum:     200.105.224/20
status:      allocated
aut-num:     N/A
owner:       PUNTONET S.A.
ownerid:     EC-PUSA-LACNIC
responsible: Enrique Quiroz R.
address:     Amazonas y Pereira, 4545, Of. 401
address:     0000 - Quito - PI
country:     EC
phone:       +593 02 2260760 [125]
owner-c:     RFC
tech-c:      RFC
abuse-c:     RFC
inetrev:     200.105.224/20
nserver:     SERVER.PUNTO.NET.EC
nsstat:      20130730 AA
nslastaa:    20130730
nserver:     DNS2.PUNTO.NET.EC
nsstat:      20130730 AA
nslastaa:    20130730
created:     20030328
changed:     20030328

nic-hdl:     RFC
person:      Roberto Falconi Cardona
e-mail:      roberto@PUNTO.NET.EC
address:     Amazonas 45 45 y Pereira Of. 401, 4545,
address:     0000 - Quito - PI
country:     EC
phone:       +593 22 2989900 [125]
```

Registro MX para correo electrónico

francelana.com	IN	A	200.105.225.2	7200s	(02:00:00)
francelana.com	IN	MX	preference: 10 exchange: mail.francelana.com	7200s	(02:00:00)
2.225.105.200.in-addr.arpa	IN	PTR	puntonet.ec	38400s	(10:40:00)
225.105.200.in-addr.arpa	IN	SOA	server: server.punto.net.ec email: administrador@punto.net.ec serial: 2007102306 refresh: 3600 retry: 600 expire: 86400 minimum ttl: 3600	38400s	(10:40:00)
225.105.200.in-addr.arpa	IN	NS	dns2.punto.net.ec	38400s	(10:40:00)
225.105.200.in-addr.arpa	IN	NS	server.punto.net.ec	38400s	(10:40:00)

Figura 2-20 Registro de Red del Dominio "francelana.com"

Pruebas: Calidad de la red

c) Medición del índice de velocidad y pérdida de paquetes en el alcance de un servicio solicitado en TCP e ICMP, tanto como solicitud de servicio como respuesta. Repetir cada solicitud en sucesión por lo menos 100 veces y anotar el promedio tanto para las solicitudes de servicio como respuestas de paquetes, para cada protocolo.

1. Herramienta Apache Bench

Se utiliza esta herramienta para obtener una estadística de la velocidad y pérdidas de paquetes al Servidor Web en base al protocolo http (puerto 80). La sintaxis del comando es:

ab (opciones) (http://) hostname (:port)/

2. Herramienta ping

Se utiliza esta herramienta para probar los servicios en base al protocolo ICMP. La sintaxis del comando:

Ping –c 100 (hostname o IP)

La opción –c indica el número de ping a enviar.

d) Determinación de envío y recepción de las tasas de paquete para un total de 6 medias por protocolo.

Utilizamos el mismo procedimiento del literal c) utilizando Apache Bench para probar las solicitudes http con una media de 6 veces.

Pruebas: Tiempo

e) Verificar la zona horaria, vacaciones y horarios de trabajo para los diferentes sistemas en el alcance incluyendo a los socios, revendedores y clientes influyentes que interactúan con el alcance.

Esta prueba es necesaria para documentación de la zona horaria de los trabajadores y los sistemas auditados.

Zona horaria del Ecuador: UTC-5.

Horario de trabajo: 8:00 am – 17:00 pm

f) Identificar el Time To Live (TTL) al Gateway y los objetivos.

El Time to Live es un campo en la estructura del paquete IP que se relaciona con el número de nodos que puede atravesar un paquete antes que sea descartado. Se lo usa para prevenir que los paquetes estén circulando de forma indefinida en la red. Cada vez que un paquete atraviesa un router este decrementa el valor de TTL en uno. Generalmente este valor está configurado de forma estándar por los sistemas TCP/UPD, teniendo como referencia máxima el valor de 255. Para sistemas Windows suele ser 128, mientras que para Unix Linux es 64 o 255.

Herramienta Ping.

Permite obtener el valor TTL, considerando que si llega a ser 1 o 0, el router descartaría este paquete enviando un mensaje ICMP[28] al origen de que el destino se encuentra "muy alejado".

g) Asegurar que el reloj del analista esté en sincronía con el tiempo de los objetivos.

Esta prueba asegura que tanto el analista como los objetivos, estén trabajando bajo la misma línea de tiempo, y no haya errores en los tiempos de las pruebas realizadas.

2.4.1.2 Módulo Verificación de la Detección Activa

Este módulo permite determinar los controles activos y pasivos a fin de detectar la intrusión para filtrar o rechazar los intentos de prueba. En esta sección se desarrollarán las siguientes pruebas: Filtrado y Detección Activa.

[28] ICMP es el Protocolo de Mensajes de Control de Internet utilizado para notificación de errores del Protocolo de Internet(IP). (Fuente: http://es.wikipedia.org/wiki/Internet_Control_Message_Protocol)

Pruebas: Filtrado

h) Probar si los datos de red entrantes y salientes o comunicaciones a través de Web, mensajería instantánea, chat, basados en la web o foros, e-mail, se monitorean o se filtran por una parte autorizada para la retransmisión de materiales inadecuados, inyecciones de código, contenido malicioso, conducta impropia, así como el registro y tiempo de respuesta.

Las políticas de filtrado están definidas a nivel de control de contenido web por el servidor Squid Proxy y el Firewall de la red con el Sistema Iptables. Las verificaciones en esta sección se desarrollarán de forma manual, validando las configuraciones de control de contenido y del firewall mediante la herramienta vía web denominada "Webmin".

1. Verificación de Control de Contenido con Squid Proxy Server

El servidor Squid Proxy usa las conexiones en el puerto 3128 filtrando el tráfico http para permitir o denegar la navegación web. Este servidor maneja reglas de control de acceso, en orden de prioridad para limitar las conexiones http. En la figura 2-21 se aprecia la interfaz gráfica del Servidor Squid Proxy vía Webmin.

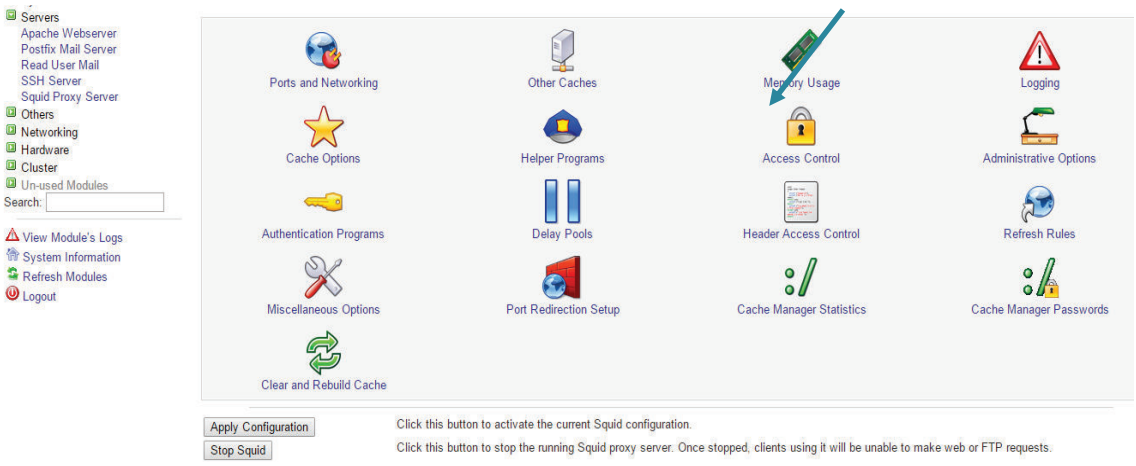


Figura 2-21 Interfaz Gráfica del Servidor Squid Proxy

2. Verificación del Firewall de la red con el Sistema Iptables del servidor

El servidor Firewall utiliza componentes como puertos, URL's para realizar filtrado de comunicaciones entrantes y salientes. El firewall de la red está implementado bajo el software iptables de Linux.

Pruebas: Detección Activa

i) Verificar las respuestas activas a las pruebas de los sistemas y servicios. Esto podría ser notificaciones legibles humano o máquina, respuestas de paquetes, viajes silenciosos de alarma, o similares.

Estas pruebas permiten identificar los puertos abiertos y relacionarlos a los sistemas y servicios habilitados dentro del perímetro de la red.

Herramienta Nmap

Esta herramienta de exploración de redes, permitirá realizar una auditoría de puertos abiertos para los sistemas objetivos en el perímetro de red.

Comando: Nmap (Opciones) Objetivo

Opción –T: Indica el tiempo en mostrar los resultados de la prueba. Tiene una escala desde 0 (Lento y Fiable) hasta 5 (Muy Rápido e Impreciso).

Opción –A: Muestra información detallada, como versión, sistema operativo, etc.

Opción –v: Indica que se incrementa el nivel de detalle.

Opción –sV: Determina el tipo de servicio y su versión a través de una consulta a su base de datos "nmap-service-probes" donde se puede consultar y reconocer los servicios.

Opción –s0: Valida los protocolos soportados por el servidor o computador personal.

2.4.2. PROCEDIMIENTO FORMAL DE PRUEBAS DE INTRUSIÓN PARA LA FASE DE INTERACCIÓN

La fase de interacción con los objetivos permite determinar el ámbito de aplicación de la auditoría. Para ello se desarrollarán las tareas más relevantes de los módulos: Auditoria de Visibilidad, Verificación de Acceso y Verificación de confianza descritos por la metodología OSSMMM V3.

2.4.2.1 Módulo Auditoría de Visibilidad

Este módulo permite enumerar e indexar los objetivos en el alcance, mediante la interacción directa e indirecta con los sistemas. En esta sección se desarrollarán las siguientes pruebas: Survey o Reconocimiento de la Red[29] y Enumeración.

Pruebas: Survey de la Red

a) Determinar el perímetro del segmento(s) de la red del objetivo y el vector que será probado.

El perímetro del segmento de red sobre el que se efectuarán las pruebas, está integrado por los siguientes elementos del perímetro de la red:

1. Servidor Web/Ftp/Dns
2. Servidor Firewall / Control de Contenidos.

[29] Survey es un término aplicado a una encuesta o recopilación de información a base de preguntas que se formulan a un conjunto o muestra de personas previamente escogidas según las necesidades del estudio. (Fuente: perugrafico.com/diccionario-de-marketing-y-tecnologia.php)

b) Usar olfateo de la red para identificar emanación de protocolos desde respuestas de servicio de red o solicitudes donde son aplicables. Por ejemplo, NetBIOS, ARP, SAP, NFS, BGP, OSPF, MPLS, RIPv2, etc.

Herramienta Wireshark

Esta herramienta permitirá recoger y analizar las respuestas de las solicitudes de servicios a los elementos del Perímetro de red, a fin de determinar los protocolos y servicios activos. La interfaz gráfica de Wireshark se muestra en la Figura 2-22. En la opción "Capture Options" de la Figura 2-23 se visualiza los filtros para el escaneo, como selección de las interfaces, el modo de escaneo de forma promiscua[30] o en todas las interfaces (tráfico de red del host).

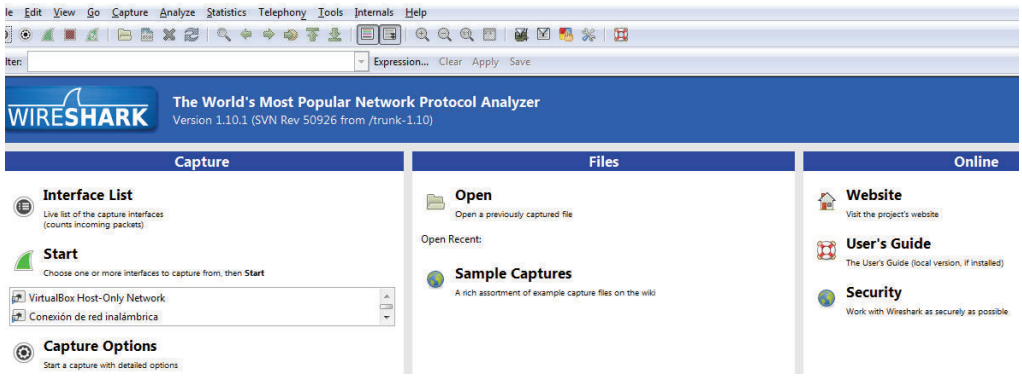


Figura 2-22 Interfaz gráfica de la Herramienta Wireshark

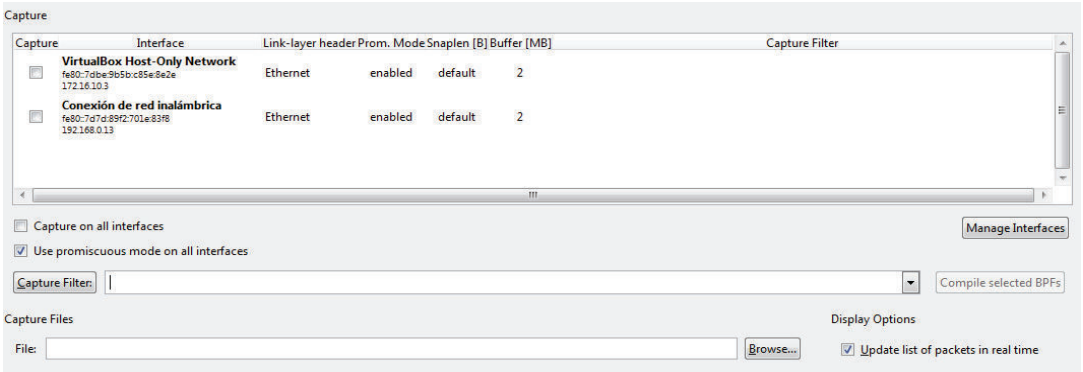


Figura 2-23 Filtros de la Herramienta Wireshark

[30] Escaneo promiscuo: Monitoreo del tráfico en todas las redes.

El resultado del escaneo indica la siguiente información:

- No. Número de paquetes enviados o recibidos
- Time: El tiempo que dura en transmitirse el paquete
- Source: Especifica dirección IP Origen
- Destination: Especifica dirección IP Destino
- Protocol: El protocolo utilizado en la transmisión
- Length: Tamaño del paquete
- Info: Información del paquete.

c) Consultar todos los servidores de nombres y los servidores de nombres del proveedor de ISP o de hosting, si está disponible, para corresponder a registros A, AAAA y PTR, así como la capacidad para realizar transferencias de zona para determinar la existencia de todos los objetivos en la red y cualquier redundancia relativa, balanceo de carga, caching, proxy y hosting virtual.

Esta prueba no se puede desarrollar para nuestro dominio local “iza.sanhueza.ec” ya que no está publicado en Internet. Sin embargo se detalla la herramienta que se puede utilizar para cumplir con este apartado con dominios públicos.

Herramienta Maltego [W4]:

Esta herramienta permite obtener información de los servicios y registros asociados al dominio auditado. En la figura 2-24 se visualiza la interfaz gráfica de Maltego, la cual tiene varias opciones para determinar información de los servicios como: Sistema Autónomo, Servidores DNS, Dominios, Direcciones IPv4, Registros MX, Registros tipo A, Registros NS, etc.

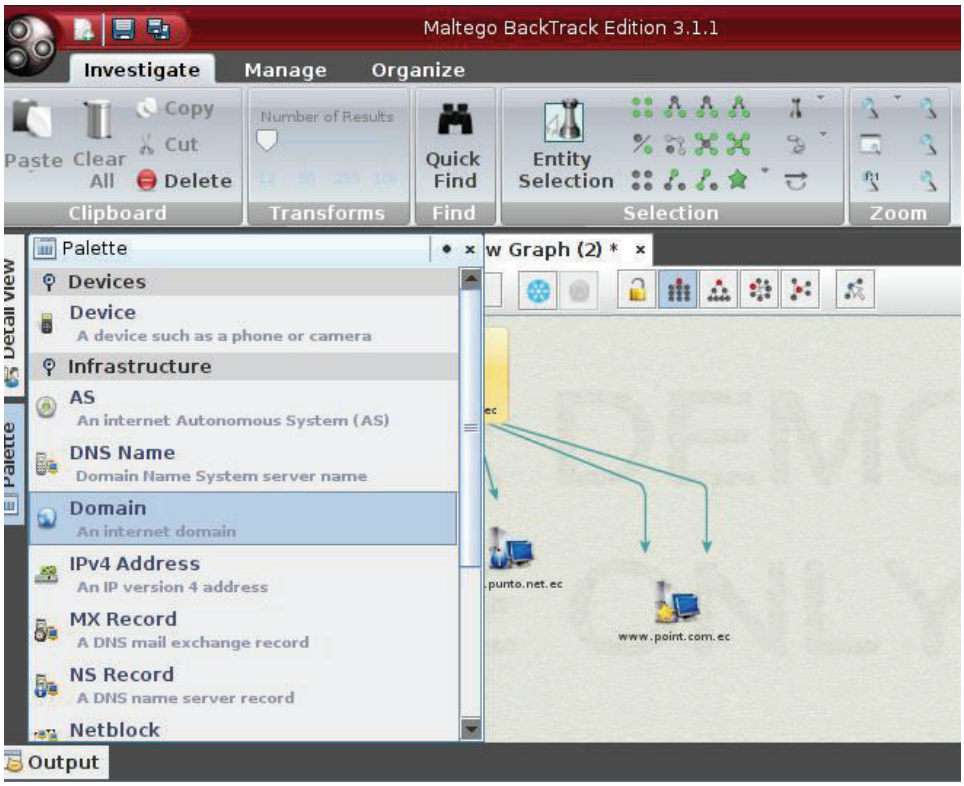


Figura 2-24 Interfaz Principal de la Herramienta Maltego

d) Verificar las solicitudes de broadcast y las respuestas de todos los objetivos.

Los broadcast son una forma de transmisión simultánea desde un nodo hacia algunos o todos los dispositivos en una red. Con ello se evita el uso de transmisiones uno a uno y el analista de seguridad puede recabar información transmitida al segmento de red.

Herramienta Wireshark

Para capturar el tráfico de broadcast, hacemos uso del filtro basado en una cadena string llamada “Broadcast”.

e) Verificar y examinar el uso de tráfico y protocolos de enrutamiento para todos los objetivos.

Determinar el tipo de tráfico y protocolos de enrutamiento, permitirá al analista de seguridad verificar los servicios, puertos que están siendo usados por los sistemas en el perímetro de red.

Herramienta Nmap

Argumento –s0 Permite realizar una exploración de protocolos IP.

f) Verificar las respuestas ICMP de los tipos ICMP 0-255 y códigos ICMP 0-2 de todos los objetivos.

ICMP Internet Control Message Protocol es un protocolo de control y gestión de errores del protocolo IP. Se usa para enviar mensajes de error, notificando el servicio indisponible o un host inalcanzable. Utiliza los mensajes de petición y respuesta Echo ICMP, para determinar si un host está disponible, el tiempo empleado y cantidad de host que atraviesa. Los tipos de paquetes ICMP más importantes se ilustran en la Figura 2-25[31]. Cuando el tipo de paquete indica el valor X de destino inalcanzable, entonces el tipo de código representa el error concreto.

Herramienta Nmap

Se utiliza esta herramienta con los argumentos PE, PP y PM para verificar las respuestas de los sistemas en el perímetro de red, ante determinados tipos de paquetes y código ICMP.

Argumento –PE

Realiza un requerimiento Ping Echo ICMP, determinando los servicios y puertos activos, correspondiente al tipo ICMP 15.

[31] Tipos de paquetes ICMP. Fuente: http://neo.lcc.uma.es/evirtual/cdd/tutorial/red/icmp.htmL.

Argumento –PP y PM

Permite verificar información de huella de tiempo y máscara de red, correspondientes a los tipos ICMP 13 Y 18. Es muy útil, en casos que el administrador de la red realiza un bloqueo de ping con ICMP tipo 15, pero no se percata que se pueden realizar las solicitudes tipo 13 y 18 para obtener de igual forma otro tipo de información.

0	Echo Reply (Respuesta de Eco)
3	Destination Unreachable (Destino Inaccesible)
4	Source Quench (Disminución del Tráfico desde Origen)
5	Redirect (Redireccionar - cambio de ruta)
8	Echo (Solicitud de Eco)
11	Time Exceeded (Tiempo excedido para un datagrama)
12	Parameter Problem (Problema de Parámetros)
13	Time Stamp (Solicitud de Marca de tiempo)
14	Time Stamp Reply (Respuesta de Marca de tiempo)
15	Information Request (Solicitud de Información) – obsoleto
16	Information Reply (Respuesta de Información) – obsoleto
17	Address mask (Solicitud de máscara de dirección)
18	Address mask Reply (Respuesta de máscara de dirección)

0	No se puede llegar a la red
1	No se puede llegar al host o aplicación destino
2	El destino no del protocolo solicitado
3	No se puede llegar al puerto destino o la aplicación destino no está libre
4	Se necesita aplicar fragmentación, pero el flag correspondiente indica lo contrario
5	La ruta de origen no es correcta
6	No se conoce la red destino
7	No se conoce el host destino
8	El host origen está aislado
9	La comunicación con la red destino está prohibida por razones administrativas
10	La comunicación con el host destino está prohibida por razones administrativas
11	no se puede llegar a la red destino debido al tipo de servicio
12	No se puede llegar al host destino debido al tipo de servicio

Figura 2-25 Tipos de paquetes ICMP. **Tipo Código**

g) Verificar los nombres por defecto y nombres de comunidades SNMP en uso que estén de acuerdo con las implementaciones de todas las versiones de SNMP.

SNMP Simple Network Managment Protocol es un protocolo de capa aplicación que ayuda al intercambio de información de administración entre dispositivos de red.[32] Es utilizado por varias herramientas de monitoreo como Nagios, Mrtg para verificar y/o monitorear el estado de un dispositivo o servicio. Trabaja en los puertos UDP 161/162. Las herramientas que se utilizarán para filtrar el tráfico snmp son: Wireshark y Nmap.

Herramienta Wireshark

Se activa el filtro de escaneo de acuerdo a la cadena string llamada “SNMP”.

Herramienta Nmap

El argumento –Su permite filtrar la búsqueda por puertos UDP. Como SNMP trabaja con los puertos UDP 161 y 162 debería mostrar la información relativa a este protocolo.

h) Verificar las respuestas de los objetivos al seleccionar los puertos con expiración TTL de vencimiento a menos de 1 y 2 saltos de los objetivos. Por ejemplo:

TCP 8, 22, 23, 25, 80, 443, 445, 1433

UDP 0, 53, 139, 161

ICMP T00: C00, T13: C00, T15: C00, T17: C00

ICMP como protocolo de gestión de errores y control de IP mantiene entre los tipos de mensajes ICMP el valor de 11 que indica tiempo excedido. Este tipo de mensaje ICMP 11 hace referencia cuando el valor de time to live (TTL) ha llegado a 0, por lo

[32] Definición de SNMP. Protocolo Simple de administración de redes. Fuente: http://es.wikipedia.org/wiki/Simple_Network_Management_Protocol.

cual el router envía un mensaje al origen de tiempo excedido. En la figura 2-26 se aprecia el formato del mensaje ICMP tipo 11.

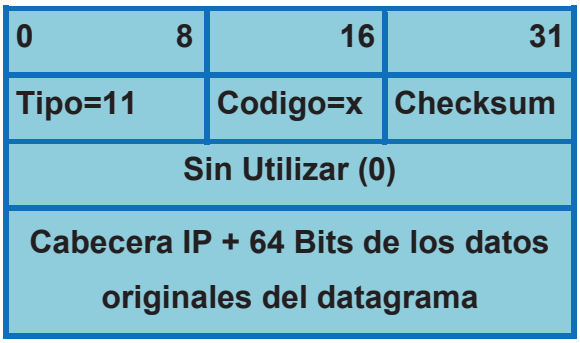


Figura 2-26 Mensaje ICMP Time Exceeded LL

Código 0= TTL excedido en tránsito

Código 1= TTL excedido durante el re ensamblaje

Herramienta Ping

Se utiliza esta herramienta con la opción –t para establecer el time to live (TTL) hacia los sistemas del perímetro de red.

i) Trazar la ruta de los paquetes ICMP a todos los destinos.

El TTL es una característica incluida en la cabecera del paquete IP, para prevenir los loops[33] de los paquetes IP dentro de una red. Cada nodo al recibir el paquete IP, decrementa en uno el valor del TTL, de tal forma que si llega a ser cero el paquete será descartado por el router y un mensaje ICMP del tipo "TTL excedido en tránsito" será remitido al origen de la petición. Traceroute utiliza este comportamiento, manipulando el contador TTL, de tal forma que cada nodo o router en la trayectoria hacia el destino generará este mensaje de error y revelará su dirección IP. La versión de Windows tracert.exe es la que utiliza pings ICMP echo, mientras que la versión de Unix Linux usa Datagramas UDP.

[33] Un Bucle de enrutamiento o routing Loop ocurre cuando los encaminadores o routers disponen de una información acerca de la red y en lugar de enviar el tráfico a su destino, se pasan los paquetes entre ellos creyendo que el otro router sabrá el camino. Fuente: http://es.wikipedia.org/wiki/Bucle_de_enrutamiento.

Herramienta Traceroute:

Se utiliza esta herramienta para verificar la traza de los paquetes ICMP hacia los objetivos, mediante la manipulación del campo TTL del protocolo IP.

Comando: traceroute {opciones} {objetivo}

Opciones:

-h: Número de saltos máximo.

-j: Lista de hosts.

-w: Tiempo de espera.

-R: Seguir la ruta de retorno (Solo IPv6)

-S: Dirección origen (Solo IPv6)

-4 ó -6: Forzar usando IPv4 o IPv6 respectivamente.

j) Trace la ruta de los paquetes TCP a todos los objetivos para los puertos SSH, SMTP, HTTP, HTTPS.

Herramienta: Traceroute

Se utilizará esta herramienta con la **Opción -p** para definir el puerto destino a probar contra los sistemas del perímetro de red.

k) Trace la ruta de los paquetes UDP a todos los objetivos para los puertos DNS y SNMP.

Herramienta Traceroute

-P Activa el escaneo con el protocolo UDP.

-p Define el puerto

l) Identificar el número de predictibilidad de secuencia TCP ISN para todos los objetivos [T1].

TCP al ser un protocolo de conexión confiable, y robusto utiliza los números de secuencia Inicial TCP ISN para mantener esta característica. TCP en su etapa de establecimiento de la conexión utiliza una comunicación de 3 vías, en donde se envían números de secuencia TCP indicando el orden de los datos dentro del flujo de bytes, y con ello reconstruir el segmento en el Receptor. Esta comunicación se detalla en la figura 2-27.

1. El cliente inicia la conexión con la bandera SYN activada, por lo cual envía el número de Secuencia inicial x (TCP ISN) del remitente.
2. El servidor responde con las banderas SYN/ACK activadas, si la conexión está abierta, o a su vez con las banderas RST/ACK activadas, si la conexión está cerrada y no responderá nada si está filtrada la conexión. Si la respuesta es un SYN/ACK el receptor envía el ack de la secuencia inicial: x+1, y su propio número de secuencia y.
3. El cliente responde con la bandera ACK si la conexión resultó abierta, con su correspondiente ack del número de secuencia del receptor: y+1 y de sí mismo: x+1.

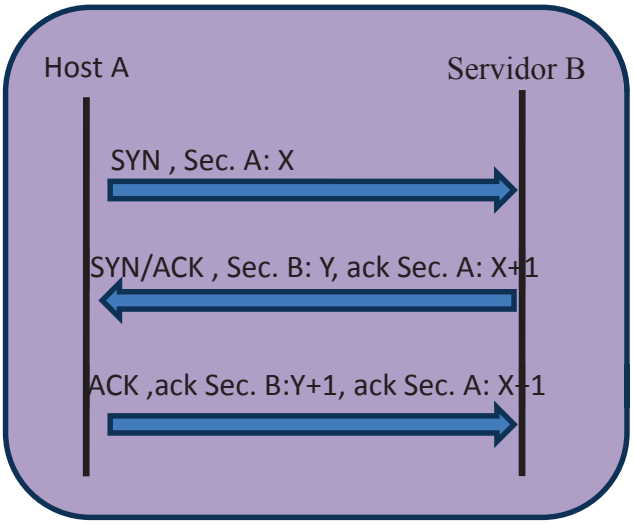


Figura 2-27 Comunicación de 3 vías de TCP para establecimiento de conexión.

Herramienta Wireshark

Este analizador de tráfico se utiliza para verificar los números de secuencia usados por los paquetes hacia los objetivos del perímetro de red.

Pruebas: Enumeración

m) Buscar los grupos de noticias, foros, IRC, mensajería instantánea, P2P, VoIP y comunicaciones basadas en web para conectar la información del objetivo y poder determinar los sistemas gateway de salida y direccionamiento interno.

La información pública de los objetivos auditados es muy importante para empezar un análisis previo de los puntos de acceso y lograr identificar sus vulnerabilidades. Para cumplir este requerimiento, se hace uso de la herramienta Maltego, a fin de obtener información pública relativa al sitio web del dominio auditado**,** tales como cuentas de correo electrónico, plataforma del sitio web, nombres y registros DNS asociados. También se utilizará el comando host para verificar el Sistema Gateway relacionado al sitio web local. En el proyecto actual no se puede cumplir esta prueba ya que el dominio auditado “iza.sanhueza.ec” no es público a nivel de internet.

n) Examinar el código fuente de la aplicación destino basada en web y scripts para determinar la existencia de objetivos adicionales en la red.

El código fuente de una aplicación Web, permite coleccionar información relevante a la estructura del Contenido, títulos, e información general de la aplicación. En base a esta información se pueden localizar exploits que permitan vulnerar el código fuente de la aplicación.

Herramienta Firebug de Firefox

Se utiliza esta herramienta para analizar el código fuente del sitio web local www.iza.sanhueza.ec. Se activa la opción HTML del panel de Herramientas (Figura 2-28). El código fuente, está compuesto por dos secciones principales:

1. La cabecera: <head>
2. El cuerpo: <bodyclass>

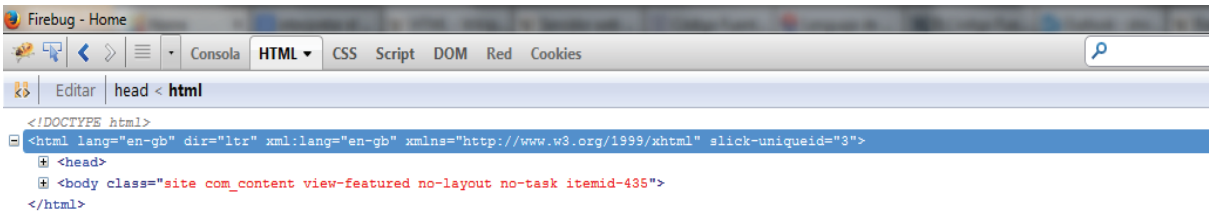


Figura 2-28 Herramienta Firebug de Firefox

o) Buscar logs web y de intrusión para el sistema de la red destino.

Los logs son registros que nos permiten llevar una bitácora de actividades en los servicios/dispositivos de un sistema. Estos registros son importantes no solo para para el seguimiento de actividades, sino también de errores que un analista de seguridad puede utilizar para identificar vulnerabilidades en los sistemas objetivos. Los logs se registran en archivos específicos para cada servicio del perímetro de red, que serán detallados en el capítulo 3 sección 3.1.2.1 Pruebas Enumeración literal o).

Herramienta: Tail

Se utilizará esta herramienta para examinar los logs web y de intrusión para los sistemas objetivos en el perímetro de red.

Opción: -f Este argumento activa la opción de mostrar la información del archivo en tiempo real.

p) Verifique todas las respuestas de las solicitudes de paquetes UDP a los puertos 0-65535.

Herramienta Nmap

Opción: -sU –p1-65535 Permite realizar un escaneo UDP al rango de puertos indicados.

q) Verifique las respuestas de las solicitudes de paquetes UDP de los puertos fuente 0, 53, 139, y 161 a los puertos destino 0, 53, 69, 131, y 161.

Cuando se configuran servicios Dns, Ftp, Web atrás de los Sistemas Firewall y/o detección de intrusos[34], en el Firewall se habilita el filtro para permitir el tráfico entrante de los puertos correspondientes a estos servicios. Un analista de seguridad puede aprovechar esta vulnerabilidad para falsificar su puerto origen por uno de los puertos permitidos en el Firewall y con ello se vería comprometida la comunicación entrante a los sistemas del perímetro de red.

Herramienta Nmap

Opción –sU → Habilita el escaneo UDP.

Opción –source-port (Número de Puerto) →Permite Falsificar el puerto de origen para aprovechar la vulnerabilidad donde los sistemas Firewall IDS, permiten el tráfico entrante en un puerto específico.

Opción –p U:(Número de Puerto) –T:(Número de Puerto) S:(Número de Puerto) → Permite establecer el escaneo hacia los puertos establecidos en base a peticiones UDP con la opción U, peticiones TCP con la opción T, y peticiones SCTP[35] con la opción S.

r) Verificar las respuestas a las solicitudes de paquetes UDP con sumas de comprobación (checksums) incorrectas a todos los puertos descubiertos en el literal q y para los puertos 0, 53, 69, 131 y 161.

El checksum es un mecanismo que permite la detección de errores en las distintas capas del modelo TCP/IP. Para la capa de transporte con paquetes TCP/UDP el

[34] Sistema de Detección de Intrusos es una herramienta de seguridad que intenta detectar o monitorizar los eventos ocurridos en un determinado sistema informático o red informática en busca de intentos de comprometer la seguridad de dicho sistema. Fuente: http://www.maestrosdelweb.com/snort/

[35] http://es.wikipedia.org/wiki/Stream_Control_Transmission_ProtocolSCTP es un protocolo de comunicación de capa transporte, una alternativa a los protocolos de transporte TCP y UDP pues provee confiabilidad, control de flujo y secuenciación como TCP. Sin embargo, SCTP opcionalmente permite el envío de mensajes fuera de orden y a diferencia de TCP, SCTP es un protocolo orientado al mensaje (similar al envío de datagramas UDP).

checksum cubre tanto la cabecera como los datos, mientras que el checksum para la cabecera IP únicamente cubre el contenido de la propia cabecera IP, delegando la funcionalidad de verificación de errores en los datos a capas superiores. El cálculo del checksum usa la operación de complemento a uno de 16 bits de la pseudo-cabecera, cabecera, datos del paquete TCP/UDP, y ceros hasta completar el múltiplo de 16. La pseudo-cabecera tiene los campos: Dirección IP Origen, Dirección IP Destino, Protocolo y Longitud de los paquetes UDP. En la figura 2-29 se muestra la pseudo-cabecera y el resto del paquete UDP. En los sistemas Firewall/IDS no suelen procesar el campo de checksum de los paquetes TCP/UDP, por lo cual esta vulnerabilidad puede ser aprovechada por el analista de seguridad para intentar filtrar las conexiones hacia los servidores detrás del Firewall.

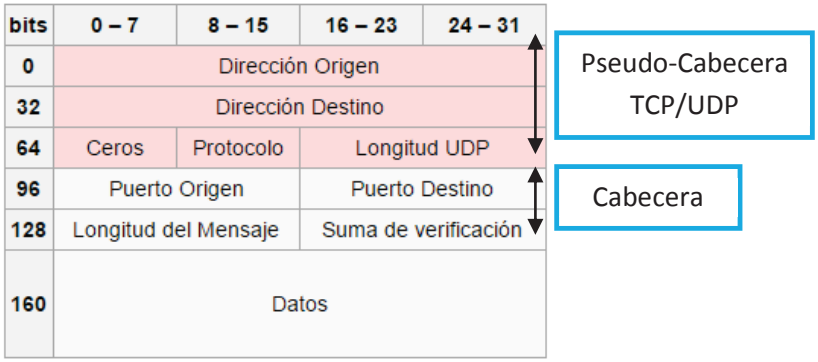


Figura 2-29 Campos para el cálculo del Checksum en paquetes TCP/UDP

Herramienta Nmap

Opción –Su –p U:(Número de puerto) T:(Número de puerto) S:(Número de puerto) –badsum → Envía paquetes con un checksum incorrecto para los tipos de paquetes: TCP con la opción P, UDP con la opción U, y SCTP con la opción S.

s) Verificar las respuestas de solicitud de servicio a los puertos malware de acceso remoto UDP común y contemporáneo.

El Malware son aplicaciones que tratan de violar la seguridad de un equipo/sistema mediante el espionaje de las pulsaciones del teclado. El objetivo es obtener accesos que posteriormente serán enviados a un servidor Web del atacante a través de la

misma aplicación (Spyware) que permite la conexión. Estas aplicaciones pueden estar al interior de archivos descargados en la red, e incluye el grupo de virus, gusanos y troyanos. Los puertos UDP comunes y contemporáneos en relación a malware de acceso remoto, se detallan en el Anexo 6, en referencia a la información registrada por la entidad IANA[36], los sistemas UNIX/Linux, y sistemas Windows respectivamente.

Herramienta: Nmap

Opción: -Su Establece el escaneo Udp

Opción: -p U:(Número de puerto) Asigna los puertos Udp a escanear

t) Verificar las respuestas a las solicitudes de paquetes TCP SYN a los puertos 0-65535.

En base a la fase de establecimiento de la conexión TCP de 3 vías de la figura 2-27, en la primera etapa el Origen envía el paquete TCP con la opción SYN activada, luego el Receptor acusa esta conexión con la opción SYN/ACK si el puerto está abierto, caso contrario envía un RST (reset) confirmando que el puerto está cerrado, o incluso no envía nada si el puerto esta filtrado por algún sistema Firewall/Ids intermedio. Finalmente el Origen envía un ACK confirmando la recepción correcta del paquete recibido.

Herramienta: Nmap

Opción: -Ss Envía paquetes TCP con la bandera SYN activada**.** Esta opción se caracteriza por no completar la conexión TCP de 3 vías, ya que solo espera recibir como respuesta el paquete SYN/ACK o RST del receptor para indicar que el puerto está abierto o cerrado. Si no recibe ninguna respuesta debido a errores de tipo ICMP no alcanzable (tipo 3, códigos 1, 2, 3, 9, 10 o 13) después de realizar

[36] IANA Internet Assigned Numbers Authority es la entidad que supervisa la asignación global de direcciones IP, sistemas autónomos, servidores raíz de nombres de dominio DNS y otros recursos relativos a los protocolos de Internet. Fuente: http://es.wikipedia.org/wiki/Internet_Assigned_Numbers_Authority.

retransmisiones, el puerto se marca como filtrado. El núcleo del sistema donde se ejecuta Nmap rompe la conexión establecida mediante un paquete RST en lugar de enviar el paquete ACK que completaría el saludo TCP.

u) Verificar las respuestas de las solicitudes de servicio TCP a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

Herramienta: Nmap

Opción –p Establece el puerto a escanear. Por defecto Nmap escanea las solicitudes de servicio TCP.

v) Verificar las respuestas de un TCP ACK con un puerto origen de 80 a los puertos 3100-3150 y 10001-10050, 33500-33550 y 50 puertos aleatorios por encima de 35000.

Herramienta Nmap

Opción -sA (sondeo TCP ACK). A diferencia de otros tipos de sondeo TCP, este en particular no determina puertos abiertos, ni abiertos/filtrados. El objetivo es mapear las reglas de algún sistema Firewall/IDS intermedio, para determinar si trabajan en modo de inspección de estados y cuáles puertos están filtrados.

La respuesta para puertos abiertos o cerrados será un paquete RST, mediante la designación no filtrada. Esto confirma que responden ante los paquetes ACK, pero no se puede determinar si están abiertos o cerrados. Mientras que los puertos que no respondan o que envíen mensajes de error ICMP en respuesta (tipo 3, código 1, 2, 3, 9, 10, o 13), se marcan como filtrados.

w) Verificar respuestas de los fragmentos TCP SYN a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

La fragmentación de los Paquetes TCP tiene el objetivo de dividir la cabecera TCP para complicar el análisis en los sistemas de filtrado de los Firewall/IDS. Esta vulnerabilidad puede ser aprovechada por el analista de seguridad para que los fragmentos puedan pasar desapercibidos e introducirse en la red con objetivos de manipulación y/o explotación de servicios. Generalmente los sistemas de filtrado con Firewall/Ids no implementan el proceso de fragmentación y Re ensamblado de los datagramas, y por ello no se detecta ni previene este tipo de actividades por origen del enmascaramiento que la fragmentación produce. Una solución es que los sistemas de prevención y detección implementen esta técnica, pero esto involucra costos adicionales a nivel de recursos (tiempo, proceso y memoria).

Herramienta Nmap. Opción –f Permite que el escaneo solicitado de tipo SYN[37], FIN[38], XMAS[39] o NULL[40] use paquetes IP fragmentados. La ventaja de este sondeo es que pueden atravesar algunos cortafuegos que no hagan inspección de estados o ruteadores que no hagan filtrado de paquetes[41].

x) Verificar las respuestas de todas las combinaciones de las banderas TCP a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

En el escaneo TCP también se puede especificar banderas TCP arbitrarias, a fin de aumentar las probabilidades de traspasar los sistemas Firewall/IDS. Estas banderas

---

[37] Escaneo SYN (–sS): Realiza un scan de la conexión TCP de forma incompleta, enviando desde el origen el paquete con la bandera SYN activada, y según ello el receptor contesta con un paquete SYN/ACK si la conexión está abierta o a su vez con un RST indicando la conexión cerrada, para que finalmente el origen cierre abruptamente la conexión con un paquete RST.

[38] Escaneo FIN (-sF): Realiza un scan con un paquete FIN vacío activando el bit TCP FIN, como prueba para determinar si está escuchando o no la conexión en un determinado puerto. Si recibe como respuesta un RST entonces el puerto está cerrado, Si no se recibe ninguna respuesta el puerto se marca como cerrado/filtrado, y si solo se marca como filtrado se recibió un error ICMP no alcanzable (tipo 3, código 1, 2, 3, 9, 10, o 13).

[39] Escaneo XMAS (-sX): Activa los bits FIN, PSH, y URG flags, con el mismo objetivo del escaneo FIN.

[40] Escaneo NULL (-sN): No fija ningún bit (la cabecera de banderas TCP es 0)

[41] Sección -sN; -sF; -sX (sondeos TCP Null, FIN, y Xmas). Fuente: http://nmap.org/man/es/man-port-scanning-techniques.html.

detalladas en la Figura 2-30 se incluyen en la cabecera de los paquetes TCP, y en total son 6 campos que pueden ser 1 o 0 según estén activadas o no respectivamente.

SYN (Synchronize): Solicita iniciar la conexión en una comunicación TCP de 3 vías confirmando el establecimiento de la misma para la sincronización de los datos.	**URG (Urgent):** Define un bloque de datos como urgente, mediante un puntero que apunta a donde terminan estos datos urgentes
ACK (Acknowledgment): Mediante esta confirmación establece la conexión en una comunicación TCP.	**PSH (Push):** Indica envío inmediato de los datos a la aplicación sin ponerlos en un buffer o esperar más datos.
RST (Reset): Indica reiniciar la conexión por motivos de paquetes corruptos y SYN duplicados, retardados, etc. Los paquetes con flag RST no solicitados son ignorados.	**FIN (Finalize):** Solicita finalizar la conexión en una comunicación TCP de 3 vías. Este campo indica el estado de finalización de la conexión.

Figura 2-30 Banderas del paquete TCP

Herramienta: Nmap [W3]

Opción --scanflags: Realiza el sondeo con la especificación de banderas TCP arbitrarias. El sondeo tipo TCP SYN se usa por defecto cuando no se especifica ningún tipo base. Las combinaciones posibles para las banderas que maneja Nmap son:

- Ninguna bandera activada con el sondeo tipo Null (–Sn)
- Solo la bandera FIN activada con el sondeo tipo FIN (–Sf)
- Solo los bits FIN, PSH y URG activados con el sondeo tipo Xmas (-sX)
- Solo la bandera ACK activada con el sondeo tipo TCP ACK –Sa
- Solo la bandera SYN activada con el sondeo tipo TCP SYN –SS

Los 3 primeros escaneos verifican las conexiones abiertas, cerradas o filtradas. Si se recibe un paquete RST, el puerto está cerrado. Si no se recibe ninguna respuesta el puerto se marca como abierto|filtrado. El puerto se marca filtrado si se recibe un error ICMP no alcanzable (tipo 3, código 1, 2, 3, 9, 10, o 13). Para el escaneo tipo TCP ACK cuando se recibe como respuesta el paquete RST indica que el puerto puede estar abierto o cerrado y Nmap lo cataloga como no Filtrado, en tanto que los puertos que no responden o que envían mensajes de error ICMP en respuesta (tipo 3, código 1, 2, 3, 9, 10, o 13), se marcan como filtrados.

Para el escaneo tipo TCP SYN, si se recibe un paquete SYN/ACK indica que el puerto está abierto, y si se recibe un RST (reset) indica que el puerto está cerrado. Si no se recibe ninguna respuesta después de realizar algunas retransmisiones entonces el puerto se marca como filtrado. También se marca el puerto como filtrado si se recibe un error de tipo ICMP no alcanzable (tipo 3, códigos 1,2, 3, 9, 10, o 13)[42]. Para realizar todas las combinaciones posibles hay que tomar en cuenta que se deberían realizar 2^6=64 pruebas posibles, por lo cual en la figura 2-31 se detallan únicamente las combinaciones frecuentes que no utiliza como estándar Nmap:

SYN	ACK	RST	URG	PSH	FIN
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	1	1
1	1	0	0	0	0
1	0	1	0	0	0
0	1	0	1	1	1
0	1	1	0	0	0
1	0	0	1	1	1

Figura 2-31 Combinaciones frecuentes de la bandera TCP

[42] Técnicas de sondeo de puertos con Nmap. (Fuente: http://nmap.org/man/es/man-port-scanning-techniques.html)

y) Verificar el uso de todos los objetivos con IPIDs secuenciales para enumerar los sistemas dentro de la red.

El PID es la identificación de un datagrama IP ubicado en la cabecera del paquete, y se usa con frecuencia en la fragmentación de datagramas IP para identificar los segmentos pertenecientes a un mismo datagrama. En la figura 2-32 se muestra el campo IPID dentro de la cabecera del datagrama IP.

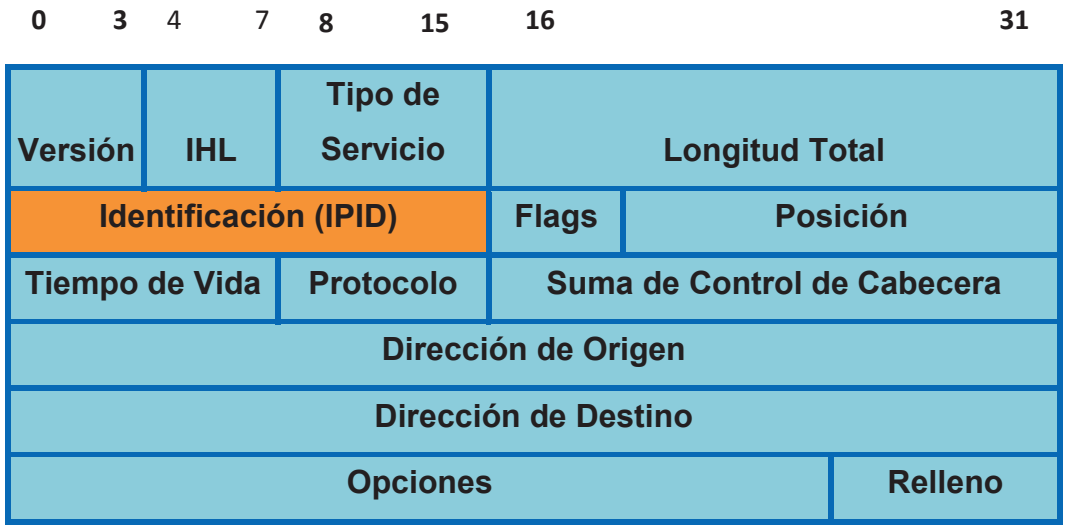


Figura 2-32 Campo IPID de la cabecera de un datagrama IP

Existe la técnica de escaneo "Idle Scan", que utiliza un "host zombie[43]" escogido en base a la predicción del IPID que nos devuelve. Si el valor de IPID es fácil de deducir, entonces el analista de seguridad puede aprovechar esta vulnerabilidad de los sistemas TCP/IP para realizar escaneos mediante un host zombie a los sistemas auditados y recolectar la información necesaria para ejecutar actividades maliciosas. Las etapas de esta técnica son:

1. Identificar el equipo que actúa como "host zombie" en base al análisis del IPID. Es recomendable escoger equipos que respondan con el IPID consecutivo (+1).

[43] Host Zombie es una computadora que tras haber sido infectado por algún tipo de malware, puede ser usado por una tercera persona para ejecutar actividades hostiles. Este uso se produce sin la autorización o el conocimiento del usuario del equipo. Fuente: http://es.wikipedia.org/wiki/Zombi_(inform%C3%A1tica)

2. Solicitud del escaneo desde la máquina de pruebas hacia el objetivo, quien asumirá que las peticiones vienen desde el “host zombie” y no desde nuestra máquina.

3. Nuestra máquina recibirá el resultado del escaneo en base al intercambio de mensajes entre el objetivo y el “host zombie”.

El esquema de los mensajes intercambiados se visualiza en la figura 2-33 para una conexión abierta y en la figura 2.34 para una conexión cerrada. Para una conexión abierta, el Host objetivo devuelve un paquete SYN/ACK al host Zombie, quien responde a su vez con el SYN/RST ya que no fue el quien originó la petición SYN inicial. Además el host zombie devuelve el IPID decrementado en uno al host objetivo y también nos devuelve a la máquina de prueba el IPID nuevamente decrementado en uno, por tanto se recibe el IPID decrementado en 2 (1 de la respuesta al host objetivo + 1 de la respuesta a nuestra máquina).

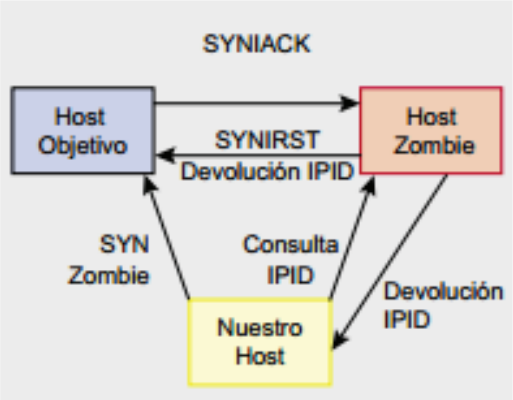


Figura 2-33 Flujo de un Idle Scan detectando un puerto abierto[44]

Para una conexión cerrada, el Host objetivo devuelve un paquete SYN/RST al host Zombie, quien no toma en cuenta este paquete. Luego el host zombie devuelve el IPID decrementado solamente en uno a nuestra máquina de prueba.

[44] Flujo de un Idle Scan detectando 1 puerto abierto. Fuente: http://ramiroencinas.com/docs/idlescan.pdf.

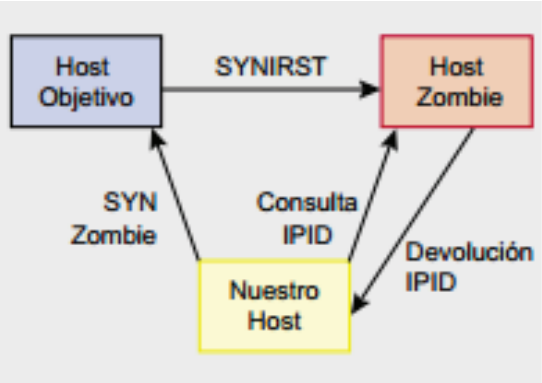


Figura 2-34 Flujo de un Idle Scan detectando un puerto cerrado[45]

Herramienta: hping2

Se utiliza esta herramienta para obtener los IPID de los datagramas IP de los sistemas objetivos, y poder analizar la predictibilidad de los IPID. Con esto se determinará los equipos que pueden ser "host Zombie" para técnicas de Idle Scan sobre los sistemas auditados en el perímetro de red. La sintaxis utilizada es:

Comando: hping2 opciones (Host)

Opciones:

-c Número de paquetes	-a Spoof source address
-i intervalo de tiempo para envío de paquetes	-t Define TTL
-V modo verbose en detalle	-f Fragmentar paquetes
-1 modo ICMP	-m Especificar MTU
-2 modo UDP	-s Especificar puerto origen
-8 modo Scan	-p Especificar puerto destino
-9 Modo escucha	

[45] Flujo de un Idle Scan detectando 1 puerto cerrado. Fuente: http://ramiroencinas.com/docs/idlescan.pdf.

Herramienta: Nmap

Opción –sI (Host Zombie) (Host Objetivo) Realiza el escaneo de manera anónima hacia el host objetivo, como si las peticiones vinieran desde el host Zombie.

Opción –PO previene que Nmap envíe un ping inicial al host Objetivo, asegurando que ningún paquete sea enviado al host objetivo.

Pruebas: Identificación

z) Identificar respuestas TTLs de los objetivos, Uptime de los sistemas, servicios, aplicaciones, fallos de aplicaciones y correlacionar esto con las respuestas de las herramientas de Fingerprinting [W7] para sistemas y servicios.

Herramienta Ping

Se utilizará para verificar las respuestas TTLs de los objetivos.

Herramienta Nmap

Opción:

-v modo verbose para obtener en detalle el escaneo a realizar

-O Activa el escaneo Fingerprinting para detección del sistema operativo e información adicional como Uptime del sistema, fallos de aplicación, etc.

Herramienta WhatWeb

Se usará esta herramienta para determinar información referente al sitio web local, como versión, tipo de Servidor, de CMS, de Lenguaje, etc. Viene incluida dentro del set de utilerías del S.O Backtrack (Figura 2-35) ubicado en Information Gathering / Web Application Analysis / CMS Identification / WhatWeb.

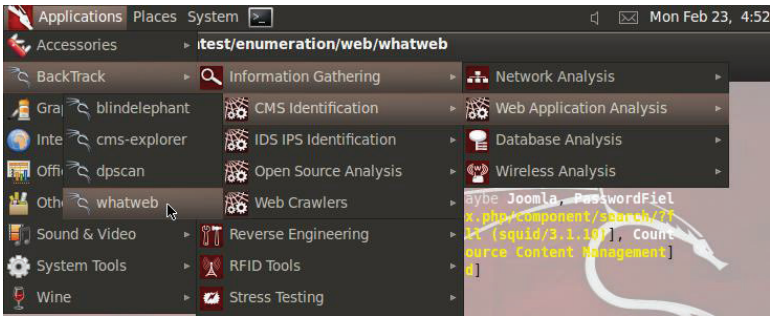


Figura 2-35 Herramienta WhatWeb

Herramienta CMS-Explorer:

Se utilizará esta herramienta para verificar los plugins[46] del sitio Web local. Viene incluida dentro del set de utilerías del S.O Backtrack (Figura 2-36) ubicado en Information Gathering / Web Application Analysis / CMS Identification / cms-explorer.

Figura 2-36 Herramienta cms-explorer

2.4.2.2 Módulo Verificación de Acceso

Este módulo permite enumerar los principales puntos de acceso dentro del alcance. En esta sección se desarrollarán las siguientes pruebas: Red, Servicios y Autenticación.

[46] Plugin es un complemento que se relaciona con otra aplicación para aportarle una función nueva y generalmente muy específica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de la API. Fuente: http://es.wikipedia.org/wiki/Complemento_(inform%C3%A1tica)

Pruebas: Red

a) Solicitar los servicios conocidos y comunes que utilizan UDP para las conexiones desde todas las direcciones

Se utilizará la herramienta Netcat para realizar pruebas de solicitud y acceso a los servicios conocidos que utilizan UDP para las conexiones dentro del perímetro de red.

Herramienta Netcat

En la figura 2-37 se muestran las principales opciones de esta herramienta:

Opción:

-u Se activa el modo UDP.

-l Escucha conexiones, dejando abierto un puerto en espera de una conexión.

-n Especifica que solo se utilice direcciones IP, y que no intente resolver nombres de dominio mediante DNS Lookup.

-p Especifica el número de puerto local.

-s Especifica una Dirección IP fuente.

-w Especifica un tiempo determinado para realizar conexiones.

-v Modo verbose, información detallada.

-g Especifica el Gateway, para utilizar hasta 8 Routers como puentes de conexión.

-G Especifica puntos de Routing.

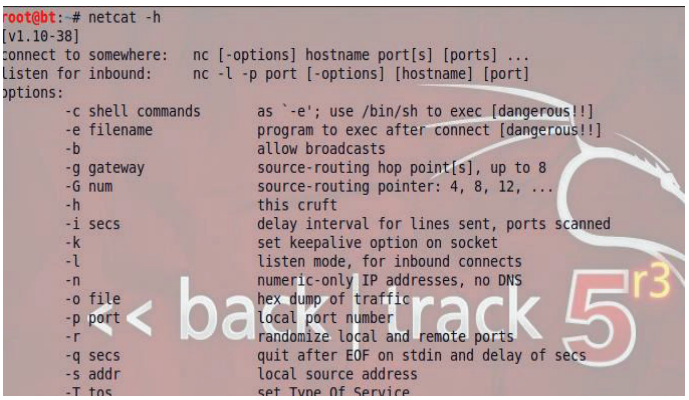


Figura 2-37 Herramienta Netcat

b) Manipular los servicios de red y enrutamiento para el acceso a restricciones dentro del alcance.

Una vulnerabilidad en las redes TCP/IP es manipular el enrutamiento de los paquetes de origen, de tal forma que se pueda controlar la ruta de ida y retorno hacia los sistemas objetivo. Un analista de seguridad puede tomar ventaja de esta vulnerabilidad y saltarse los sistemas Firewall/Ids que podrían bloquear esta comunicación, o a su vez enviar el tráfico por rutas erróneas. Para ello hay que tomar en cuenta que existen 2 tipos de encaminamiento dependiendo del origen:

- El estricto
- El relajado

El estricto indica que el paquete debe definir cada salto hasta llegar al destino en la ruta establecida, mientras que el relajado indica a los routers y dispositivos de red como Firewall, que ellos pueden generar la mayor parte del encaminamiento hasta llegar al destino. Algunos sistemas de protección (Firewall/Ids) y routers permiten el encaminamiento dependiente del origen estricto y muy pocos admiten el encaminamiento dependiente del origen relajado, en donde se permitirá que el paquete pase por nuestra máquina al regresar. Con este sistema de encaminamiento dependiente del origen es posible suplantar una dirección IP y obtener las respuestas en el viaje de vuelta. La mayoría de los sistemas de protección y routers ignoran las opciones del encaminamiento dependiente del origen, pero no todos.[47]

Herramienta Netcat con la opción de Spoofing permite escoger las direcciones IP para el encaminamiento de paquetes.

Opción –g Permite especificar hasta 8 saltos para enrutar el paquete a su destino.

Opción –G Hace referencia a un puntero de Salto que indica desde donde comienza el enrutamiento de paquetes definidos anteriormente con la opción –g.

[47]http://issuu.com/griillo7/docs/el-sendero-del-hacker. Suplantación de una dirección IP. Pág. 200

Pruebas: Servicios

c) Pedir todas las etiquetas de servicio (banners) de los puertos descubiertos, mediante interacciones con el servicio tanto con peticiones válidas y no válidas.

Esta prueba hace referencia al término Banner Grabbing, que permite la identificación de las características (plataforma, versión, bases de datos, etc) de los servicios/sistemas mediante los banners. Se recomienda ocultar información relativa a los sistemas en los banners, para minimizar la cantidad de datos que un atacante puede obtener acerca de los servicios/sistemas auditados.

Herramienta: Netcat

Opción: -vv Activa el modo detallado: Verbose

d) Mapear cada puerto abierto a un demonio (servicio), aplicación (producto o código específico que usa el servicio) y protocolo (el medio para interactuar con ese servicio o aplicación).

En base a los resultados del literal anterior c, se procede a asignar cada puerto abierto a su correspondiente servicio, aplicación y protocolo específico.

e) Verificar el uptime del sistema con respecto a las últimas vulnerabilidades y liberación de parches.

Utilizamos el comando **uname –a** para verificar el nivel de parcheado del sistema GNU/Linux y el **comando cat /etc/issue** para constatar la versión.

Comando: uname –a

Permite verificar la siguiente información: Detalles de la arquitectura del Sistema Operativo, Nivel de parcheado, Procesador de la máquina, etc.

Comando: cat /etc/issue

Permite constatar la versión y tipo de distribución instalada.

f) Identificar los componentes del servicio en escucha.

El servicio de un sistema en escucha trabaja bajo un esquema cliente-servidor, en donde el puerto del servicio escucha conexiones en el servidor y posteriormente el cliente solicita la conexión en dicho puerto.

Herramienta Netcat

Opción -l Activa el modo de escucha de conexiones en un puerto específico

Opción –p Define el puerto origen

La sintaxis del comando para activar la escucha de conexiones en un puerto es:

Ncat –vv –l –p (puerto_origen) (Host que escuchará peticiones) (puerto destino).

Posteriormente el cliente se conecta al servicio con el puerto específico para iniciar la comunicación cliente-servidor. El comando es el siguiente:

Ncat –vv (Ip_Servidor) (Puerto_de_escucha)

g) Verificar el uptime de los servicios comparado con las últimas vulnerabilidades y liberación de parches.

El uptime hace referencia al tiempo de permanencia activo de un servicio, que se relaciona con la disponibilidad y confiabilidad del mismo. A partir de este valor, se puede correlacionar con el tiempo que no ha recibido actualizaciones ni parches del servicio. Según esto es posible realizar un análisis de las vulnerabilidades para cada servicio del perímetro de red. En esta sección detallamos las herramientas para verificar el uptime de los servicios y sus vulnerabilidades. El detalle de cada vulnerabilidad se detalla en el capítulo 3 Sección Pruebas de Servicios literal g.

1. Herramientas para verificar uptime de los servicios.

Herramienta: Módulo server-status del archivo apache del servidor Web

Este módulo permite supervisar vía browser información del consumo de recursos del servidor Apache, como Versión, tipo de servicio, lenguaje y uptime del sistema.

Herramienta: Nmap

Esta utilidad con la **opción –Sv** nos permite identificar el tipo de servicio y versión del servicio auditado.

Comando: service named status

Named es el servicio que utiliza el servidor Dns. Este comando permite verificar las características tales como tipo, versión y estado del servidor Dns.

2. Herramientas para verificar vulnerabilidades en los servicios.

Herramienta: Nikto

Se utilizará esta herramienta para verificar pruebas de vulnerabilidad XSS contra el servidor Web local. Esta herramienta viene incluida dentro del set de utilidades de Bactrack en la sección “Web Vulnerability Scanners” de la Figura 2-38.

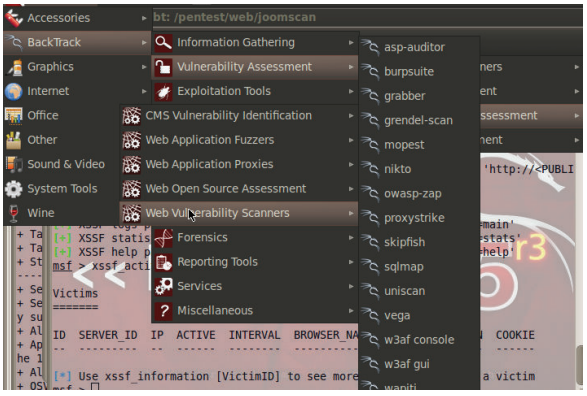


Figura 2-38 Herramienta Nikto en Backtrack

Herramienta SqlMap

Se utilizará esta herramienta para detectar y explotar las vulnerabilidades de tipo Inyección Sql, a fin de obtener acceso total a la base de datos del servidor Web infectado con el código Sql malicioso. Esta utilidad viene incluida dentro del set de herramientas de Backtrack en la sección “Web Vulnerability Scanners”.

Herramienta XSSF

Permite verificar víctimas de ataques XSSF genéricos en un Servidor Web. **P**ara acceder se abre una consola de metasploit framewok con el comando “msfconsole” y se carga el plugin Xssf con el comando “load xssf”.

Herramienta Metasploit Msf

Esta herramienta permitirá verificar la vulnerabilidad Backdoor Vsftpd_234, aplicada a versiones Vsftpd 2.3.4. Para verificar el Backdoor se ejecuta “search vsftpd” y luego se aplica los comandos “use exploit/unix/ftp/vsftpd_234_backdoor”, “set RHOST Ip_Servidor” y “exploit”.

Verificación de la sección “allow query”, “allow recursión” y “allow-transfer” en los archivos de configuración del servidor Dns.

Esta prueba nos permite determinar si el servidor Dns local está afectado por las vulnerabilidades: Entropía de insuficiencia de zócalos (sockets) de DNS y transferencias de zona libre.

h) Verificar los hosting virtuales por HTTP y HTTPS.

El hosting virtual es el servicio de alojamiento de sitios web en un mismo servidor, que se puede implementar localmente o a su vez subcontratar al proveedor de servicios. Utiliza recursos disponibles y compartidos, tales como procesador, memoria, espacio de disco y carga del sistema. Es una de las modalidades más utilizadas por las empresas cuyo giro de negocio es el alojamiento Web.

Herramienta: Verificación de hosting virtual en el archivo de configuración "httpd.conf" del servidor Web

Verificamos en el archivo de configuración apache "httpd.conf" si está activa la opción de Virtual Hosts, la cual permite mantener múltiples dominios en el mismo servidor.

Pruebas: Autenticación

i) Enumerar los accesos que requieren autenticación y documentar todos los privilegios descubiertos, que pueden ser usados para proveer acceso.

Los accesos que requieren autenticación son:

- Acceso a la página de administración del sitio Web local
- Acceso a la interfaz gráfica de administración "Webmin" del Servidor Firewall/Control de Contenido.
- Acceso al servidor Ftp con el usuario de prueba creado.

Herramienta Wireshark

Se utilizará esta herramienta para verificar el tráfico generado al probar los distintos accesos que requieren autenticación.

j) Verificar la fortaleza de la autenticación a través del cracking de claves y re-aplicación de claves descubiertas para todos los puntos de acceso que requieren autenticación.

El cracking de contraseñas permite recuperar contraseñas a partir de los datos que han sido transmitidos y almacenados por un sistema informático. El objetivo es el ingreso a los sistemas operativos, recursos de red, sistemas, aplicaciones, archivos, etc. Entre los métodos para realizar el cracking de claves tenemos:

1. Vulnerabilidad de algoritmos criptográficos: Depende del algoritmo criptográfico. En cifrados modernos, como MD5, resulta imposible encontrar una coherencia lógica entre el texto cifrado y el descifrado, ya que cada clave ha sido generada a partir de una cadena diferente llamada "semilla"[48].

2. Adivinar: Es una técnica basada en ingeniería social, relacionando el perfil, actividad del usuario para probar la contraseña del sistema auditado.

3. Fuerza bruta (Incremental): Es un método menos efectivo que prueba todas las combinaciones posibles, por lo cual es muy costoso en tiempo computacional.

4. Diccionario: En base a una lista de palabras fáciles de recordar, existente en la lengua, realiza las combinaciones posibles para hallar la contraseña. Es más eficiente que el cracking por fuerza bruta.

5. Tablas Rainbow (Arcoiris): Obtiene la clave en texto simple a partir del resultado de una función de hash. Se manejan las tablas rainbow que son de búsqueda especial, donde se tiene pre computado los pares “hash- clave texto claro”

Herramienta: Jhon the Ripper [W10]

Utilizamos esta herramienta para determinar las contraseñas de los usuarios de los servidores en el perímetro de red. Esta herramienta por defecto sin ningún argumento específico, comienza a descifrar las contraseñas en modo de cracking simple, luego utiliza un diccionario de reglas y por último usa el modo incremental (Fuerza Bruta). La sintaxis del comando se muestra a continuación. En la figura 2-39 se muestra las opciones de esta herramienta.

[48]http://es.wikipedia.org/wiki/Password_cracking.

John (Opciones) (Archivos-Passwords)

--single → Establece el cracking en modo simple

--wordlist=File → Establece el cracking en modo Diccionario o Word List leyendo las palabras establecidas en el archivo.

--rules → Establece el manejo de reglas para el modo Word List

Figura 2-39 Opciones de la herramienta John the Ripper

k) Verificar el proceso para recibir una autenticación

En esta sección es importante aclarar la diferencia entre autenticación y autorización. La autenticación es el proceso en el cual se valida la identidad del remitente quien mediante una petición, solicita conectarse algún sistema o servicio. En tanto que la autorización es el proceso de verificación que autoriza al usuario ya identificado el acceder a recursos dentro del sistema. En el capítulo 3, sección 3.1.2.2 en las Pruebas de Autenticación se detallará el proceso para recibir una autenticación.

l) Evaluar en busca de errores lógicos en la aplicación de la autenticación.

Los errores lógicos más importantes derivados del sistema de autenticación se detallan en la Figura 2-40[49].

Vulnerabilidad	Descripción
Sistemas que permitan saltarse el método de autenticación.	Con estas pruebas, buscamos verificar la fortaleza del sistema de autenticación, y sus recursos. Las pruebas incluyen verificar el acceso a una página de un recurso interno saltándose la página de registro de autenticación. Además probaremos si es posible saltarse las medidas de autenticación mediante modificaciones en las peticiones URL, manipulación de formularios o falsificación de sesiones.
Sistemas que permitan recolectar un conjunto válido de usuarios, al interactuar con el sistema de autenticación.	Se puede derivar errores lógicos cuando el sistema devuelve mensajes que permiten predecir los usuarios del sistema, tales como mensajes que confirman la existencia de un usuario y que la contraseña suministrada es incorrecta. Un atacante puede usar esta información para realizar futuros ataques por fuerza bruta o ataques de usuarios/contraseñas predeterminadas.
Sistemas que no gestionen correctamente el almacenamiento y reseteo de la contraseña en el	Es importante verificar la gestión del sistema de recuperación/reset de la contraseña del aplicativo web, en caso de pérdida u olvido de la misma, mediante envío de mails de reset de contraseña y/o solicitando responder preguntas de seguridad, entre otros mecanismos.

[49] Guía de Pruebas OWASP V3.0, Comprobación del Sistema de Autenticación. (Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf, Página 121)

navegador.	También es relevante verificar el procedimiento de cacheo de las contraseñas en el navegador del cliente, mediante la característica "cache password" en navegadores Web, o el almacenamiento de la contraseña en una cookie permanente.
Sistemas que no gestionen correctamente la caché de navegación y salida de sesión	Es importante evaluar correctamente la comprobación de las funciones de cierre de sesión, para verificar que no sea posible utilizar la misma después del cierre, o que tras un tiempo de inactividad se cierre la sesión automáticamente y ningún dato sensible permanezca en la caché del navegador

Figura 2-40 Errores lógicos en el sistema de autenticación Web

Herramienta: Web Scarab [W5]

Se utilizará esta herramienta para verificar y modificar las peticiones Http en busca de errores lógicos en el sistema de autenticación de la aplicación Web. También se comprobará el sistema de cacheo de las contraseñas en el navegador al analizar las cookies interceptadas. Para activar esta herramienta se configura el navegador para que el proxy sea nuestra propia máquina en el puerto 8008, de forma que se intercepte todo el tráfico http. En la figura 2-41 se muestra la interfaz gráfica de Web Scarab. En el panel superior está el árbol de las peticiones/respuestas http/https, y en el inferior el detalle de los paquetes interceptados.

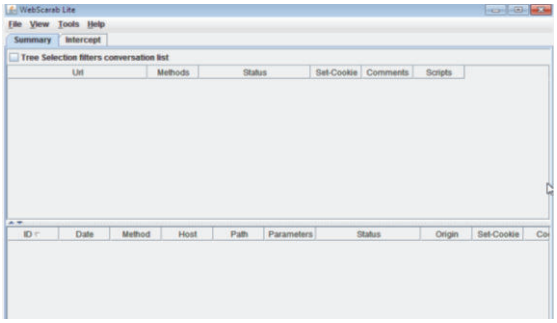


Figura 2-41 Interfaz gráfica de herramienta Web Scarab

2.4.2.3 Módulo Verificación de Confianza

Este módulo permite realizar las pruebas de confianza a los sistemas del Perímetro de Red, sin necesidad de identificación o autenticación. En esta sección se desarrollarán las siguientes pruebas: Spoofing, Phishing, y Abuso de Recursos.

Pruebas: Spoofing

a) Verificar las formas de acceder a la propiedad dentro del alcance haciendo spoofing de la dirección de red como uno de los sistemas confiados.

Para cumplir con esta sección, realizamos un ataque denominado Main in the Midle (MITM) **[W9]** hacia el Servidor Web/Ftp/Dns desde una máquina víctima, que en este caso es la máquina de pruebas Windows. Este ataque permite leer, interceptar y modificar a voluntad, los mensajes entre 2 partes sin que ninguna de ellas conozca que el enlace entre ellos ha sido violado[50]. En la figura 2-42 se esquematiza este ataque, en el cual se intercepta el tráfico proveniente desde la máquina Víctima hacia el Servidor Web, sin que ambos sistemas conozcan que esta comunicación ha sido intervenida.

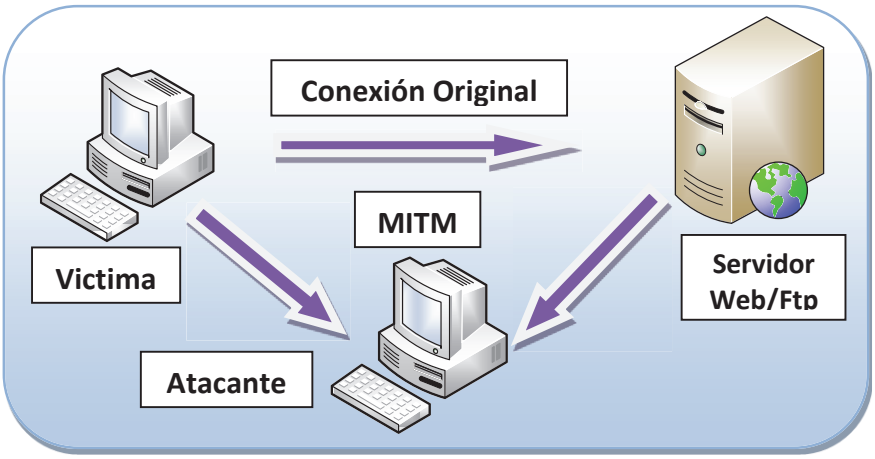


Figura 2-42 Diagrama de técnica "Main in the Midle" al Servidor Web/Ftp

50 http://es.wikipedia.org/wiki/Ataque_Man-in-the-middle.

Herramienta: Cain y Abel [T2]

Se utilizará esta herramienta para ejecutar la técnica de Spoofing y capturar el tráfico proveniente de una máquina víctima, en este caso la máquina de pruebas Windows hacia el Servidor Web/Ftp y capturar las contraseñas para el acceso al sistema Web local y al Servidor Ftp.

b) Verificar sí los mecanismos de caching Dns pueden ser envenenados.

Uno de los mecanismos de envenenamiento Dns consiste en la técnica de suplantar la tabla Dns, para falsear las traducciones y redirigir las peticiones hacia direcciones web falsas. Esto se debe a una vulnerabilidad en el Servidor Dns que no valida correctamente las respuestas Dns para asegurarse que provengan de una fuente autoritativa, y permite que el servidor guarde localmente información incorrecta, conduciendo a que los clientes puedan acceder a exploits para robo de información, inyección de troyanos, código malicioso, etc.

Herramienta Cain&Abel

Se utiliza esta herramienta con el módulo ARP-Dns para suplantar la tabla Dns al redirigir la petición de google.com hacia nuestra Página Web local. El objetivo es verificar si el Servidor Web local puede contaminarse localmente con una entrada en su tabla Dns para conducir a un Sitio Web de forma intencionada.

Pruebas: Phishing

c) Verificar que las URL's para el envío y consultas en el objetivo sean concisos, dentro del mismo dominio, usando únicamente el método POST y una marca consistente.

El estándar HTTP/1.0 utiliza 3 métodos para representar las operaciones de envío, recepción de información y chequeo de estado. El método GET se usa en el acceso a un recurso que depende de la información proporcionada por el usuario. El método

POST se utiliza en operaciones que crean, borran o actualizan información, como en el caso de datos contenidos en un formulario[51]. El método HEAD en cambio solicita información de un fichero como tamaño, tipo y fecha de modificación.

Existen ataques relacionados a la modificación de las URL's cuando se transmiten peticiones y respuestas en un modelo cliente servidor. Estos ataques aprovechan la vulnerabilidad de las peticiones con métodos GET para intentar modificar los parámetros enviados en la URL. Estas llamadas GET pueden ser cacheadas, indexadas por buscadores, agregar el enlace a favoritos e incluso pasar la URL completa a otra persona[52]. Para evitar esta vulnerabilidad se recomienda el uso de métodos POST donde los parámetros son enviados en el cuerpo de la petición.

Herramienta Web Scarab

Se utilizará esta herramienta para interceptar las URL's en la comunicación cliente al Servidor Web local y verificar el tipo de peticiones http transmitidas.

d) Examinar los registros de dominio de alto nivel para dominios similares a los identificados dentro del alcance.

Los dominios de nivel superior denominados TLD son asignados por la IANA (Internet Assigned Numbers Authority) para su uso a nivel del Internet. Se tienen 3 grupos detallados a continuación:

1. Dominios de nivel superior de infraestructura (.arpa)

El dominio .arpa fue establecido en 1985 para que facilitara la transición hacia los sistemas DNS y luego ser eliminado[53].

51 Envío de parámetros con la petición Http. Fuente: http://librosweb.es/libro/ajax/capitulo_7/interaccion_con_el_servidor.html.
52 http://blog.micayael.com/2011/02/09/metodos-get-vs-post-del-http/.
53 http://es.wikipedia.org/wiki/.arpa.

2. Dominios de nivel superior geográficos (ccTLD)

Identifican un territorio y están compuestos por las dos primeras letras de un país, conforme a los códigos estándar ISO3166-1 para la representación de nombres de países o territorios[54].

3. Dominios de nivel superior genéricos (gTLD)

Están conformados por un mínimo de 3 letras y son regularizados directamente por la ICANN[55]. Ofrecen marcos regulatorios legalizados, coherentes y unificados a nivel mundial. Entre los más frecuentes tenemos: .com, .net, .org, .info, .biz, .name.

4. Dominios de nivel superior internacionalizados

Usan caracteres distintos del alfabeto latín.

Las pruebas validarán si los registros del dominio local "www.iza.sanhueza.ec" son similares a otros registros para dominios de alto nivel gTLD. El dominio local es un dominio de tercer nivel, tal y como se muestra en la figura 2-43. El dominio de primer nivel "ec" se relaciona al dominio geográfico del código territorial Ecuador. El dominio de segundo y tercer nivel se relaciona al nombre de la organización, empresa o el dueño del dominio. Cuando existe un dominio de tercer nivel, es frecuente hacer referencia a los subdominios para indicar una subdivisión dentro de la empresa.

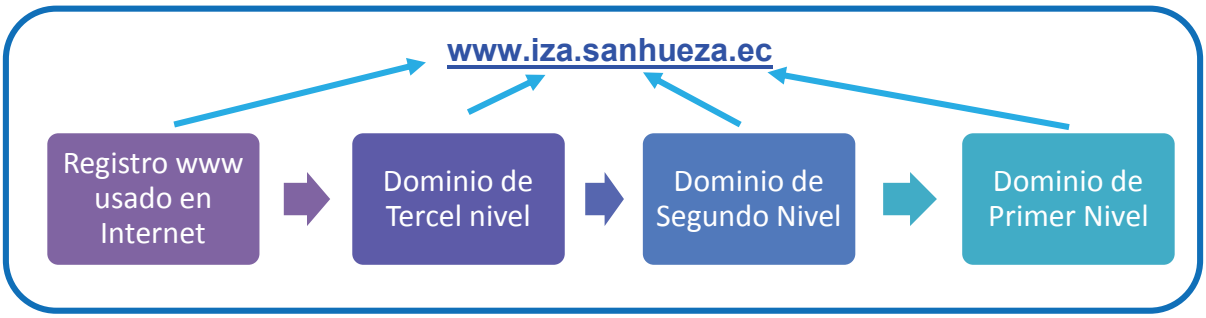


Figura 2-43 Relación del dominio local con los de alto nivel gTLD

[54] Dominios geográficos. Fuente: http://www.movistar.es/rpmm/estaticos/negocios/fijo/servicios-sobre-adsl/manuales/tutorial-dominios.pdf.

[55] ICANN (Corporación de Internet para la Asignación de Nombres y Números) es una organización que opera a nivel multinacional/internacional, responsable de asignar las direcciones del protocolo IP, los identificadores de protocolo, las funciones de gestión del sistema de dominio y la administración del sistema de servidores raíz. Fuente: https://www.icann.org/es

Herramienta: Vía Web "nic.ec"

Este operador de registro se utilizará para examinar si existen registros de dominio de alto nivel gTLD similares al dominio local www.iza.sanhueza.ec.

Pruebas: Abuso de Recursos

d) Evaluar la profundidad del acceso a la información de negocios o confidencial disponible en servidores web sin credenciales establecidas.

Estas pruebas permiten identificar el grado de exposición de la información confidencial/negocios del servidor web local sin necesidad de autenticación, que pueda contraer actividades maliciosas tales como: robo de información, accesos no autorizados, denegación de servicios, etc.

2.4.3. PROCEDIMIENTO FORMAL DE PRUEBAS DE INTRUSIÓN PARA LA FASE DE INVESTIGACIÓN.

Esta fase permite verificar la forma de trabajo de los activos, y las configuraciones de control y servicios en el perímetro de red. Para ello se cumplirán las tareas más relevantes de los módulos: Verificación de configuración y de exposición.

2.4.3.1 Módulo Verificación de Configuración

En este módulo se verifica la forma de trabajo de los activos y la posibilidad de eludir la seguridad funcional, explotando la configuración de controles de acceso y aplicaciones. Se desarrollarán las siguientes pruebas: Controles de Configuración, Errores de configuración común y Mapeo de Limitaciones, las cuales serán realizadas de forma manual, validando las configuraciones de los servicios auditados.

Pruebas: Controles de Configuración

a) Examinar los controles para verificar las configuraciones y líneas base de los sistemas, equipos y aplicaciones para cumplir con la intención de la organización y reflejar una justificación de negocio.

Los controles son medios para influir en el impacto de las amenazas y sus efectos cuando las interacciones con un sistema, aplicación o servicio son necesarios[56]. El control de Privacidad es el que influye directamente sobre la verificación de las configuraciones y líneas base de los sistemas/servicios. Este control trata de asegurar los medios cuando se accede, muestra o se intercambia información con un activo para que no sea conocido fuera de las partes autorizadas. En esta sección se analizará el control de privacidad aplicado a los servicios del Perímetro de red para cumplir con la intención del proyecto y reflejar la justificación del mismo.

b) Examinar Access Control Lists (ACL) y funciones de negocios configurados en redes, sistemas, servicios y aplicaciones en el alcance para garantizar que cumplan la intención de la organización y reflejen una justificación de negocio.

Las ACL son listas de control que determinan las reglas de acceso configuradas en los servicios de un sistema. En esta sección se verificará que las ACL estén trabajando de acuerdo a la funcionalidad apropiada de las redes, sistemas y aplicaciones dentro del alcance para cumplir las políticas de seguridad definidas acorde a los objetivos e intención del proyecto. Estas pruebas consisten en verificar las ACLs representadas por las políticas de iptables en el Servidor Firewall y las reglas de control de acceso Http definidas en el Servidor Squid Control.

[56] Controles. Sección 1.2 Fuente: http://www.isecom.org/mirror/OSSTMM.3.pdf.

Pruebas: Errores de Configuración Comunes

c) Verificar servicios disponibles que no son redundantes innecesariamente y que coinciden con el rol definido para la intención de los sistemas.

En esta sección se documenta todos los servicios disponibles en el sistema con su respectivo rol, que esté acorde a los objetivos e intención de los sistemas y no tengan redundancia innecesaria en la red.

d) Verificar que los parámetros default han sido cambiados. Algunos dispositivos o aplicaciones se entregan con una cuenta administrativa default u oculta. Estas cuentas deberían ser cambiadas o, sí es posible, inhabilitarlas o eliminarlas y reemplazarlas con una nueva cuenta administrativa.

En esta sección es importante constatar que las configuraciones por default que entregan los sistemas/servicios se cambien debido a los siguientes motivos: evitar amenazas en el sistema de autenticación por accesos no autorizados, configuraciones por defecto de los sistemas/servicios que un atacante pueda usar para realizar actividades maliciosas, tales como inyección de puertas traseras, exploits para pruebas de escaneo y vulnerabilidades en los sistemas/servicios del perímetro de red.

e) Verificar que la administración se realice localmente o con controles para limitar quien o que puede acceder por interfaces remotas de administración del equipo.

En los sistemas del perímetro de red se manejan 2 tipos de administración: Local y Remota. Detallaremos los mecanismos de seguridad implícitos en la administración, tal que cumpla con las políticas de seguridad y garantice los objetivos y justificación del proyecto.

Pruebas: Mapeo de Limitaciones

f) Chequear por servicios/características disponibles no necesarias en el alcance.

Estas pruebas permiten determinar los servicios activos dentro del Perímetro de Red, pero que no son necesarios en el alcance de acuerdo a los objetivos e intención del proyecto. Es primordial referir estos servicios, ya que pueden representar agujeros de seguridad o puntos de acceso a los servicios de la red perimetral, exponiendo la información sensible a amenazas, como: intercepción y/o modificación de datos, secuestro de sesiones, accesos no autorizados, etc.

2.4.3.2 Módulo Verificación de Exposición

En esta sección se valida el grado de exposición de información sensible al público que pueda conducir al acceso desde múltiples lugares con la misma autenticación. Se desarrollarán las siguientes pruebas: Enumeración de Exposición. Estas pruebas serán realizadas de forma manual, validando la exposición de la información relativa a los servicios auditados.

Pruebas: Enumeración de Exposición

a) Enumerar información relativa a la organización como los gráficos o tablas de la organización, títulos clave del personal, descripciones de puestos, números de teléfono personales y de trabajo, números de teléfono móvil, las cartas de empresas, documentos compartidos, hojas de vida, afiliaciones de organización, direcciones e-mail pública y privada, log-ins, esquemas de login, contraseñas de programas, métodos de respaldo, aseguradoras, particulares o cualquier información de la organización señalada implícitamente como confidencial en los reglamentos y la política.

b) Enumerar la exposición de los sistemas, servicios y aplicaciones que detallan el diseño, tipo, versión o estado de los objetivos o de los recursos fuera del alcance como anuncios.

Herramienta Google Hacking GH

Se puede utilizar esta herramienta para cumplir los literales a y b de esta sección y poder constatar la exposición de los sistemas, servicios y aplicaciones mediante el motor de búsqueda público de Google Hacking. Los operadores básicos se detallan en el Anexo N. 7. En el proyecto actual para los sistemas del perímetro de red no es posible ejecutar estas pruebas, ya que el Sitio Web no es público a nivel de internet. En la figura 2-44 se visualiza este motor de búsqueda con la base de datos de Google Hacking denominada GHDB a nivel del internet.

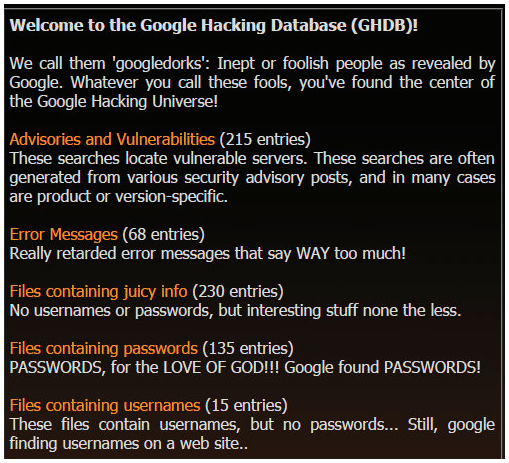


Figura 2-44 Base de datos de Google Hacking GHDB

2.4.4 PROCEDIMIENTO FORMAL DE PRUEBAS DE INTRUSIÓN PARA LA FASE DE INTERVENCIÓN.

La fase de intervención permite probar los sistemas y servicios dentro del perímetro de red. Para cumplir esta sección se desarrollarán las tareas más relevantes de los módulos: Auditoría de Privilegios y Validación de Supervivencia.

2.4.4.1 Módulo Auditoria de Privilegios

Este módulo define pruebas para validar los sistemas de autenticación, autorización y escalada de privilegios, comprobando la posibilidad de ingresos no autorizados y si es posible obtener privilegios mayores a los proporcionados en el acceso inicial. En esta sección se desarrollarán las siguientes pruebas: Autorización y Escalación de Privilegios.

Pruebas: Autorización

a) Examinar y verificar cualquier forma de obtener una autorización fraudulenta para ganar privilegios similares a las del otro personal.

Una vez que el mecanismo de autenticación ha finalizado, el siguiente paso es el proceso de autorización donde se consultan los roles, permisos y privilegios que dispone el usuario. En esta etapa de la auditoría se toma en consideración las pruebas del tipo Path Transversal[57] detalladas a continuación:

Pruebas de Path Transversal: El objetivo es la búsqueda de vulnerabilidades en el proceso de autenticación del tipo Path Transversal, que permita acceder a información sensible, tal como datos personales de otros usuarios en el sistema. Las pruebas que se realizan en esta sección son la enumeración de Vectores de Entrada, que verifican las partes vulnerables de la aplicación ante las entradas del usuario, como consultas Http Get y Post, opciones de carga de archivos y formularios Html.

Herramienta Web Scarab

Se utilizará esta herramienta para verificar las variables de entorno usadas en las consultas http tipo get cuando el usuario administrador se loguea al sitio web local.

[57] Guía de pruebas OWASP_VER_3.0 Sección 4.6.3 Pruebas de Pruebas de Path Transversal. Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf.

b) Pruebas para evadir el esquema de autenticación y acceder a información reservada/sensible en el sistema.

En esta sección se evalúa la robustez del sistema de autenticación para acceder a la información sensible del sitio web local. Se realizarán las siguientes pruebas:

1. Llamadas directas a páginas internas, saltándose la página de registro.

El objetivo de las peticiones directas a páginas internas es verificar si el sistema de autenticación puede saltarse. Este procedimiento se realizará de forma manual.

2. Modificación de las peticiones para engañar al sistema de autenticación mediante técnicas de manipulación de formularios, URL's o falsificando sesiones.

Si el sistema de autenticación referencia valores con parámetros de valor fijo, un analista de seguridad puede intentar alterar estos parámetros y obtener acceso a áreas protegidas sin proporcionar credenciales válidas. Este procedimiento se realizará de forma manual.

3. Predicción de los IDs de sesión.

La mayoría de aplicaciones Web gestionan el sistema de autenticación mediante valores de identificación de sesión (Sesión ID), o cookies. Si se llegan a predecir estos valores, el acceso no autorizado a la aplicación sería corrompida, haciéndose pasar por un usuario previamente autenticado. Las cookies se usan para el manejo de sesiones, mediante el registro de acciones e identificación del usuario durante el proceso de múltiples peticiones. Una o más cookies son generadas por el servidor y son enviadas al cliente, el mismo que envía la cookie de regreso al servidor en todas las conexiones futuras hasta que esta expire o sea destruida[58]. Los valores dentro de las cookies pueden incrementar linealmente según el diseño del programa, lo que permitiría a un atacante deducir IDs de sesiones válidas. Se utilizará la **herramienta**

[58] Pruebas para el esquema de gestión de sesiones. Sección 4.5.1. Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf.

Web Scarab para analizar los IDs de sesión generados una vez que el usuario administrador accede al sitio web local e interactúa con el mismo.

Pruebas: Escalación de privilegios

c) Coleccionar información acerca de personas con altos privilegios. Buscar por roles o posiciones de confianza, gateways de acceso para personas de confianza y cualquier medio físico tales como tokens o tarjetas inteligentes.

En esta sección se definen los roles y/o usuarios de confianza dentro del sistema, para realizar la escalada de privilegios. Este mecanismo ocurre cuando un usuario logra acceder a más recursos de las que le fueron asignadas, y tales cambios deberían haber sido prevenidos por la aplicación. Esto es usualmente causado por una falla en la aplicación. Las pruebas se detallan en el capítulo 3 sección 3.1.4.1 Pruebas Escalación de Privilegios.

d) Verificar los límites de los privilegios en los objetivos y si los medios existen para escalar estos privilegios

Las pruebas de elevación de privilegios determinan si es posible que un usuario pueda elevar su rol y acceder a recursos a los que inicialmente no debería acceder. El CVE: 2012-0056 publicado en Enero 2012 detalla una vulnerabilidad que se basa en el acceso a /proc/pid/mem de los sistemas Linux/Unix, que aprovecha la debilidad de no revisar los permisos en el acceso y no validar correctamente los servicios, consiguiendo el acceso a una cuenta con privilegios de administrador.

Herramienta Mempodipper [W6]

Se utilizará esta herramienta para validar si los sistemas auditados son vulnerables ante pruebas de escalada de privilegios.

2.4.4.2 Módulo Validación de Supervivencia

En este módulo se verificará la resistencia y continuidad de los servicios en el perímetro de la red, ante cambios excesivos o denegación del servicio. En esta sección se desarrollarán las siguientes pruebas: Resiliencia.

Pruebas: Resiliencia

a) Verificar los puntos únicos de fallo (cuellos de botella) en la infraestructura donde el cambio o el fracaso puede causar una interrupción del servicio.

Esta prueba se realizará de forma manual detallando los puntos vulnerables a fallos o cambios excesivos en los servicios del perímetro de red. Con esta información el analista de seguridad puede determinar los puntos únicos de fallo que puedan provocar interrupción de los servicios.

b) Verificar el impacto de acceso a los objetivos en la cual se puede causar fallas del sistema o servicio [W8].

Un ataque de denegación de servicio, también llamado ataque DoS (de las siglas en inglés Denial of Service) o DDoS (de Distributed Denial of Service), es un ataque a un sistema de computadoras o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos[59]. Estos ataques se logran mediante el consumo de ancho de banda en el sistema objetivo y/o la sobrecarga de sus recursos computacionales. Dentro de la clasificación de estos ataques, el DDoS se logra mediante la saturación de flujo de información desde varias conexiones, con el uso de una botnet, que son un conjunto de robots informáticos o bots que se ejecutan de manera autónoma y automática para enviar las peticiones concurrentes al sistema objetivo.

[59] Ataque de Denegación de servicio. Fuente: http://es.wikipedia.org/wiki/Ataque_de_denegaci%C3%B3n_de_servicio.

Para ejecutar este tipo de ataques se usa la conexión incompleta en el proceso de establecimiento de la conexión de 3 vías de las redes TCP/IP. La vulnerabilidad consiste en evitar la confirmación Ack del agente que origina las peticiones Syn, ante las respuestas TcpSyn/Ack enviadas por el servidor objetivo. En la Figura 2-45 se esquematiza el diagrama utilizado para realizar el ataque de denegación de servicio.

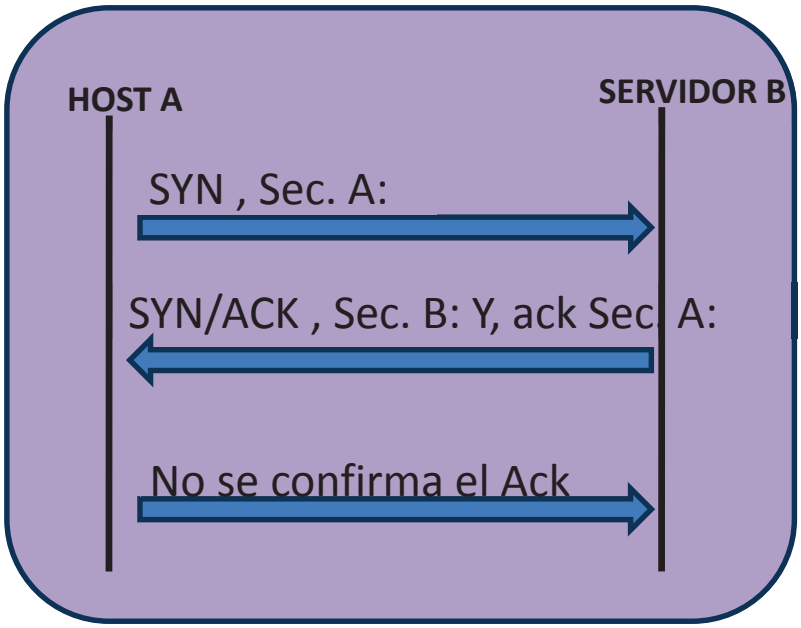


Figura 2-45 Comunicación Incompleta en conexión de 3 vías de TCP.

Herramienta: hping3

Se utilizará esta herramienta para enviar un flood de peticiones con el bit Syn activado, permitiendo aleatoriedad en las IPs origen falsificado. La sintaxis del comando es:

Hping3 –S HostObjetivo –flood –rand-source –d TamañoDatos –p PortDest.

2.5 GUÍA RESUMEN CAPÍTULO 2

En esta sección se detalla brevemente una guía resumen para los procedimientos desarrollados en el capítulo actual.

2.5.1 DESARROLLO DEL PERÍMETRO DE LA RED DE DATOS

En la Tabla 2-5 se detalla el Control, Elemento y Política de seguridad implementada para el desarrollo del perímetro de la Red de datos, bajo los lineamientos de la metodología Safe de Cisco.

Control	Elemento	Política de Seguridad
Construir el perímetro de red sobre una arquitectura de seguridad	Modulo Internet de la Empresa definido por la metodología Cisco Safe	Elementos del módulo: Servidor Firewall, Filtro URL, Web/Ftp, Dns.
Evitar accesos no autorizados y ataques a los recursos de red	Servidor Firewall	Filtrar los paquetes de entrada y salida del iptables para permitir solo el tráfico de los servicios en el perímetro de red.
Filtro de conexiones http	Servidor de Control de Contenido con Squid Proxy	Filtrar las direcciones URL para limitar el tráfico http
Bloquear las respuestas Dns y las transferencias de zona de servidores externos	Servidor Dns	Configuraciones de seguridad en la zona directa, inversa y en el archivo named.conf

Proteger contra ataques de desbordamiento Syn de TCP y limitar las respuestas a los clientes	Servidor Web/Ftp	Configuraciones de seguridad en el Kernel del servidor Firewall y en los archivos de configuración "httpd.conf" y "vsftpd.conf" para los servicios web y Ftp respectivamente.

Tabla 2-5 Control, Elemento y Políticas de Seguridad para el desarrollo del Perímetro de la Red de datos

2.5.2 PROCEDIMIENTRO FORMAL DE PRUEBAS DE INTRUSIÓN APLICADO AL PERÍMETRO DE RED

En las Tablas 2-6, 2-7, 2-8 y 2-9 se detallan los módulos, pruebas y herramientas utilizadas para el desarrollo del Procedimiento Formal de pruebas de intrusión según la fase recomendada por la metodología OSSTMM V3.

Fase	Módulo	Pruebas	Herramientas
Fase de Inducción	Módulo Logística	Framework	Whois, host, dig, Central Ops
		Calidad de la red	Apache Bench, ping
		Tiempo	Procedimiento manual, ping
	Módulo Verificación de detección Activa	Filtrado	Procedimiento manual
		Detección Activa	Nmap

Tabla 2-6 Módulos, Pruebas y Herramientas para el desarrollo del Toolkit de Pruebas de Intrusión en la Fase de Inducción

Fase	Módulo	Pruebas	Herramientas
Fase de Interacción	Auditoria de Visibilidad	Reconocimiento de la Red	Wireshark, Maltego, Nmap, ping, traceroute.
		Enumeración e Identificación	Firebug de Firefox, tail, hping2, whatweb, cms-Explorer, nmap.
	Verificación de Acceso	Acceso en la red, servicios y autenticación	Netcat, uname, Procedimientos manuales, nikto, SqlMap, XSSF, Metasploit Msf, Wireshark, Jhon the Ripper, Web Scarab.
	Verificación de Confianza	Spoofing, Pishing y Abuso de Recursos	Cain y Abel, Web Scarab, nic.ec

Tabla 2-7 Módulos, Pruebas y Herramientas para el desarrollo del Toolkit de Pruebas de Intrusión en la Fase de Interacción

Fase	Módulo	Pruebas	Herramientas
Fase de Investigación	Verificación de Configuración	Controles y errores de configuración	Procedimiento manual para validar los controles y errores de configuración
		Mapeo de Limitaciones	Procedimiento manual
	Verificación de Exposición	Enumeración de Exposición	Google Hacking

Tabla 2-8 Módulos, Pruebas y Herramientas para el desarrollo del Toolkit de Pruebas de Intrusión en la Fase de Investigación

Fase	Módulo	Pruebas	Herramientas
Fase de Intervención	Auditoría de Privilegios	Autorización y Escalación de Privilegios	Web Scarab, exploit Mempodipper
	Validación de Supervivencia	Resiliencia	Hping3

Tabla 2-9 Módulos, Pruebas y Herramientas para el desarrollo del Toolkit de Pruebas de Intrusión en la Fase de Intervención

CAPÍTULO 3: PRUEBAS Y RESULTADOS

En este capítulo se documenta en la sección 3.1, las pruebas y resultados de aplicar las herramientas de intrusión sobre los sistemas y servicios del perímetro de red. Al finalizar esta documentación en la sección 3.2, se realiza el análisis de resultados que permiten al analista de seguridad reducir los falsos positivos, negativos y errores humanos de las pruebas descritas en el presente proyecto.

3.1 DOCUMENTACIÓN DE PRUEBAS Y RESULTADOS

En esta sección se documenta los resultados de las pruebas de intrusión sobre los servicios, sistemas del Perímetro de Red conforme a la sección 2.4 anteriormente descrita. Para ello se ha dividido en las subsecciones 3.1.1 hasta la 3.1.4 para detallar las pruebas y resultados conforme a las 4 fases descritas por la metodología OSSTMM

3.1.1 RESULTADO DE PRUEBAS: FASE DE INDUCCIÓN

En esta sección se documenta los resultados de aplicar las pruebas de la fase de Inducción detalladas por la metodología OSSTMM V3. Los módulos desarrollados en esta fase son: Logística y Verificación de la Detección Activa.

3.1.1.1 Módulo Logística

Dentro de este módulo se documentará las pruebas: Framework de la red, Calidad de la red y tiempo.

Pruebas: Framework de la Red

a) Verificación de información del registro de dominio

En el proyecto actual se trabajó con el dominio local “iza.sanhueza.ec”, el cual no está publicado en internet, por lo cual no se puede probar con los motores de búsqueda de bases de datos de internet como es el caso de Whois y Central Ops. Entonces se realizó estas pruebas con las herramientas host y dig para proporcionar información detallada del dominio registrado localmente. En la figura 3-1 se visualiza el resultado de aplicar el comando dig al dominio local, el cual Indica los registros tipo A, SOA, y el ttl de propagación asociados al dominio.

Figura 3-1 Herramienta dig aplicada al dominio local “iza.sanhueza.ec”

En la figura 3-2 se verifica el resultado de aplicar la herramienta host –a4 al dominio auditado, el cual indica información adicional como los registros SOA, NS[60] y tipo A.

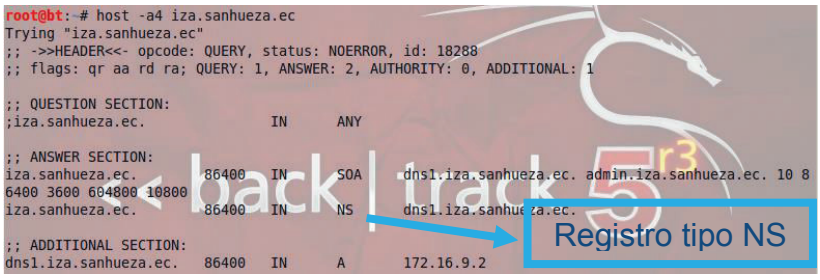


Figura 3-2 Herramienta host –a4 al dominio local “iza.sanhueza.ec”

[60] Registro tipo NS: Se utiliza para indicar la zona para los servidores autoritativos de nombre de dominio. Fuente: www.queesdns.com/ns

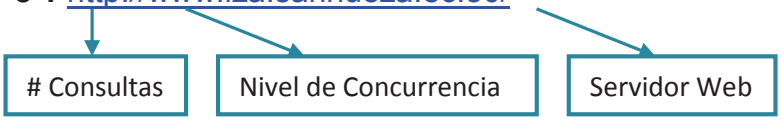
Pruebas: Calidad de la Red

c) Medición del índice de velocidad y pérdidas de paquetes en el alcance de un servicio solicitado en TCP e ICMP, tanto como solicitud de servicio como respuesta. Repetir cada solicitud en sucesión por lo menos 100 veces y anotar el promedio tanto para las solicitudes de servicio como respuestas de paquetes, para cada protocolo.

Se utilizará la herramienta Apache Bench para probar el servidor web en base al protocolo http (puerto 80) indicado en la Figura 3-3 y la herramienta ping para cumplir el procedimiento con el protocolo ICMP indicado en la Figura 3-4. Para la prueba de la figura 3-3, la información más importante que se revela es la siguiente:

- Software, nombre y puerto del servidor web testeado.
- Nivel de Concurrencia: En este caso 1 que indica la simultaneidad de acceso al servidor web.
- El tiempo transcurrido en la prueba.
- Número de consultas realizadas al servidor web, en este caso 100.
- Número de errores en caso que existieran.
- Total de bytes transferidos en la consulta, en este caso 40700 bytes.
- Número de consultas por segundo, el promedio del tiempo que toma realizar una consulta, la tasa de transferencia.
- El tiempo en milisegundos (ms) para la conexión mínima, media, máxima y promedio.
- Porcentaje de las peticiones servidas en un tiempo determinado.

Comando: ab –n 100 –c 1 http://www.iza.sanhueza.ec:80/



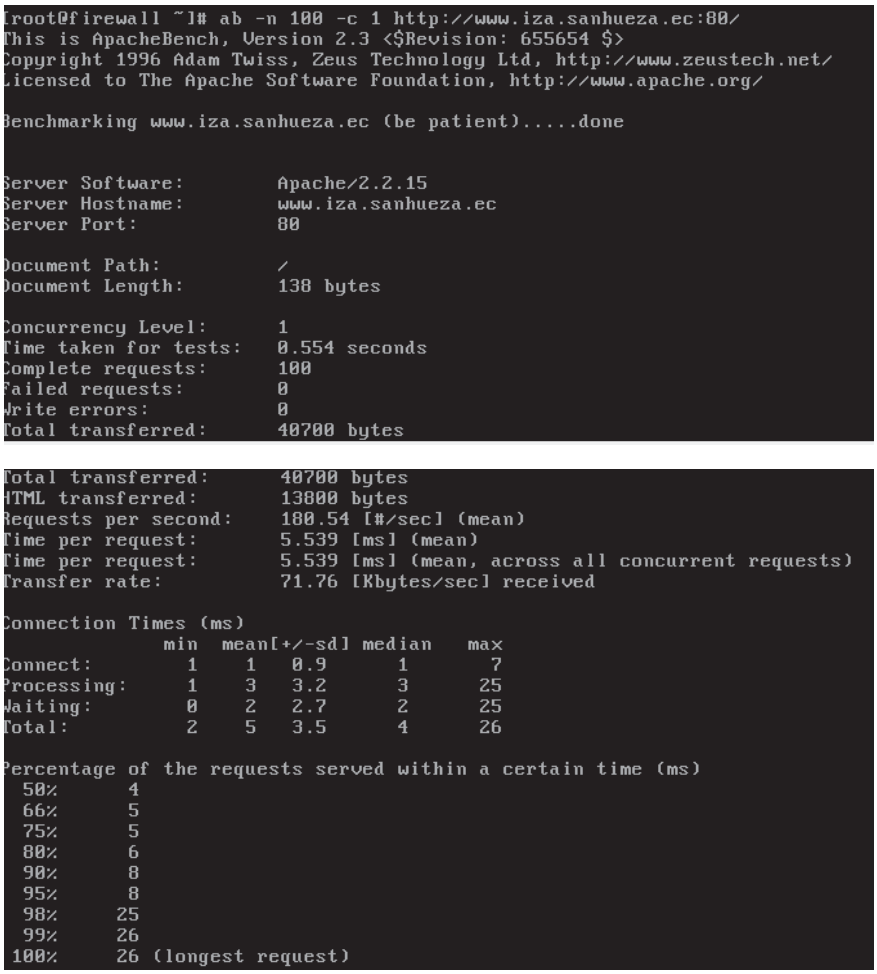


Figura 3-3 Herramienta Apache Bench para prueba de solicitud Http

En la Figura 3-4 se tiene el resultado de aplicar la herramienta Ping para probar solicitudes ICMP en sucesión de 100 veces hacia el servidor web, obteniendo resultados como la tasa de transferencia mínima, promedio, máxima, así como los paquetes transmitidos, recibidos y perdidos.

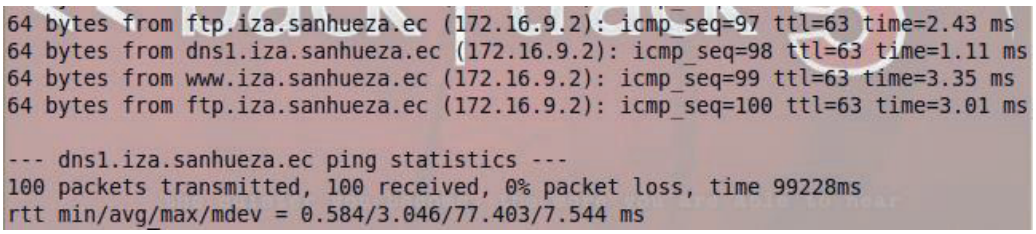


Figura 3-4 Herramienta ping para prueba de solicitud ICMP

d) Determinación de envío y recepción de las tasas de paquete para un total de 6 medias por protocolo.

Para cumplir esta prueba, se realiza el mismo procedimiento del literal c utilizando la herramienta Apache Bench para probar las solicitudes http con una media de 6 veces.

Resultado: (79.43+88.93+84.93+59.99+83.24+71.76) / 6 = 78.04 kbytes/sec.

Pruebas: Tiempo

e) Identificar el Time To Live (TTL) al Gateway y los objetivos.

1. TTL al Gateway

En la figura 3-5 se muestra el resultado de aplicar el comando ping al Gateway (Firewall), donde el TTL toma el valor de 64.

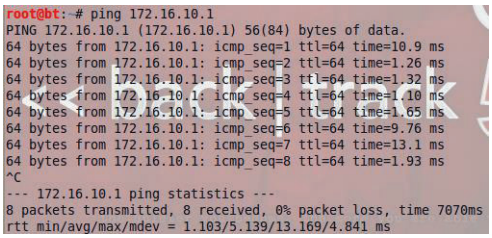


Figura 3-5 TTL al Gateway

2. TTL al servidor Web/Ftp/Dns

En la figura 3-6 se muestra el resultado de aplicar el comando Ping al Servidor Web/Ftp/Dns, donde el TTL toma el valor de 63, lo que indica que el Servidor está a 2 saltos de la máquina de pruebas.

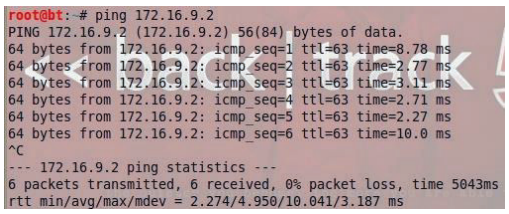


Figura 3-6 TTL al Servidor Web / Ftp / Dns

3.1.1.2 Módulo Verificación de la Detección Activa

En este módulo se documenta las pruebas: Filtrado y Detección Activa.

Pruebas: Filtrado

h) Probar si los datos de red entrantes y salientes o comunicaciones a través de Web, mensajería instantánea, chat, basados en la web o foros, e-mail, se monitorean o se filtran por una parte autorizada para la retransmisión de materiales inadecuados, inyecciones de código, contenido malicioso, conducta impropia, así como el registro y tiempo de respuesta.

Las políticas de filtrado están definidas a nivel de control de contenido web por el Squid Proxy Server y el Firewall de la red con el Sistema Iptables.

1. Control de Contenido

En la figura 3-7 se verifica la sección de Control de acceso del Servidor Squid Proxy, que contiene la regla llamada Redes Sociales, la cual realiza filtrado http en base a las palabras definidas en este archivo. En la figura 3-8 se verifica la aplicación de esta regla llamada Redes Sociales mediante la acción Deny.

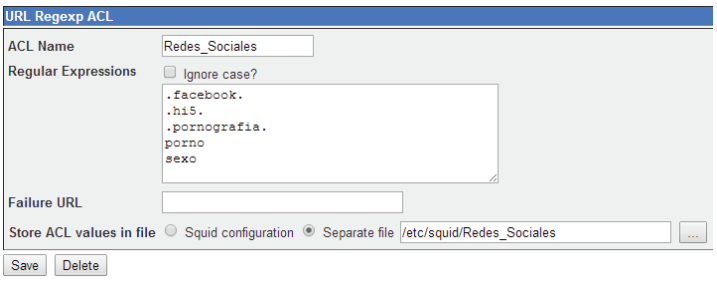


Figura 3-7 **Filtro del Squid Proxy Server llamada "Redes Sociales"**

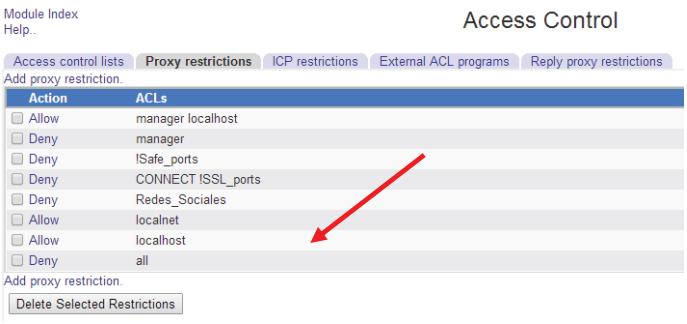


Figura 3-8 **Denegación de Regla "Redes Sociales" en Proxy restrictions**

2. Firewall Iptables del servidor

El firewall del servidor está implementado bajo el software iptables de Linux, el cual maneja las secciones Filter y Nat.

Sección Filter

En la figura 3-9 se verifica el flujo de tráfico permitido cuando el destino de puertos TCP, sean los siguientes: 25, 20:21, 80, 110, 22, 11000, 53, 23. Adicionalmente detalla que acepta las conexiones si la interfaz de entrada es la eth1 (Segmento de red del Servidor Web) y la eth2 (Segmento de red de las máquinas de pruebas).

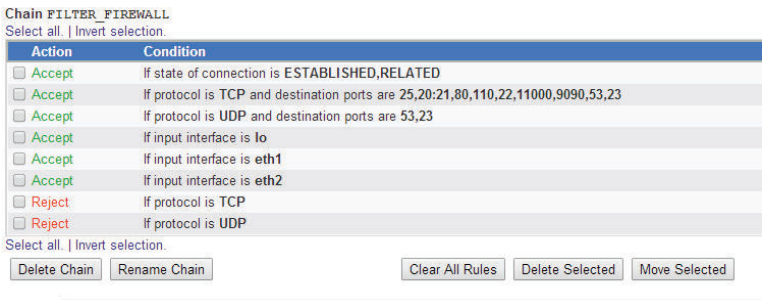


Figura 3-9 Sección Filter del Firewall Iptables.

Sección Nat

Incluye 2 cadenas para los paquetes entrantes y salientes de la tabla de enrutamiento del servidor. En la cadena Prerouting se define el re direccionamiento

del puerto 80 al 3128, a fin de re direccionar las peticiones http hacia los filtros del Squid Proxy Server. En la cadena Postrouting se enmascaran los puertos necesarios para permitir la comunicación hacia los servicios externos, y enmascarar la conexión a internet para los segmentos Lan (Servidor Web 172.16.9.0/24, Máquinas de pruebas 172.16.10.0/24). En la figura 3-10 se verifica esta configuración.

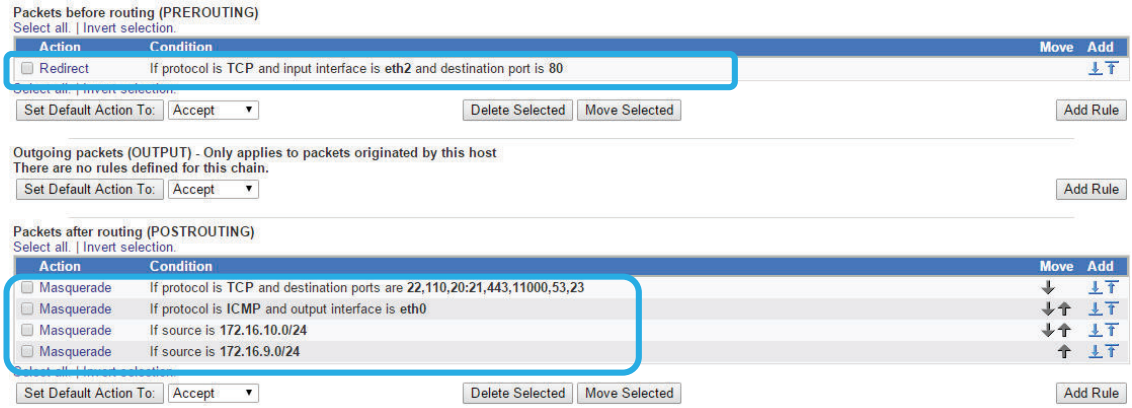


Figura 3-10 Sección Nat del Firewall Iptables.

Pruebas: Detección Activa

i) Verificar las respuestas activas a las pruebas de los sistemas y servicios. Esto podría ser notificaciones legible humano o máquina, respuestas de paquetes, viajes silenciosos de alarma, o similares.

Para cumplir con este literal, se utiliza la Herramienta Nmap, para verificar las respuestas ante el sondeo de solicitudes a los sistemas dentro del perímetro de red.

Comando: Nmap –T4 –A –v Objetivo

Opción –T4 indica el tiempo que tomará reflejar los resultados de la prueba.

Opción –A muestra información detallada, tal como versión, sistema operativo, etc.

Opción –v indica que se incrementa el nivel de detalle.

1. Objetivo: Servidor Web/Ftp/Dns

En la figura 3-11 se muestra el resultado de la prueba utilizando la herramienta Nmap al servidor Web/Ftp/Dns. Los resultados indican la siguiente información:

1. El tiempo de la prueba
2. Número y Servicios activos en el servidor: Detalla puerto, estado, servicio y versión
3. Puertos abiertos: 53, 21, 2, 22, y 80
4. Traceroute hacia el servidor
5. Latencia al servidor
6. Plataforma de enumeración del Sistema Operativo OS CPE
7. Información general del Sistema Operativo
8. Tiempo de la prueba y paquetes enviados y recibidos.

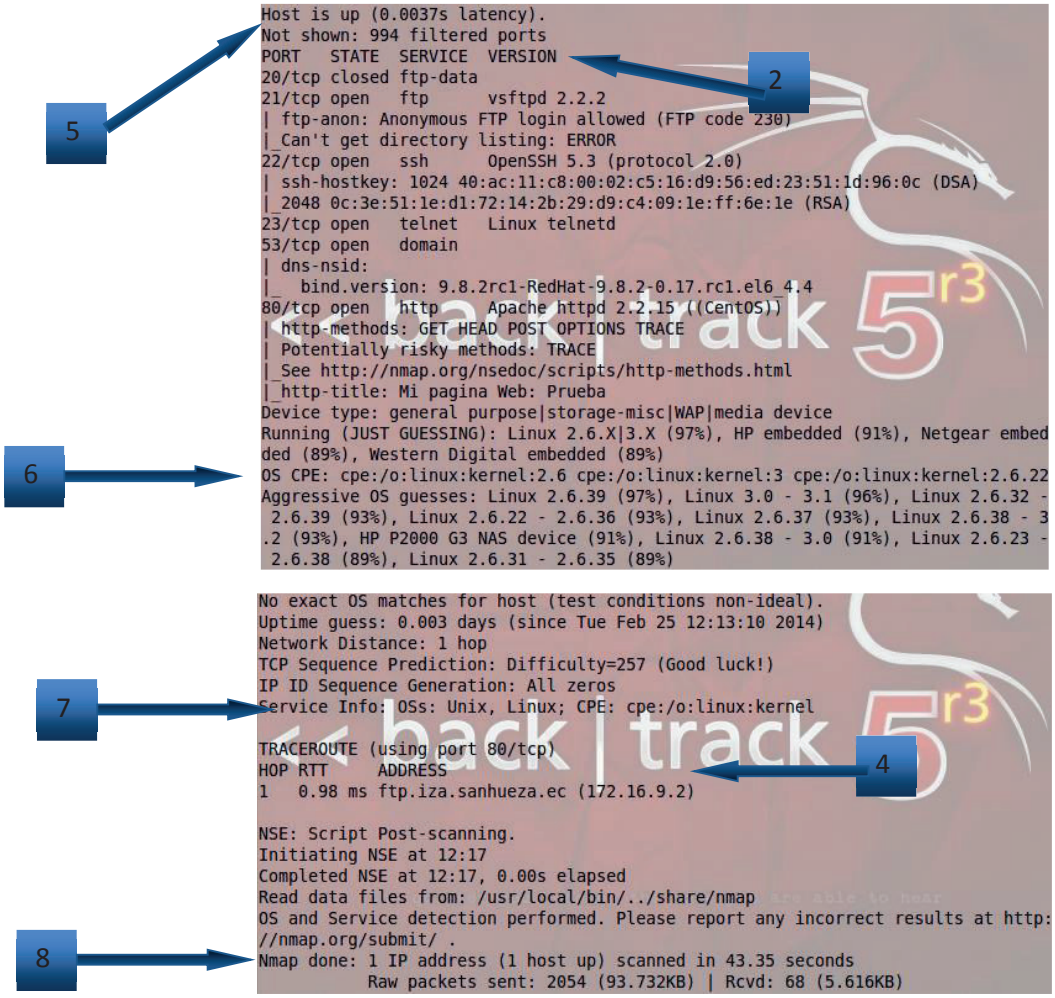


Figura 3-11 Nmap al Servidor Web/Ftp/Dns

2. Objetivo: Servidor Firewall / Control de Contenidos

En la figura 3-12 se muestra el resultado de la prueba utilizando la herramienta Nmap al Servidor Firewall/Control de Contenido. Se determina la siguiente información:

1. El tiempo de la prueba
2. Número y Servicios activos en el servidor:
3. Puertos abiertos: 23, 80, 22, 3128
4. Traceroute hacia el servidor
5. Latencia hacia el servidor

6. Plataforma de enumeración del Sistema Operativo OS CPE
7. Información general del Sistema Operativo
8. Tiempo de la prueba y paquetes enviados y recibidos

Figura 3-12 Nmap al Servidor Firewall / Control de Contenido

3.1.2 RESULTADO DE PRUEBAS: FASE DE INTERACCIÓN

En esta sección se documenta los resultados de aplicar las pruebas de la fase de interacción, detalladas por la metodología OSSTMM V3. Los módulos desarrollados son: Auditoría de Visibilidad, Verificación de Acceso y Verificación de Confianza.

3.1.2.1 Módulo Auditoría de Visibilidad

Dentro de este módulo se documenta las pruebas: Survey de la Red, Enumeración e Identificación.

Pruebas: Survey de la Red

b) Usar olfateo de la red para identificar emanación de protocolos desde respuestas de servicio de red o solicitudes donde son aplicables. Por ejemplo, NetBIOS, ARP, SAP, NFS, BGP, OSPF, MPLS, RIPv2, etc.

A continuación se detallan las pruebas realizadas, utilizando la máquina de pruebas Backtrack, aplicada a la interfaz Wan "eth0"**.**

1. En la figura 3-13 se verifica el establecimiento de la conexión TCP cuando solicitamos la Página web local **www.iza.sanhueza.ec**. Existe un intercambio de 3 mensajes: la petición "Get" desde la máquina de pruebas, la respuesta "Http Response" desde el Servidor Web y el Ack para acusar la recepción correcta del paquete response.

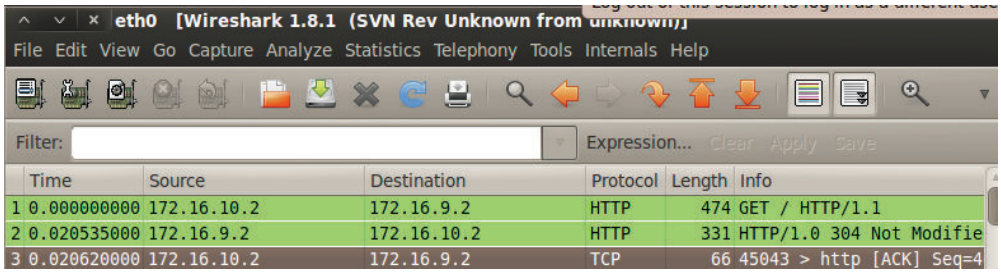


Figura 3-13 Captura Wireshark en la petición de acceso a la página Web.

2. En la figura 3-14 se verifica los paquetes intercambiados en una conexión Ping hacia la página local **www.iza.sanhueza.ec**. El primer paquete del tipo ICMP "echo request" envía la solicitud al servidor web, el segundo del tipo "echo reply" es la respuesta desde el servidor Web y el tercero del tipo DNS detalla la traducción del nombre de dominio **www.iza.sanhueza.ec** a la dirección IP correspondiente.

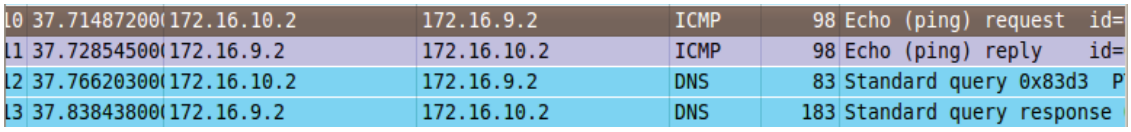


Figura 3-14 Captura Wireshark ante la conexión Ping a la Página Web.

3. En la figura 3-15 se verifica los paquetes en una conexión Ping hacia el Servidor Firewall. Se valida el intercambio de 2 mensajes: ICMP Echo request y Echo reply para el envío y respuesta de la solicitud Ping al Servidor Firewall.

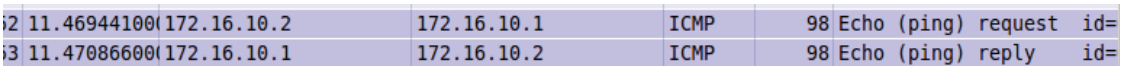


Figura 3-15 Captura Wireshark ante la conexión Ping al firewall.

d) Verificar las solicitudes de broadcast y las respuestas de todos los objetivos.

En la figura 3-16 se muestra el resultado de utilizar la Herramienta Wireshark para filtrar el tráfico de broadcast mediante el filtro respectivo. El monitoreo se aplica a la interfaz Wan de la máquina de pruebas Backtrack.

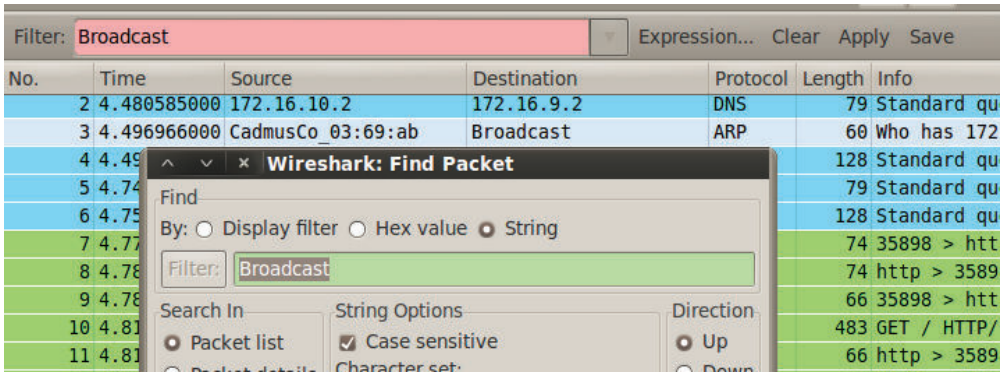


Figura 3-16 Herramienta Wireshark con filtro Broadcast.

e) Verificar y examinar el uso de tráfico y protocolos de enrutamiento para todos los objetivos.

Para cumplir con este apartado se hará uso de la Herramienta Nmap con el argumento –s0 para realizar una exploración de protocolos IP en los objetivos del Perímetro de red.

1. Objetivo: Servidor Web/Ftp/Dns

En la figura 3-17 se verifica el resultado de escanear los protocolos IP con la herramienta Nmap aplicado al Servidor Web/Ftp/Dns. Se valida que el protocolo IP – ICMP está abierto.

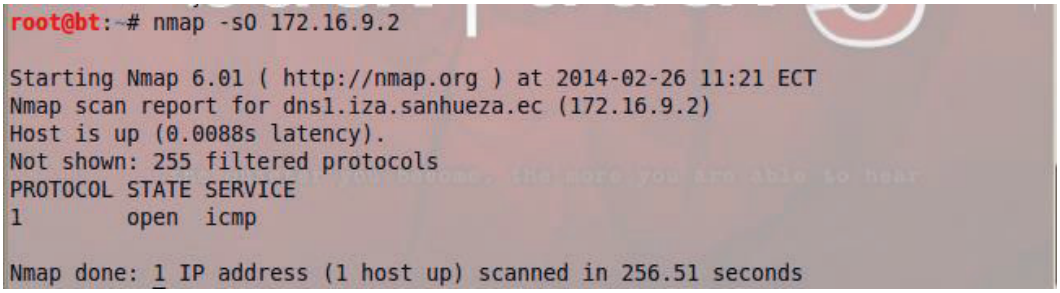


Figura 3-17 Nmap argumento –So en el servidor Web.

2. Objetivo: Servidor Firewall / Control de Contenido.

En la figura 3-18 se verifica el resultado de escanear los protocolos IP con la herramienta Nmap aplicado al Servidor Firewall/Control de Contenido. Se confirma los protocolos: ICMP, IGMP[61], TCP, UDP, PIM[62], UDPLITE[63] abiertos.

---

61 El protocolo de red IGMP se utiliza para intercambiar información acerca del estado de pertenencia entre enrutadores IP que admiten la multidifusión y miembros de grupos de multidifusión. Fuente: http://es.wikipedia.org/wiki/IGMP.

62 Protocol Independent Multicast (PIM), es un Protocolo de encaminamiento que crea una estructura de árbol de distribución entre los clientes multicast formando dominios. Fuente: http://es.wikipedia.org/wiki/Pim.

63 UDP Lite es un protocolo sin conexión que permite que una carga útil de datos potencialmente con errores sea entregado a una aplicación en lugar de ser desechado por la estación receptora. Fuente: http://en.wikipedia.org/wiki/UDP_Lite

Figura 3-18 Nmap argumento –So en el servidor Firewall.

f) Verificar las respuestas ICMP de los tipos ICMP 0-255 y códigos ICMP 0-2 de todos los objetivos.

Se utiliza la herramienta Nmap con los argumentos PE, PP y PM para verificar las respuestas de los sistemas del perímetro de red, ante determinados tipos de paquetes y código ICMP. En la figura 3-19 se verifica el uso de la herramienta Nmap con el argumento –PE hacia los objetivos Servidor Web/Ftp/Dns y Firewall/Control de Contenido. El resultado indica información de los servicios y puertos abiertos, correspondiente al tipo ICMP 15.

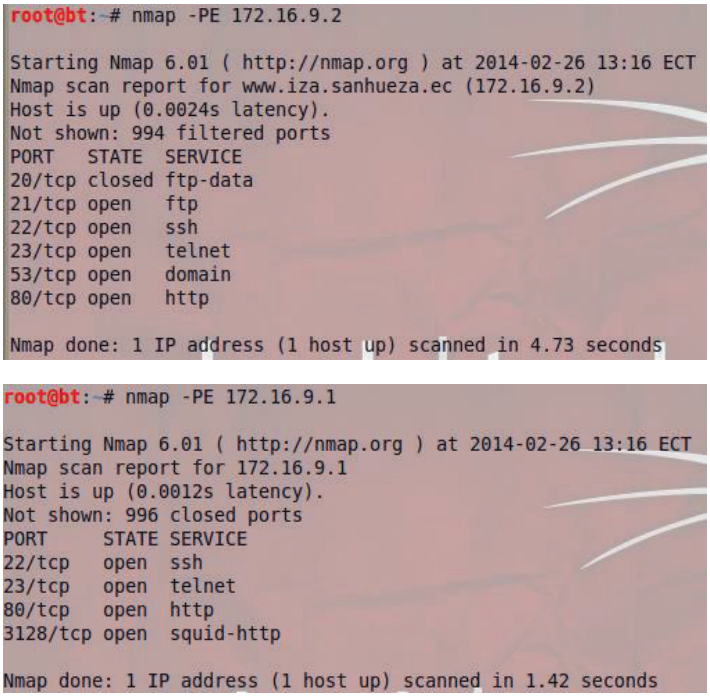


Figura 3-19 Herramienta Nmap –PE

En la figura 3-20 se verifica el uso de la herramienta Nmap con el argumento –PP hacia los objetivos Servidor Web/Ftp/Dns y Firewall/Control de Contenido. El resultado indica información de la huella de tiempo en que se realiza la prueba, correspondiente al tipo ICMP 13.

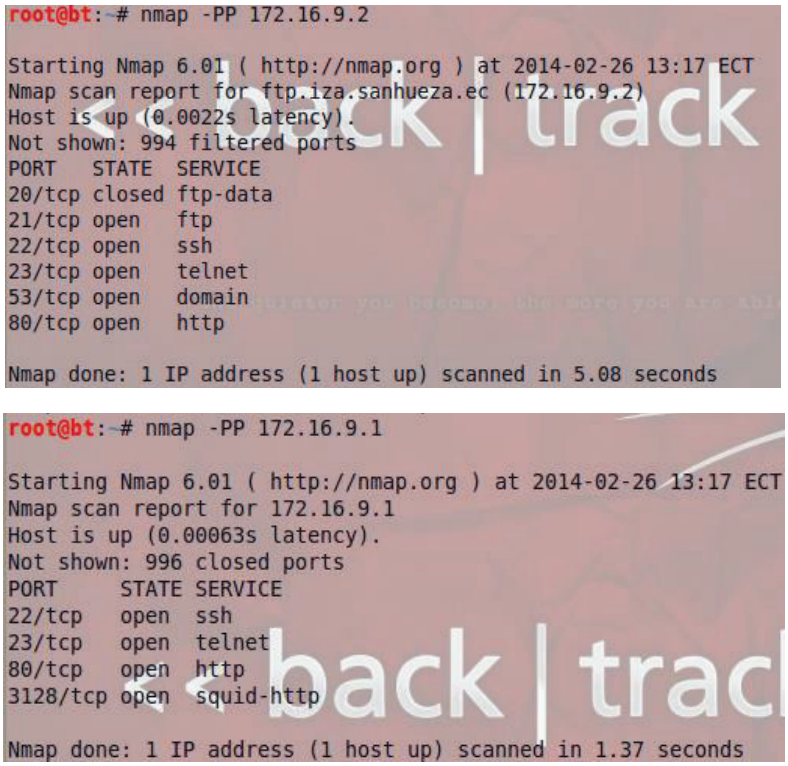


Figura 3-20 Herramienta Nmap –PP

En la figura 3-21 se verifica el uso de la herramienta Nmap con el argumento –PM hacia el objetivo Firewall/Control de Contenido. El resultado muestra información de la máscara de red de los hosts que están realmente activos, pero valida que está bloqueada este tipo de información, correspondiente al tipo ICMP 18.

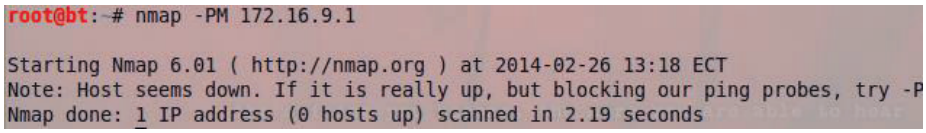


Figura 3-21 Herramienta Nmap–PM

g) Verificar los nombres por defecto y nombres de comunidades SNMP en uso que estén de acuerdo con las implementaciones de todas las versiones de SNMP.

Herramienta Wireshark

En la Figura 3-22 se aplica la opción de filtrado con la palabra SNMP para verificar el flujo de datos proveniente de este protocolo.

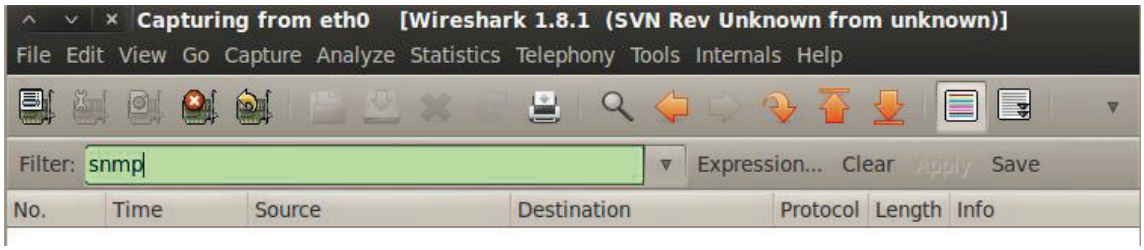


Figura 3-22 Wireshark con filtro snmp.

Herramienta Nmap

El argumento –Su nos permite filtrar la búsqueda por puertos UDP. En la figura 3-23 se verifica el resultado de la prueba aplicado al Servidor Web/Ftp/Dns, el cual confirma el puerto 53 Dns abierto y el puerto 23 telnet cerrado.

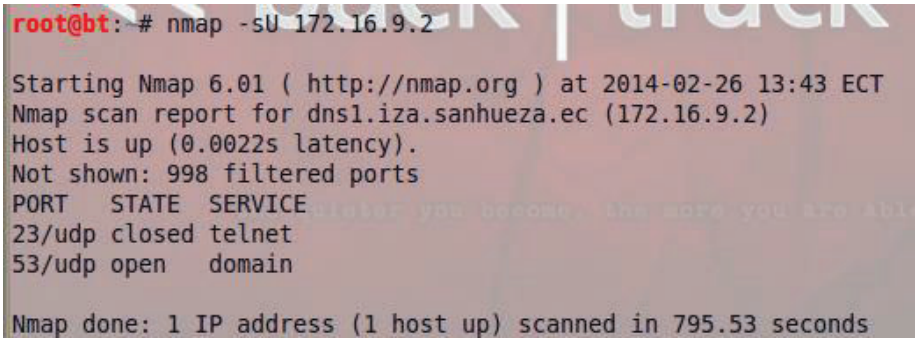


Figura 3-23 Herramienta Nmap –su al servidor Web/Ftp/Dns

h) Verificar las respuestas de los objetivos al seleccionar los puertos con expiración TTL de vencimiento a menos de 1 y 2 saltos de los objetivos. Por ejemplo:

TCP 8, 22, 23, 25, 80, 443, 445, 1433

UDP 0, 53, 139, 161

ICMP T00: C00, T13: C00, T15: C00, T17: C00

Conociendo la funcionalidad del mensaje ICMP tipo 11 Time Exceeded, se utiliza la herramienta ping con el argumento –t para definir el número de saltos TTL e indicar los equipos intermedios para alcanzar los objetivos.

1. Pruebas con ttl= 2 hacia el Servidor Dns/Ftp/Web y al Firewall/Control de Contenido

En la figura 3-24 se tiene el resultado de aplicar la herramienta ping con el ttl establecido en 2 hacia el Servidor Dns/Ftp/Web. El comando ping hace uso de los tipos de mensaje ICMP 0 para Echo Request y 8 para Echo Reply, confirmando conectividad estable con tiempo promedio de 1 ms y ttl establecido en 63 ya que atraviesa únicamente un host (el firewall) para alcanzar el objetivo. El TTL por defecto para esta conexión TCP usa el valor 64.

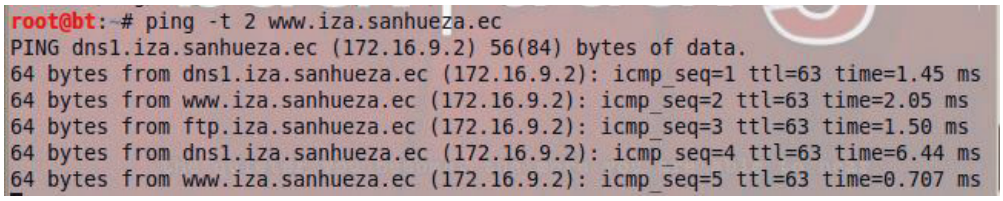


Figura 3-24 Prueba ping con ttl=2 hacia Servidor Dns/Web/Ftp

En la figura 3-25 se tiene el resultado de aplicar la herramienta ping con el ttl establecido en 2 hacia el Servidor Firewall/Control de Contenido. El comando ping hace uso de los mensajes ICMP tipo 0 para Echo Request y tipo 8 para Echo Reply, confirmando conectividad estable con tiempo promedio de 1 ms y ttl establecido en 64, por lo cual no atraviesa ningún equipo intermedio ya que el firewall está

conectado directamente a la interfaz eth0 de la máquina de pruebas Backtrack. El TTL por defecto para esta conexión TCP usa el valor 64.

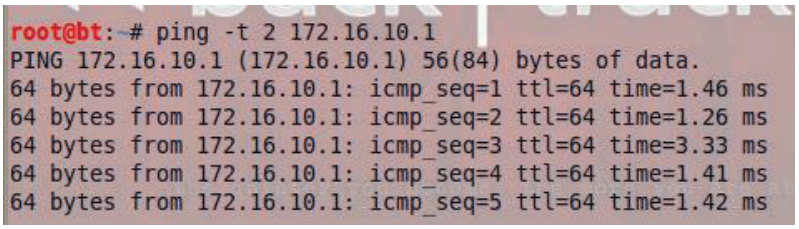


Figura 3-25 Prueba ping con ttl=2 hacia Servidor Firewall/Control de Contenido

2. Pruebas con ttl= 1 hacia el Servidor Dns/Ftp/Web y al Firewall/Control de Contenido

En la figura 3-26 se tiene el resultado de aplicar el comando ping con el ttl establecido en 1 al Servidor Dns/Ftp/Web. El comando ping hace uso del mensaje ICMP tipo 11 para Time Exceeded y código 0 para TTL excedido en tránsito, por lo cual el TTL alcanzó el valor de 0 y transmitió al origen el mensaje Time to live exceded. Esto debido a que se necesitan 2 saltos para poder llegar al Server Web/Ftp/Dns.

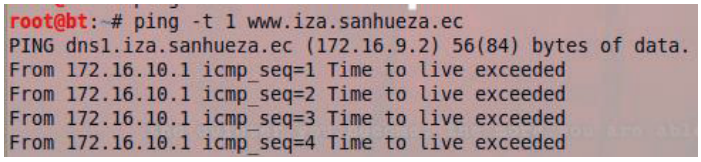


Figura 3-26 Prueba ping con ttl=1 hacia Servidor Dns/Web/Ftp

En la figura 3-27 se tiene el resultado de aplicar la herramienta ping con el ttl establecido en 1 al Servidor Firewall/Control de Contenido. El comando ping hace uso de los mensajes ICMP tipo 0 para Echo Request y 8 para Echo Reply, confirmando conectividad estable con tiempo promedio de 1 ms y ttl establecido en 64, lo que indica que no atraviesa ningún equipo intermedio ya que el firewall está conectado directamente a la interfaz eth0 de la máquina de pruebas Backtrack. El TTL por defecto para esta conexión TCP usa el valor 64.

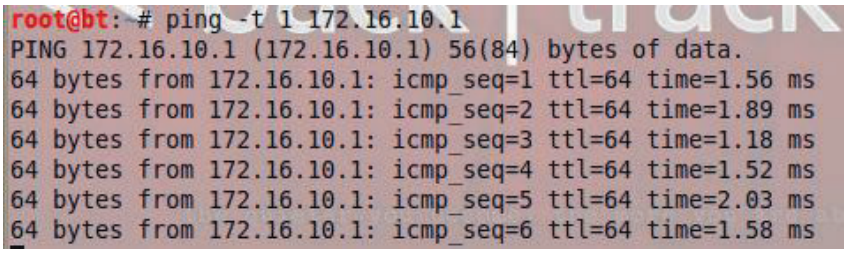


Figura 3-27 Prueba ping con ttl=1 hacia Servidor Firewall/Control de Contenido

l) Trazar la ruta de los paquetes ICMP a todos los destinos.

La Herramienta Traceroute se utiliza para verificar la traza de los paquetes ICMP tanto al servidor Web/Ftp/Dns en la figura 3-28 y al Servidor Firewall/Control de Contenido en la figura 3-29.

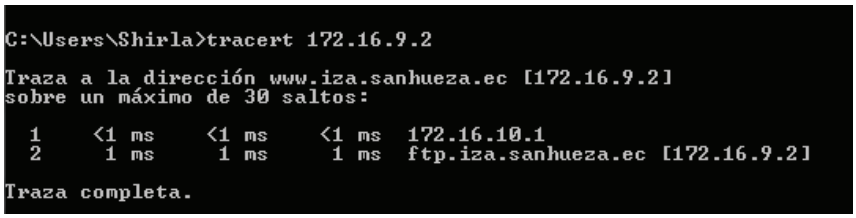


Figura 3-28 Traza de paquetes ICMP al Servidor Web/Ftp/Dns

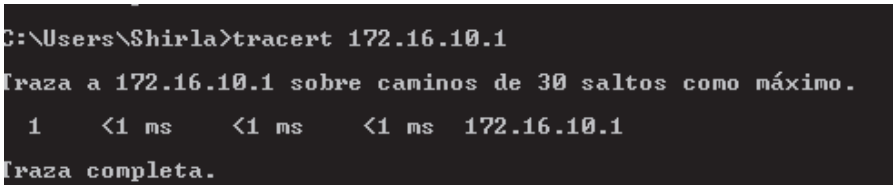


Figura 3-29 Traza de paquetes ICMP al Servidor Firewall/Control de Contenido

j) Trace la ruta de los paquetes TCP a todos los objetivos para los puertos SSH, SMTP, HTTP, HTTPS.

Para cumplir esta sección se utilizará la herramienta Traceroute con el argumento –p, el cual define el puerto del servicio a probar.

1. Objetivo: Servidor Web /Ftp/Dns

En la Figura 3-30 se verifica la ruta hacia el servidor, con el argumento –p que especifica el puerto TCP para la prueba. En todas las respuestas se verifica completar la traza.

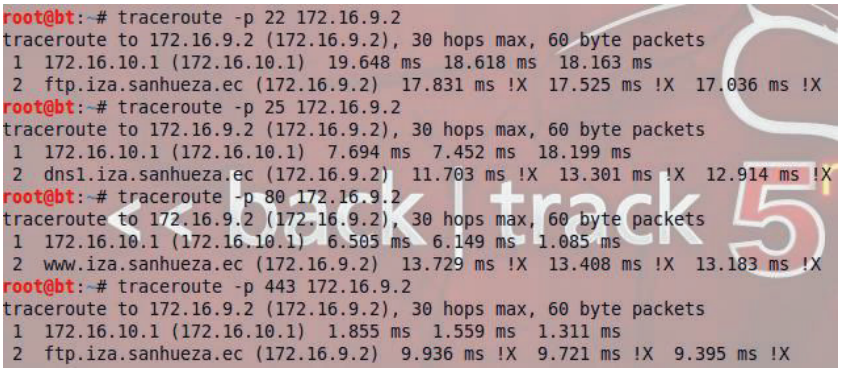


Figura 3-30 Traza de paquetes TCP al Servidor Web

2. Objetivo: Servidor Firewall / Control de Contenidos

En la Figura 3-31 se verifica la traza hacia el servidor, con el argumento –p para especificar el puerto TCP en la prueba. En todas las respuestas se valida completar la traza.

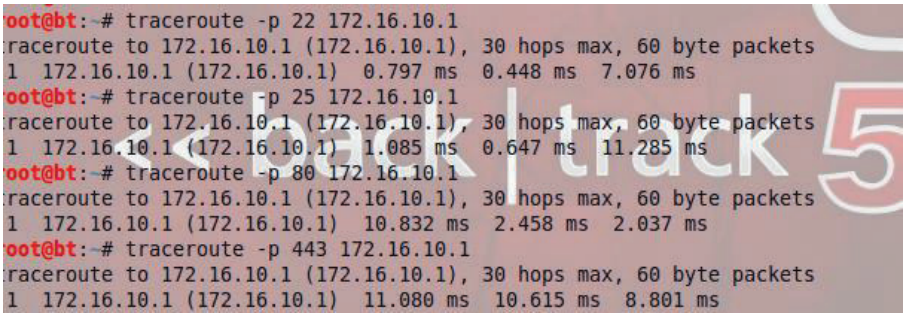


Figura 3-31 Traza de paquetes TCP al Servidor Firewall

k) Trace la ruta de los paquetes UDP a todos los objetivos para los puertos DNS y SNMP.

Se utilizará la herramienta Traceroute con los argumentos –P para definir el protocolo UDP, y el argumento –p para definir el puerto.

1. Objetivo: Servidor Web/Ftp/Dns

En la figura 3-32 se visualiza la traza de los paquetes UDP al servidor Web/Ftp/Dns con los puertos UDP: Dns y Snmp.

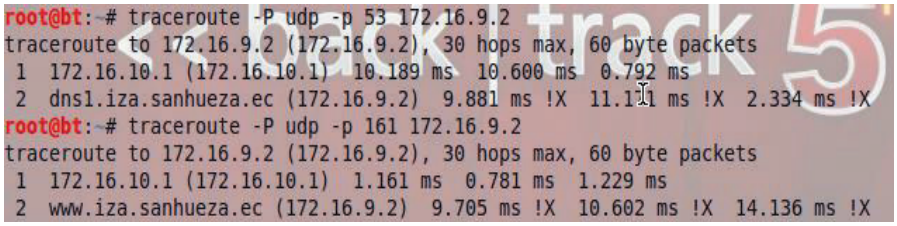


Figura 3-32 Traza de los paquetes UDP al Servidor Web.

2. Objetivo: Servidor Firewall/Control de Contenidos

En la figura 3-33 se aprecia la traza de los paquetes UDP al servidor Firewall/Control de Contenido con los puertos UDP: Dns y Snmp.

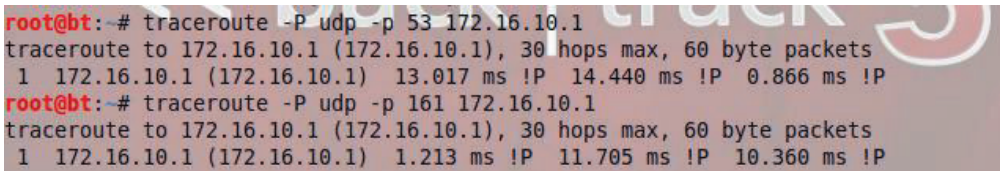


Figura 3-33 Traza de los paquetes UDP al Servidor Firewall

I) Identificar el número de predictibilidad de secuencia TCP ISN para todos los objetivos.

Se utilizará la Herramienta Wireshark para verificar los números de secuencia empleados por los paquetes hacia cada objetivo del perímetro de red.

1. Objetivo: Servidor Web/Ftp/Dns

Con una prueba de petición FTP al servidor 172.16.9.2, se captura los paquetes intercambiados en el establecimiento de la conexión TCP. El primer segmento TCP (Figura 3-34) activa la bandera SYN, y confirma el número de secuencia inicial TCP-ISN=0. El segundo paquete enviado desde el servidor (Figura 3-35) acusará este número de secuencia inicial con ACK igual a 1, y su propio número de secuencia

inicial en 0. El tercer paquete (Figura 3-36) envía el flag ACK activado con el ack del servidor 1, y el ack propio en 1.

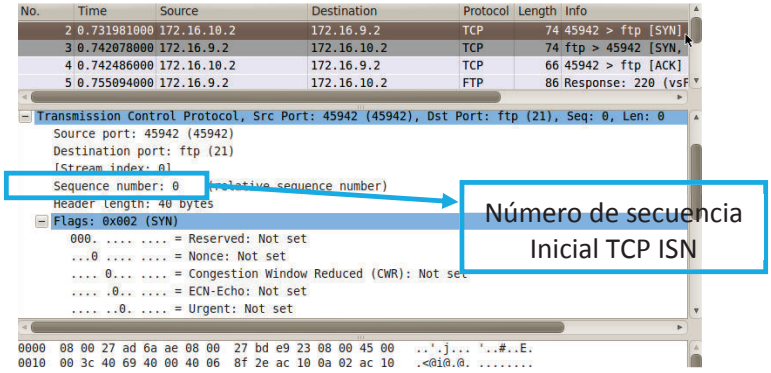


Figura 3-34 Primer segmento TCP en el establecimiento de conexión al Servidor Web/Ftp/Dns

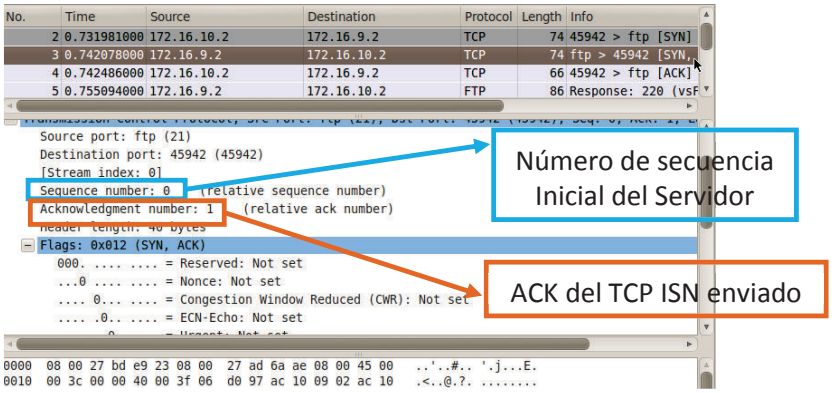


Figura 3-35 Segundo segmento TCP en el establecimiento de conexión al Servidor Web/Ftp/Dns

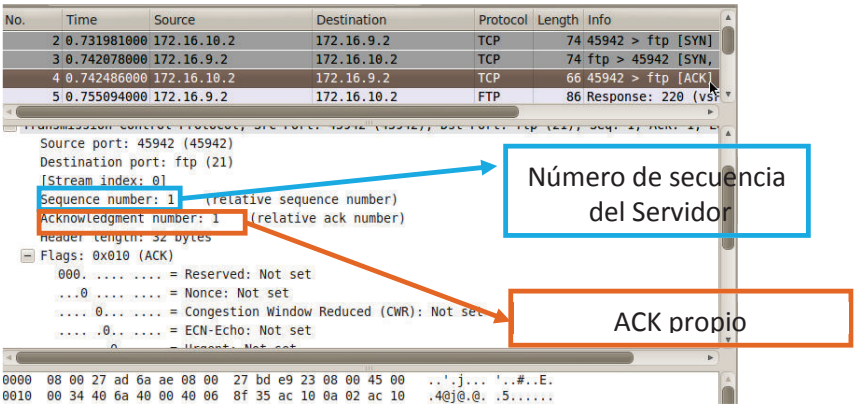


Figura 3-36 Tercer segmento TCP en el establecimiento de conexión al Servidor Web/Ftp/Dns

2. Objetivo: Servidor Firewall/Control de Contenido

De manera similar en una conexión SSH al servidor firewall 172.16.10.1, se intercambian 3 paquetes en el establecimiento de la conexión. En las figuras 3-37, 3-38 y 3-39 se detallan los 3 paquetes intercambiados.

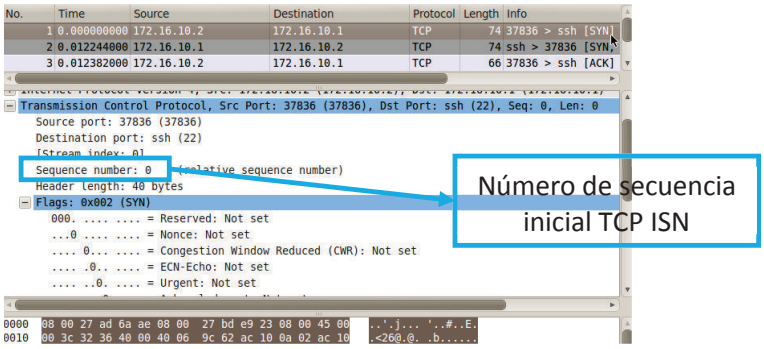


Figura 3-37 Primer segmento TCP en el establecimiento de conexión al Servidor Firewall

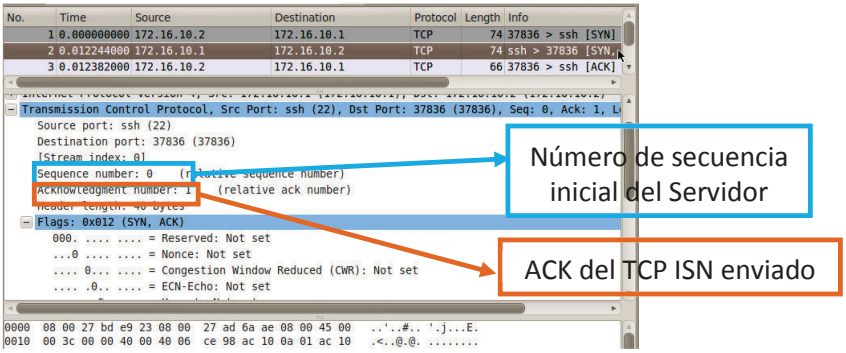


Figura 3-38 Segundo segmento TCP en el establecimiento de conexión al Servidor Firewall

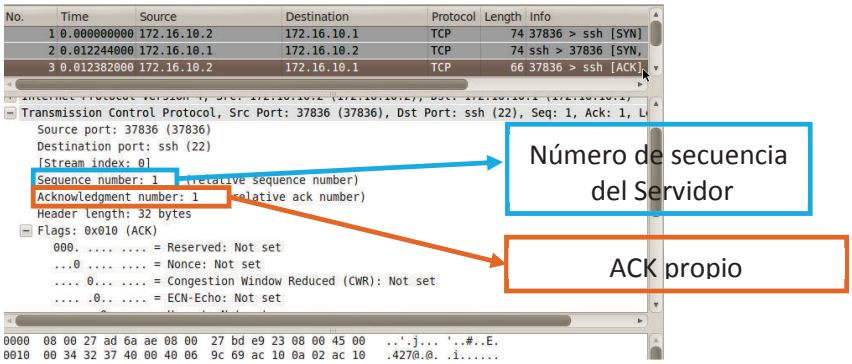


Figura 3-39 Tercer segmento TCP en el establecimiento de conexión al Servidor Firewall

Pruebas: Enumeración

n) Examinar el código fuente de la aplicación destino basada en web y scripts para determinar la existencia de objetivos adicionales en la red.

Se utilizará la Herramienta Firebug de Firefox, para analizar el código fuente de la página Web local “www.iza.sanhueza.ec”. El código fuente contiene 2 secciones: cabecera y cuerpo. En la figura 3-40 se muestra la información de la cabecera “head”, la cual está compuesta por las siguientes subsecciones:

- Referencia al sitio web **http://www.iza.sanhueza.ec**
- Tipo de CMS: Joomla
- La configuración del template en uso para la interfaz principal del sitio web.

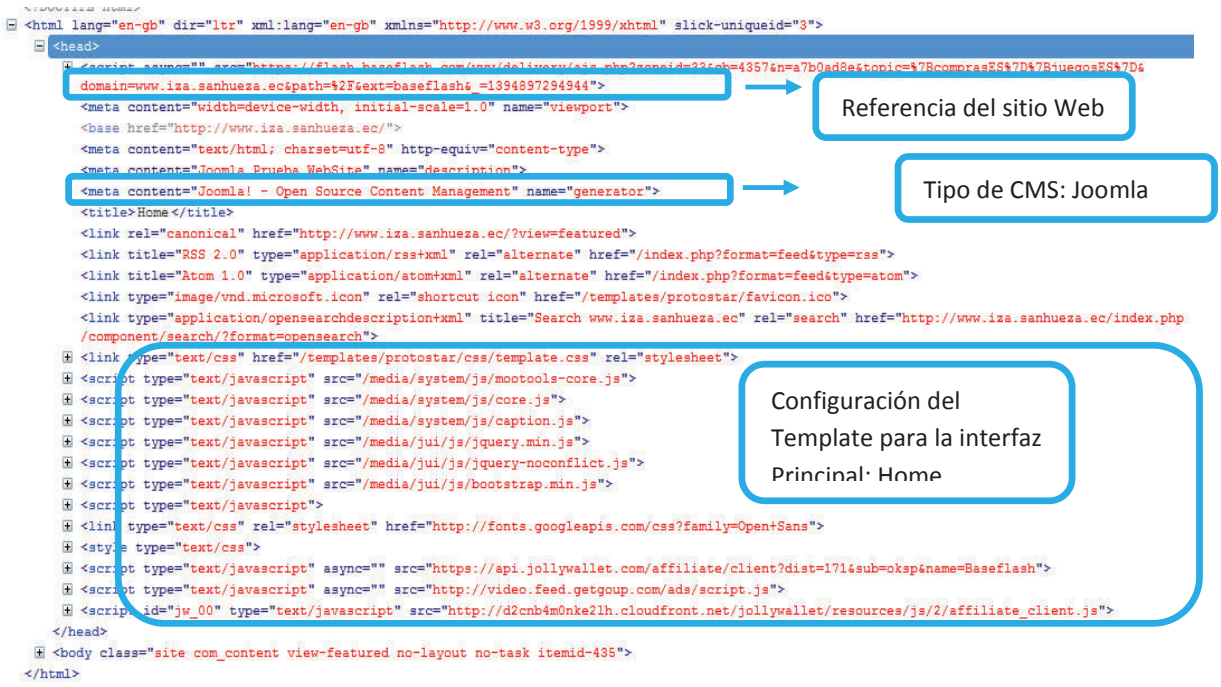


Figura 3-40 Cabecera del código fuente HTML.

En la figura 3.41 se visualiza la información del cuerpo “body” del código fuente, el cual está compuesto por las siguientes subsecciones:

- El contenido está estructurado mediante div class.

- El div class llamado body, contiene la estructura general del cuerpo con la información de la interfaz principal de la página web.

Figura 3-41 Cuerpo del código fuente HTML.

o) Buscar logs web y de intrusión para el sistema de la red destino.

Se utiliza la herramienta Tail con el argumento -f para verificar los logs en tiempo real de los sistemas en el perímetro de red.

1. Objetivo: Firewall / Iptables

En la figura 3-42 se verifican los logs del firewall que hacen referencia al directorio /var/log/messages. Los resultados indican logs relacionados a solicitudes Dhcp.

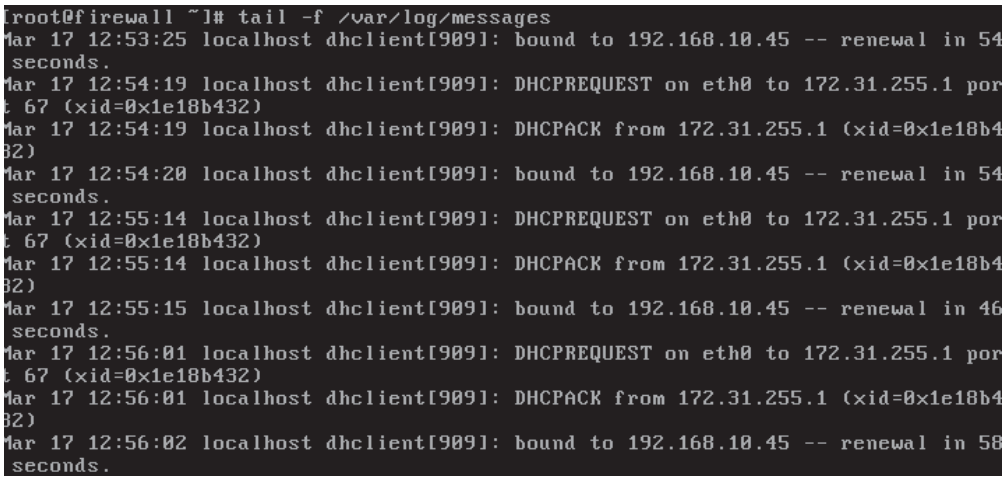


Figura 3-42 Logs del Firewall

2. Objetivo: Control de Contenido / Proxy Squid

En la figura 3-43 se verifican los logs del Servidor Proxy Squid que hacen referencia al directorio /var/log/squid/Access.log. Los resultados indican logs relacionados a la navegación web de la máquina de pruebas Backtrack 172.16.10.2.

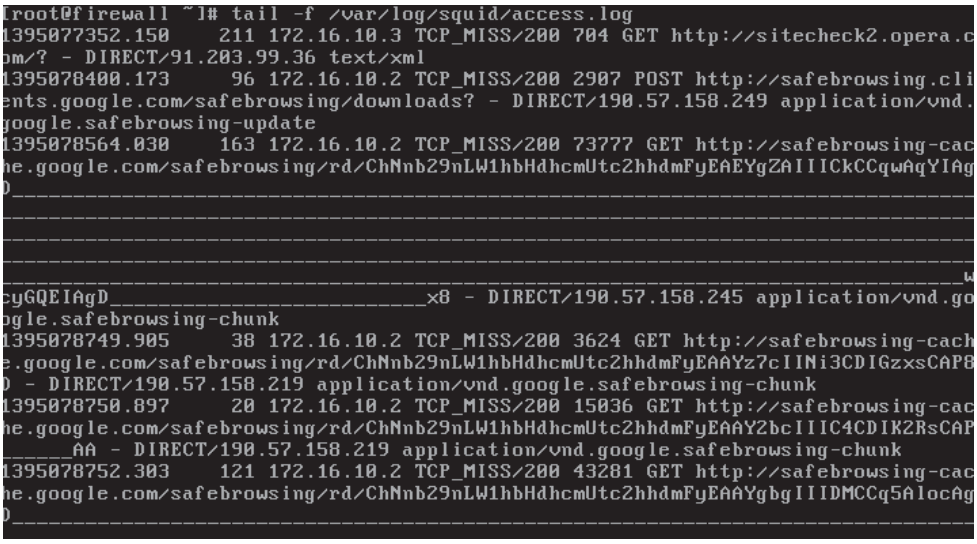


Figura 3-43 Logs del Squid Proxy Server

3. Objetivo: Servidor Web

En la figura 3-44 se verifican los logs del Servidor Web que hacen referencia al directorio /var/log/httpd/Access.log a nivel de usuarios. En la figura 3-45 en cambio los logs que hacen referencia al directorio /var/log/httpd/error.log con información de los procesos que corren bajo el servicio httpd.

Figura 3-44 Logs de acceso /var/log/httpd/Access.log

```
[root@dns1 lib]# tail -f /var/log/httpd/error_log
[Thu Mar 13 22:52:22 2014] [warn] mod_wsgi: Compiled for Python/2.6.2.
[Thu Mar 13 22:52:22 2014] [warn] mod_wsgi: Runtime using Python/2.6.6.
[Thu Mar 13 22:52:22 2014] [notice] Apache/2.2.15 (Unix) DAV/2 PHP/5.3.3 mod_ssl/2.2.15 OpenSSL/1.0.0-fips mod_wsgi/3.2 Python/2.6.6 mod_perl/2.0.4 Perl/v5.10.1 configu
red -- resuming normal operations
[Thu Mar 13 22:55:35 2014] [error] [client 172.16.9.1] File does not exist: /var/www/html/favicon.ico
[Mon Mar 17 10:44:00 2014] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[Mon Mar 17 10:44:00 2014] [notice] Digest: generating secret for digest authentication ...
[Mon Mar 17 10:44:00 2014] [notice] Digest: done
[Mon Mar 17 10:44:04 2014] [warn] mod_wsgi: Compiled for Python/2.6.2.
[Mon Mar 17 10:44:04 2014] [warn] mod_wsgi: Runtime using Python/2.6.6.
[Mon Mar 17 10:44:04 2014] [notice] Apache/2.2.15 (Unix) DAV/2 PHP/5.3.3 mod_ssl/2.2.15 OpenSSL/1.0.0-fips mod_wsgi/3.2 Python/2.6.6 mod_perl/2.0.4 Perl/v5.10.1 configu
red -- resuming normal operations
```

Figura 3-45 Logs de errores /var/log/httpd/error_log

4. Servicio Ftp

En la figura 3-46 se muestran los logs ubicados en el directorio /var/log/vsftpd.log para los procesos generales del servicio vsftpd y en el directorio /var/log/xferlog (Figura 3-47) a nivel de los archivos que han sido cargados y descargados en el servidor.

```
[root@dns1 log]# tail -f /var/log/vsftpd.log
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP command: Client "172.16.9.1", "CWD /directorioejemplo/ejemplo"
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP response: Client "172.16.9.1", "550 Failed to change directory."
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP command: Client "172.16.9.1", "PASV"
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP response: Client "172.16.9.1", "227 Entering Passive Mode (172,16,9,2,145,119)."
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP command: Client "172.16.9.1", "RETR /directorioejemplo/ejemplo"
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP response: Client "172.16.9.1", "150 Opening BINARY mode data connection for /directorioejemplo/ejemplo (0 bytes)."
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] OK DOWNLOAD: Client "172.16.9.1", "/directorioejemplo/ejemplo", 0.00Kbyte/sec
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP response: Client "172.16.9.1", "226 Transfer complete."
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP command: Client "172.16.9.1", "QUIT"
Mon Mar 17 13:08:03 2014 [pid 3426] [ftp] FTP response: Client "172.16.9.1", "221 Goodbye."
```

Figura 3-46 Logs del servidor ftp

```
[root@dns1 log]# tail -f /var/log/xferlog
Mon Mar 17 11:39:22 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
Mon Mar 17 11:53:10 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
Mon Mar 17 11:53:21 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
Mon Mar 17 12:19:04 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
Mon Mar 17 13:07:56 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
Mon Mar 17 13:08:03 2014 1 172.16.9.1 0 /directorioejemplo/ejemplo b _ o a chrome@example.com ftp 0 * c
```

Figura 3-47 Logs de archivos de carga y descarga del servidor ftp

5. Servidor DNS

El archivo de logs indicado en la Figura 3-48 hace referencia al directorio /var/log/messages para el servidor Dns.

```
[root@dns1 lib]# tail -f /var/log/messages
Mar 17 14:03:12 dns1 named[1499]: client 172.16.9.1#41553: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:13 dns1 named[1499]: client 172.16.9.1#56550: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:14 dns1 named[1499]: client 172.16.9.1#45651: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:15 dns1 named[1499]: client 172.16.9.1#41777: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:16 dns1 named[1499]: client 172.16.9.1#44236: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:17 dns1 named[1499]: client 172.16.9.1#48178: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:18 dns1 named[1499]: client 172.16.9.1#60015: query: 114.37.194.173.in-addr.arpa IN PTR + (172.16.9.2)
Mar 17 14:03:30 dns1 named[1499]: client 172.16.9.1#55099: query: \027[A IN A + (172.16.9.2)
Mar 17 14:03:39 dns1 named[1499]: client 172.16.9.1#54221: query: www.point.com.ec IN A + (172.16.9.2)
Mar 17 14:03:44 dns1 named[1499]: client 172.16.9.1#56517: query: www.google.com.ec IN A + (172.16.9.2)
```

Figura 3-48 Logs del servidor DNS

p) Verifique todas las respuestas de las solicitudes de paquetes UDP a los puertos 0-65535.

Se utiliza la herramienta Nmap con el argumento –Su –p para activar el modo de escaneo UDP en los puertos establecidos.

1. Objetivo: Servidor Web/Ftp/Dns

Se detalla el resultado del escaneo UDP al servidor Web/Ftp/Dns en la figura 3-49. El resultado indica que escucha conexiones en el puerto 53 Udp para el servicio Dns.

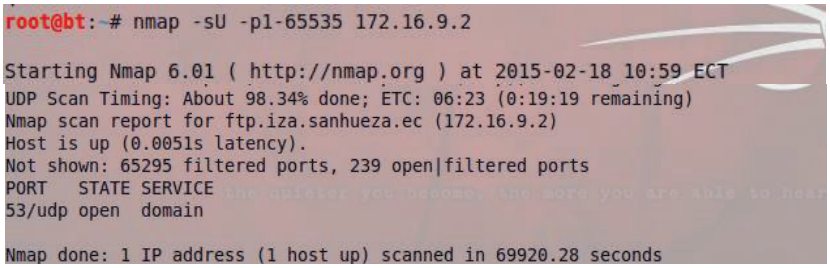


Figura 3-49 Escaneo a los puertos UDP 1-65535 del Servidor Web/Ftp/Dns

2. Objetivo: Servidor Firewall/Control de Contenido

Se detalla el resultado del escaneo UDP al servidor Firewall en la figura 3-50. El resultado indica que escucha conexiones en el puerto 11000 Udp, para el acceso al servicio Webmin del servidor.

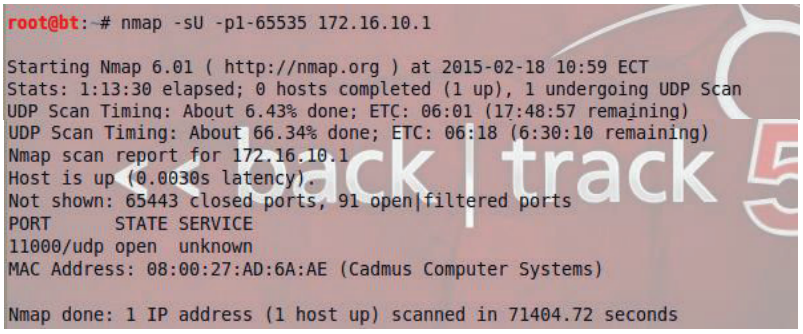


Figura 3-50 Escaneo a los puertos UDP 1-65535 del servidor Firewall

q) Verificar las respuestas de las solicitudes de paquetes UDP de los puertos fuente 0, 53, 139, y 161 a los puertos destino 0, 53, 69, 131, y 161.

Utilizamos la herramienta Nmap con los siguientes argumentos:

Opción -sU Establece l escaneo en modo UDP

Opción –source-port (Puerto Origen) Permite falsificar el puerto de origen con los indicados en la prueba.

Opción –p U: Establece el escaneo UDP a los puertos destinos indicados en la prueba.

1. Objetivo: Servidor Dns/Ftp/Web

En la figura 3-51 se muestra el resultado de aplicar la herramienta Nmap para escanear los puertos UDP falsificando el puerto origen y destino por los indicados en la prueba. El resultado indica que el servidor responde con el estado de algunos puertos como filtrados y abierto/filtrado. El estado filtrado indica que no se recibió ninguna respuesta tras varios intentos o que se recibió un error del tipo ICMP no alcanzable (tipo 3, códigos 1, 2, 3, 9, 10 o 13). El estado abierto/filtrado indica que el puerto podría estar abierto o es posible que exista un filtro que bloquee la comunicación. Para diferenciar entre estas 2 opciones realizamos una prueba utilizando el sondeo de versión con la opción -sV, indicado en la figura 3-52.

Figura 3-51 Escaneo UDP con puerto origen hacia el servidor Dns/Ftp/Web

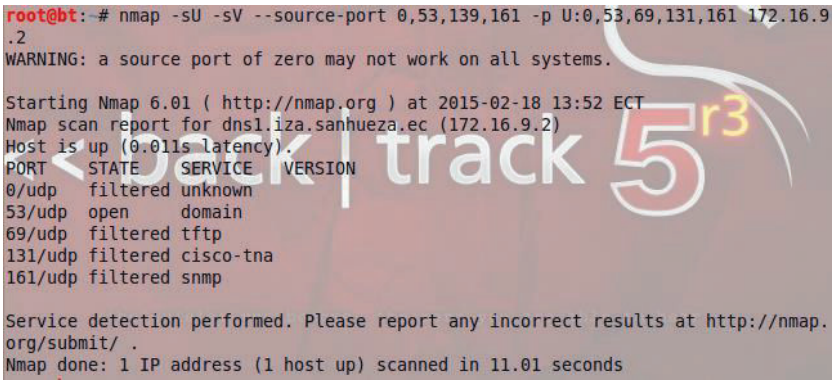


Figura 3-52 Escaneo UDP con puerto origen y sondeo de versión hacia el servidor Dns/Ftp/Web

2. Objetivo: Servidor Firewall/Control de Contenido

En la figura 3-53 se muestra el escaneo UDP con puerto origen falsificado y en la Figura 3-54 también el escaneo UDP con puerto origen falsificado y sondeo de versión hacia el Servidor Firewall/Control de Contenido.

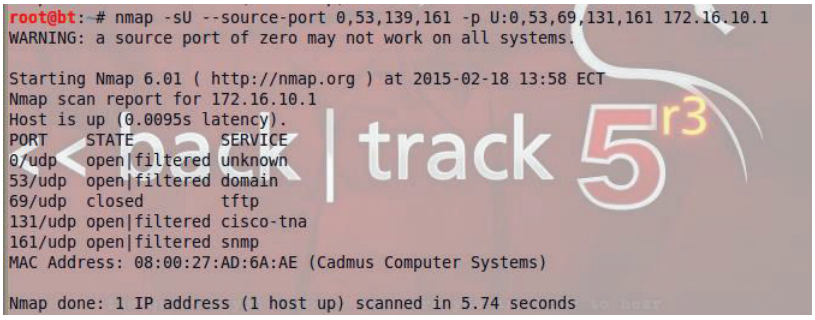


Figura 3-53 Escaneo UDP con puerto origen hacia el servidor Firewall/Control de Contenido

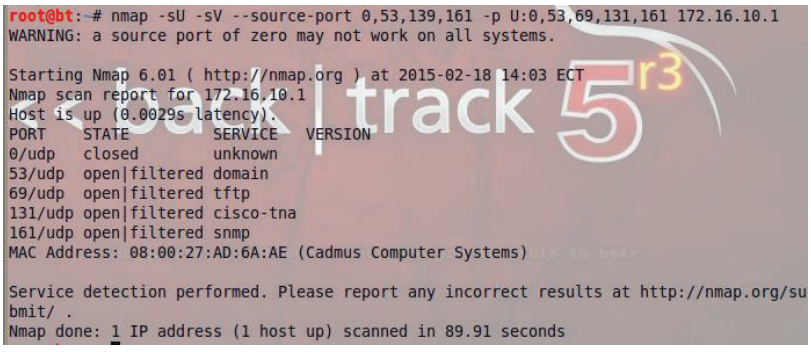


Figura 3-54 Escaneo UDP con puerto origen y sondeo de versión hacia el servidor Firewall/Control de Contenido

r) Verificar las respuestas a las solicitudes de paquetes UDP con sumas de comprobación (checksums) incorrectas a todos los puertos descubiertos en el literal q y para los puertos 0, 53, 69, 131 y 161.

Se utiliza la Herramienta Nmap con los argumentos apropiados para escanear las solicitudes UDP con checksum incorrecto. La sintaxis del comando es:

Comando: nmap –sU –p U:0,53,69,131,161 –badsum (Objetivo)

1. Objetivo: Servidor Dns/Ftp/Web

En la figura 3-55 se muestra el resultado del escaneo UDP con checksum incorrecto. Las solicitudes incluso pueden ni salir de la misma máquina de pruebas o si llegan a salir, el Receptor descarta el paquete, ya que asume que no ha recibido bien el paquete al realizar el cálculo del checksum y compararlo con el de la transmisión. Entonces muestra el estado de todos los puertos como filtrado, asumiendo que la conexión no se pudo establecer. Esto es transparente a que los puertos 53 Dns y 11000 Webmin realmente si están abiertos en los servidores Dns/Ftp/Web y Firewall/Control de Contenido respectivamente.

Figura 3-55 Escaneo UDP con checksum incorrecto al servidor DNS/Ftp/Web

2. Objetivo: Servidor Firewall/Control de Contenido

En la figura 3-56 se muestra el resultado del escaneo UDP con checksum incorrecto. Los resultados de igual forma indican el estado de todos los puertos como

abierto/filtrado asumiendo que la conexión no se pudo establecer o que existe un filtro que bloquea la comunicación.

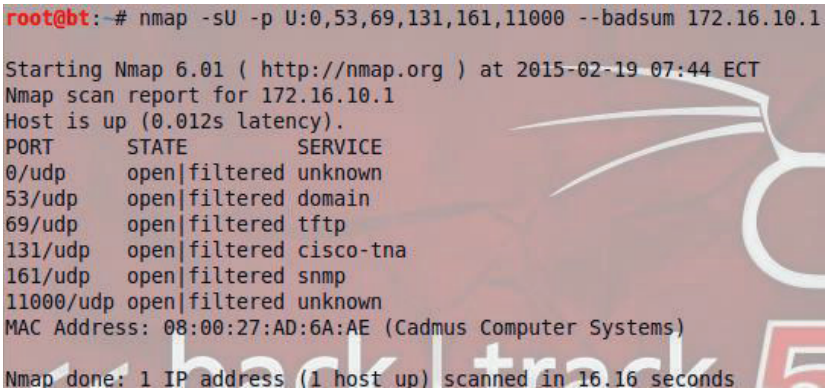


Figura 3-56 Escaneo UDP con checksum incorrecto hacia el servidor Firewall/Control de Contenido

s) Verificar las respuestas de solicitud de servicio a los puertos malware de acceso remoto UDP común y contemporáneo.

Se utiliza la Herramienta Nmap con la opción –Su –p U: para realizar el escaneo a los puertos malware de acceso remoto UDP, analizados en las tablas del anexo N. 6.

1. Objetivo: Servidor Dns/Ftp/Web

En las figuras 3-57, 3-58, y 3-59 se visualiza el resultado del escaneo nmap a los puertos UDP malware de acceso remoto del Servidor Dns/Ftp/Web, con referencia a los puertos registrados en la entidad IANA, Sistemas Unix Linux y Sistemas Windows respectivamente. Los puertos están en estado open/filtered o a su vez filtered, lo cual confirma que no existen conexiones abiertas hacia el servidor en esos puertos.

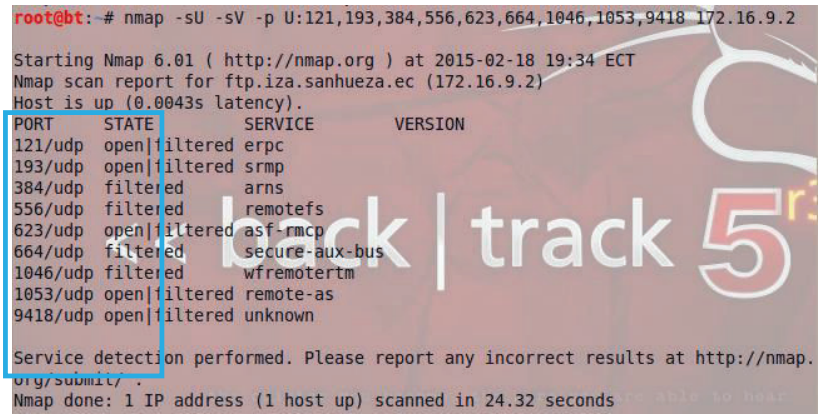


Figura 3-57 Escaneo nmap a los puertos UDP malware de acceso remoto con referencia IANA.

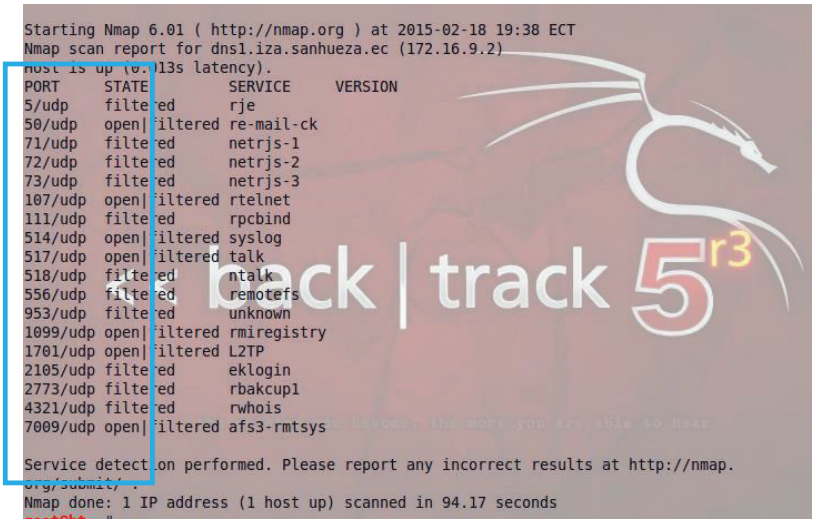


Figura 3-58 Escaneo nmap a los puertos UDP malware de acceso remoto con referencia Sistema Unix/Linux

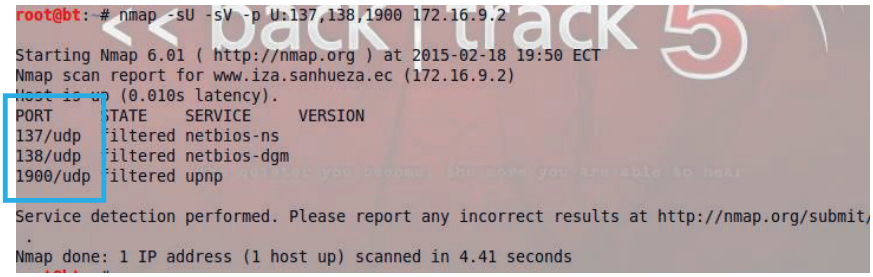


Figura 3-59 Escaneo nmap a los puertos UDP malware de acceso remoto con referencia Sistemas Windows

t) Verificar las respuestas a las solicitudes de paquetes TCP SYN a los puertos 0-65535.

Se utiliza la herramienta nmap –Ss –p- que realiza el escaneo TCP con la opción SYN activada a los puertos 1-65535.

1. Objetivo: Servidor Dns/Ftp/Web

En la Figura 3-60 se muestra el resultado del escaneo TCP al servidor Dns/Ftp/Web con la opción SYN activada a los puertos 1-65535. Se confirma los puertos TCP abiertos 21, 22, 53, 80 y 443.

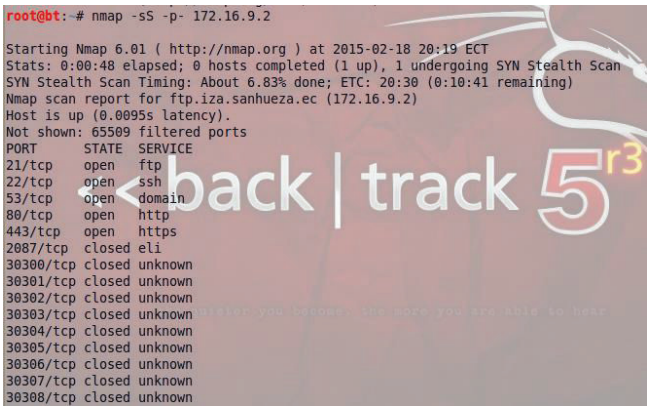


Figura 3-60 Escaneo TCP con la opción Syn activada a los puertos 1-65535

2. Objetivo: Servidor Firewall/Control de Contenido

En la Figura 3-61 se muestra el resultado del escaneo TCP al servidor Firewall/Control de contenido con la opción SYN activada a los puertos 1-65535. Se confirma los puertos TCP abiertos 22, 23, 80, 3128 y 11000.

Figura 3-61 Escaneo TCP con la opción Syn activada a los puertos 1-65535

u) Verificar las respuestas de las solicitudes de servicio TCP a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

Se uutiliza la herramienta Nmap –p (Número de puerto), para escanear las solicitudes TCP en los puertos indicados.

1. Objetivo: Servidor Dns/Ftp/Web

En la figura 3-62 se aprecia el resultado del escaneo TCP a los puertos indicados en la prueba para el servidor Dns/Ftp/Web.

Figura 3-62 Escaneo TCP a los puertos indicados en la prueba para el servidor Dns/Ftp/Web

2. Objetivo: Servidor Firewall/Control de Contenido

En la Figura 3-63 se aprecia el resultado del escaneo TCP a los puertos indicados en la prueba para el servidor Firewall/Control de Contenido.

Figura 3-63 Escaneo TCP a los puertos indicados en la prueba para el servidor Firewall/Control de Contenido

v) Verificar las respuestas de un TCP ACK con puerto origen de 80 a los puertos 3100-3150 y 10001-10050, 33500-33550 y 50 puertos aleatorios por encima de 35000.

Para cumplir esta sección se utilizará la **herramienta Nmap** con la opción -sA para activar el escaneo de tipo Tcp Ack. La sintaxis del comando es:

Comando: nmap –sA –source-port 80 –p (rango de puertos) Objetivo

1. Objetivos: Servidor Dns/Ftp/Web y Firewall/Control de Contenido

En la figura 3-64 se muestra el escaneo Tcp Ack en el rango 3100-3150, en la Figura 3-65 en el rango 10001-10050, en la Figura 3-66 en el rango 33500-33550 y en la Figura 3-67 en el rango 35100-35150.

Para el Servidor Dns/Ftp/Web, los resultados validan algún tipo de filtro dado a que los puertos no responden o envían mensajes de error ICMP en respuesta (tipo 3, código 1, 2, 3, 9, 10, o 13). Para el servidor Firewall/Control de Contenido se valida los puertos en estado de No Filtrado, lo que indica que pueden estar abiertos o cerrados y no están siendo filtrados.

Figura 3-64 Escaneo TCP con la opción ACK activada desde el puerto 80 al rango 3100-3150

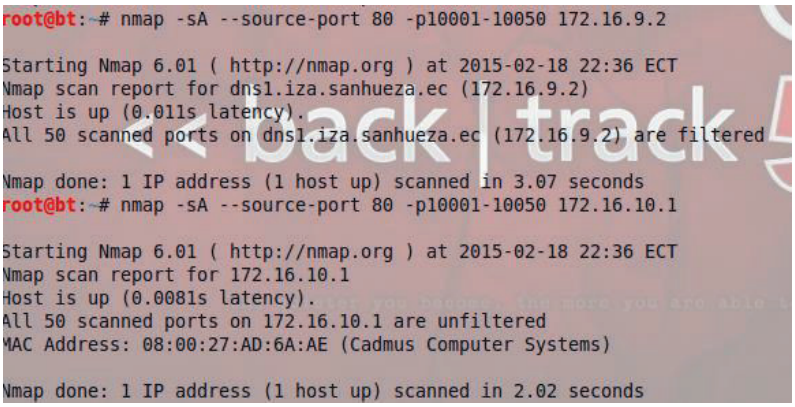


Figura 3-65 Escaneo TCP con la opción ACK activada desde el puerto 80 al rango 10001-10050

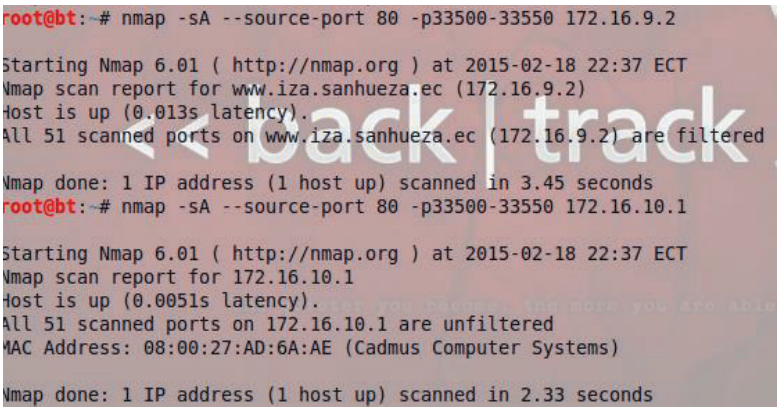


Figura 3-66 Escaneo TCP con la opción ACK activada desde el puerto 80 al rango 33500-33550

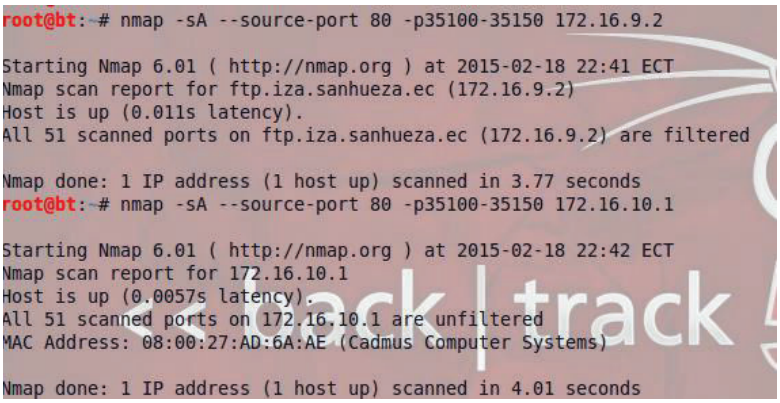


Figura 3-67 Escaneo TCP con la opción ACK activada desde el puerto 80 al rango 35100-35150

w) Verificar respuestas de los fragmentos TCP SYN a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

Se utilizará la Herramienta Nmap Opción –f que activa el escaneo con paquetes IP fragmentados. La sintaxis del comando es:

Comando: nmap -f –p (puertos destino) Objetivo

1. ObjetivoServidor Web/Ftp/Dns

En la Figura 3-68 se visualiza el resultado del escaneo TCP SYN con Fragmentación al Servidor Web/Ftp/Dns. Los resultados indican puertos abiertos y en estado filtrado.

Figura 3-68 Escaneo TCP SYN con Fragmentación al Servidor Web/Ftp/Dns

2. Objetivo: Servidor Firewall/Control de Contenido

En la figura 3-69 se aprecia el escaneo TCP SYN con Fragmentación al Servidor Firewall/Control de Contenido. Los resultados indican puertos abiertos y cerrados.

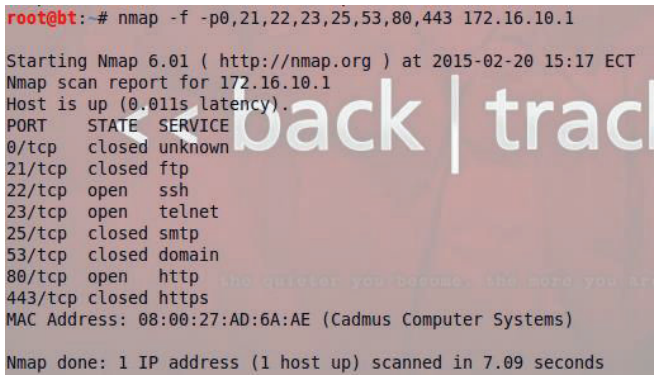


Figura 3-69 Escaneo TCP SYN con Fragmentación al Servidor Firewall/Control de Contenido

x) Verificar las respuestas de todas las combinaciones de las banderas TCP a los puertos 0, 21, 22, 23, 25, 53, 80 y 443.

Se utilizará la Herramienta Nmap con las siguientes opciones:

-Sn para el escaneo tipo Null con ninguna bandera activada

-sF para el escaneo tipo FIN con la bandera FIN activada

-Sx para el escaneo tipo XMAS con las banderas FIN, PSH, y URG activadas

--scanflag para el escaneo en las combinaciones de la Figura 2.31 literal x, detalladas anteriormente en el capítulo 2, sección 2.4.2.1.

1. Objetivo: Servidor Dns/Ftp/Dns

Sondeo –sN: Todas las banderas desactivadas.

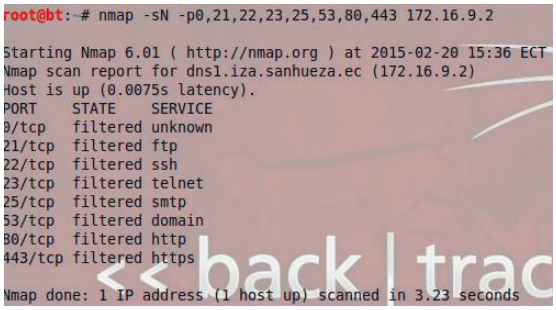


Figura 3-70 Sondeo –sN al servidor Dns/Ftp/Dns

Sondeo –sF solo bandera FIN activada

Figura 3-71 Sondeo –sF al servidor Dns/Ftp/Dns

Sondeo –sX solo las banderas FIN, PSH y URG activadas

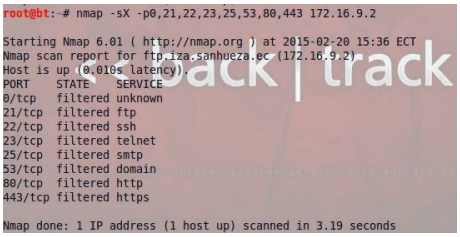


Figura 3-72 Sondeo –sX al servidor Dns/Ftp/Dns

Sondeo RST Activado

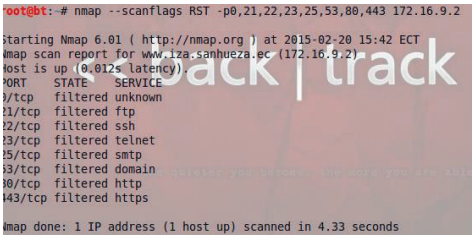


Figura 3-73 Sondeo –scanflag RST al servidor Dns/Ftp/Dns

Sondeo URG Activado

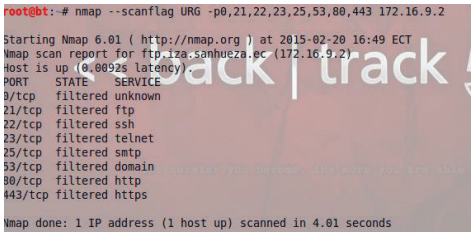


Figura 3-74 Sondeo –scanflag URG al servidor Dns/Ftp/Dns

Sondeo PUSH Activado

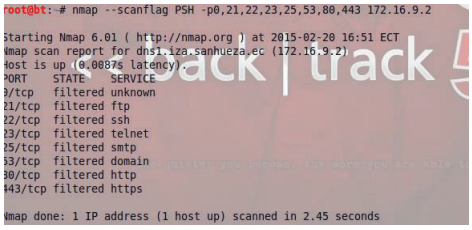


Figura 3-75 Sondeo –scanflag PSH al servidor Dns/Ftp/Dns

Sondeo FIN Y PUSH Activados

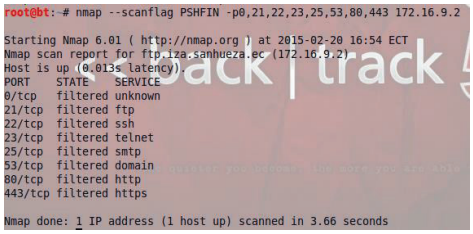


Figura 3-76 Sondeo –scanflag PSHFIN al servidor Dns/Ftp/Dns

Sondeo SYN y ACK activados

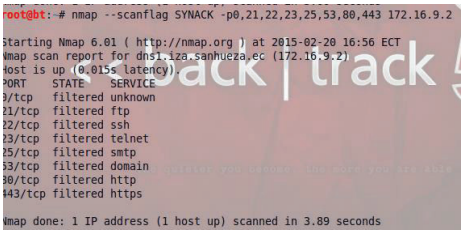


Figura 3-77 Sondeo –scanflag SYNACK al servidor Dns/Ftp/Dns

Sondeo SYN y RST activados

Figura 3-78 Sondeo –scanflag SYNRST al servidor Dns/Ftp/Dns

Sondeo ACK, FIN, PSH, URG Activados

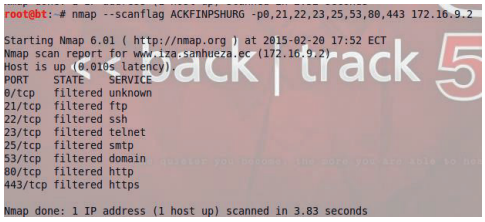


Figura 3-79 Sondeo –scanflag ACKFINPSHURG al servidor Dns/Ftp/Dns

Sondeo ACK y RST Activados

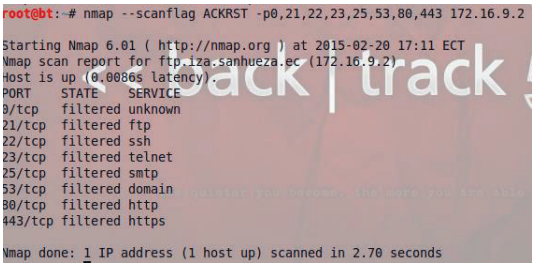


Figura 3-80 Sondeo –scanflag ACKRST al servidor Dns/Ftp/Dns

Sondeo SYN URG PSH FIN Activados

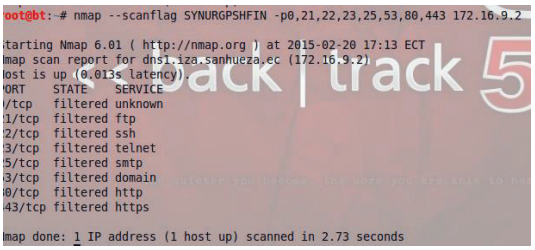


Figura 3-81 Sondeo –scanflag SYNURGPSHFIN al servidor Dns/Ftp/Dns

y) Verificar el uso de todos los objetivos con IPIDs secuenciales para enumerar los sistemas dentro de la red.

Se utilizará la herramienta Hping2 para analizar los IPIDs secuenciales de los objetivos y determinar los equipos que pueden ser “host Zombie” para técnicas de Idle Scan sobre host objetivos. También se usará la herramienta Nmap opción –Si –Pn –PO –p- para el escaneo de los puertos al host objetivo falsificando las peticiones como si se originaran desde el Host Zombie.

1. Prueba de hping2 para determinar “Host Zombie”

En la figura 3-82 se verifica el uso de la herramienta hping2 aplicado a la máquina de pruebas Windows 172.16.10.3. Los resultados indican parámetros como: longitud del paquete (46), el ttl (128), El bit No Fragmentar activado, la secuencia (seq), el tiempo

(Round Trip time), el tamaño de la ventana (win) y el campo ID correspondiente al identificador del datagrama IP. Esta prueba confirma que este equipo es ideal para ser host zombie ya que presenta el IPID con una secuencia positiva +1.

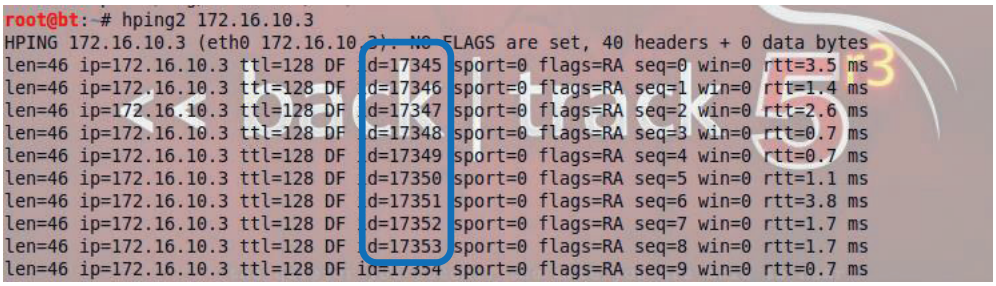


Figura 3-82 Hping2 a la máquina de pruebas Windows

2. Prueba de Nmap para escaneo Idle Scan al Servidor Dns/Ftp/Web

En la figura 3-83 se visualiza el uso de la herramienta Nmap para el escaneo Idle Scan sobre el objetivo Servidor Dns/Ftp/Web, identificando los puertos abiertos: 21, 22, 53, 80 y 443. La sintaxis del comando es:

Comando: nmap –Pn –PO –p- -Si (Host Zombie) (Host Objetivo)

Asegura que no se envíe ping inicial al host objetivo desde nuestra máquina de pruebas.	Define los puertos a escanear. Rango 1-65535	Uso de la técnica Idle Scan para hacer spoof de la IP que envía las peticiones como si fuera desde nuestra máquina

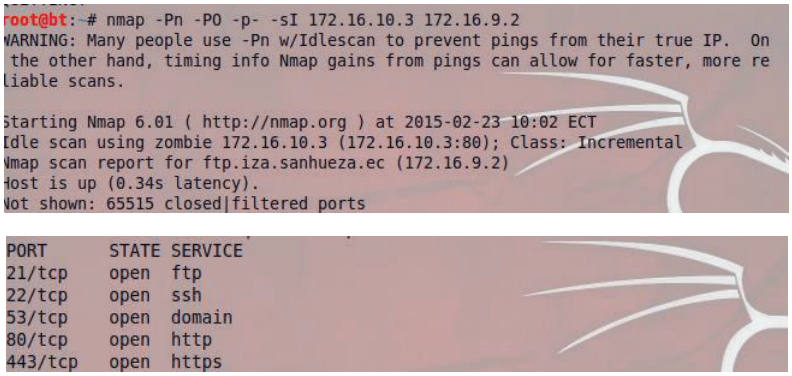


Figura 3-83 Herramienta Nmap para ejecutar Idle Scan al servidor Dns/Ftp/Web

Pruebas: Identificación

z) Identificar respuestas TTLs de los objetivos, Uptime de los sistemas, servicios, aplicaciones, fallos de aplicaciones y correlacionar esto con las respuestas de las herramientas de Fingerprinting para sistemas y servicios.

Para cumplir con esta sección se utilizarán las siguientes herramientas:

Ping: Para identificar respuestas TTLs de los objetivos

Nmap: con la opción –v –O para activar el escaneo Fingerprinting e indique información de Uptime, fallos de aplicación, etc.

WhatWeb: Para determinar información del sitio web local como versión, tipo de servidor, CMS, lenguaje, etc.

CMS-Explorer: Para verificar los plugins del sitio web local.

1. Verificación de TTL del Servidor Dns/Ftp/Web

En la Figura 3-84 se muestra la prueba de ping al servidor. El resultado confirma el valor de TTL a 63, por lo cual el TTL del sistema objetivo es 64 (Sistema Linux) y está a un salto de nuestra máquina de pruebas.

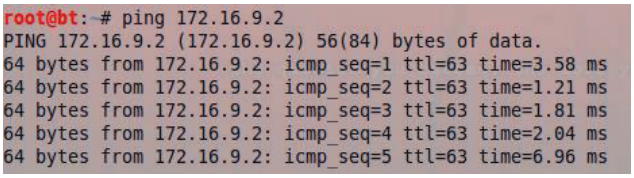


Figura 3-84 Ping al servidor Dns/Ftp/Web

2. Verificación de TTL del Servidor Firewall/Control de Contenido

En la figura 3-85 se muestra la prueba de ping al servidor. El resultado del valor de TTL es 64, por lo cual el TTL del sistema objetivo es 64 (Sistema Linux) y está conectado directo a la interfaz de la máquina de pruebas.

```
root@bt:~# ping 172.16.10.1
PING 172.16.10.1 (172.16.10.1) 56(84) bytes of data.
64 bytes from 172.16.10.1: icmp_seq=1 ttl=64 time=1.65 ms
64 bytes from 172.16.10.1: icmp_seq=2 ttl=64 time=1.37 ms
64 bytes from 172.16.10.1: icmp_seq=3 ttl=64 time=0.968 ms
64 bytes from 172.16.10.1: icmp_seq=4 ttl=64 time=1.83 ms
64 bytes from 172.16.10.1: icmp_seq=5 ttl=64 time=1.42 ms
64 bytes from 172.16.10.1: icmp_seq=6 ttl=64 time=1.85 ms
```

Figura 3-85 Ping al servidor Firewall/Control de contenido

3. Pruebas de Fingerprinting al servidor Dns/Ftp/Web

Utilizamos la herramienta nmap –v –O para detectar el sistema operativo del host objetivo con información del uptime. La sintaxis del comando es:

Comando: nmap –v –O (Objetivo)

En la figura 3-86 se visualiza el resultado de esta prueba. Se confirma que el sistema Operativo corresponde a Linux 3.2 con uptime de 0.170 días.

Figura 3-86 Nmap para Fingerprinting al servidor Dns/Ftp/Web

4. Pruebas de Fingerprinting al Servidor Firewall/Control de Contenido

En la figura 3-87 se visualiza el resultado de esta prueba, la cual confirma que el firewall está bloqueando el resultado del escaneo del sistema operativo, y el uptime corresponde a 0.230 días.

Figura 3-87 Nmap para Fingerprinting al servidor Firewall/Control de Contenido

5. Prueba de validación del Back-end[64] del Sitio Web local

En la figura 3-88 se verifican las pruebas para buscar el Back-end que contiene el logotipo de la plataforma. Resultó satisfactoria con: /administrator.

[64] El front-end es la parte del software que interactúa con el o los usuarios y el back-end es la parte que procesa la entrada desde el front-end. En diseño web hace referencia a la visualización del usuario navegante (front-end)), y del administrador del sitio con sus respectivos sistemas (back-end). Fuente: http://es.wikipedia.org/wiki/Front-end_y_back-end.

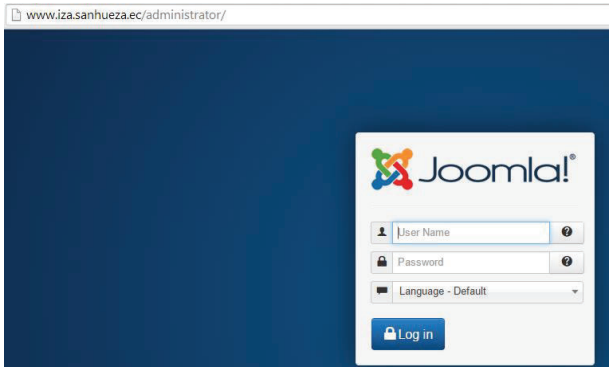
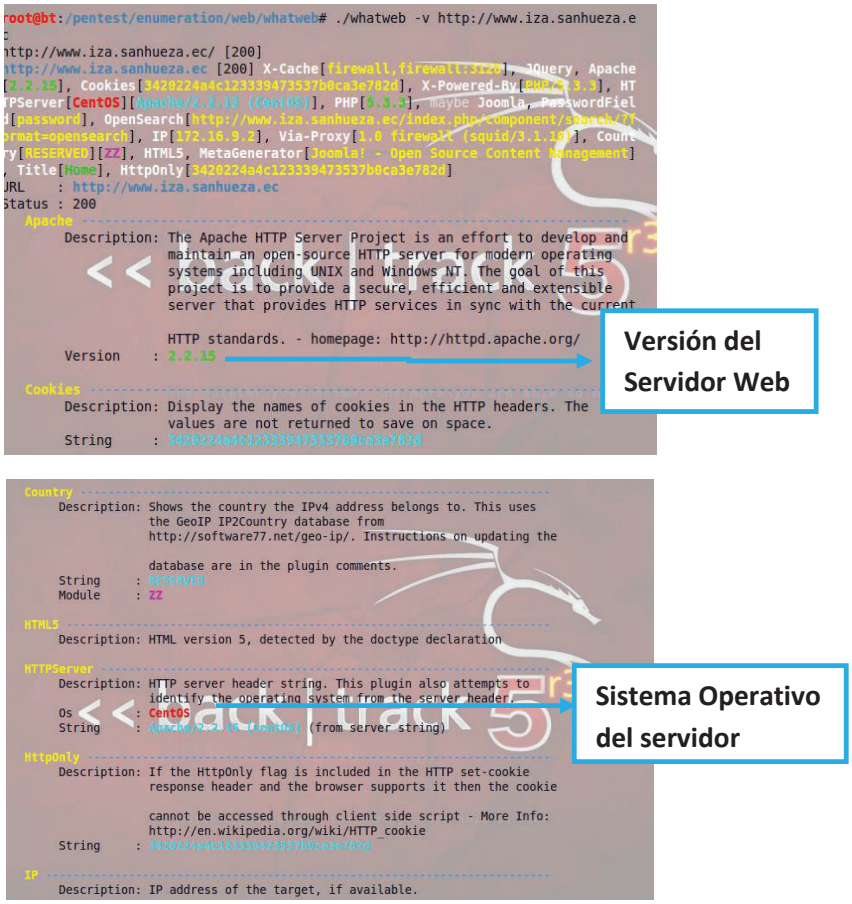


Figura 3-88 Prueba para identificar el Back-end del Sitio Web

6. Prueba de Fingerprinting a la página Web con WhatWeb

En la figura 3-89 utilizamos la herramienta WhatWeb para determinar información del sitio Web local, tal como versión, Sistema Operativo, Dirección IP, tipo de CMS, descripción de la librería Java Script, tipo y versión del lenguaje PHP, etc.



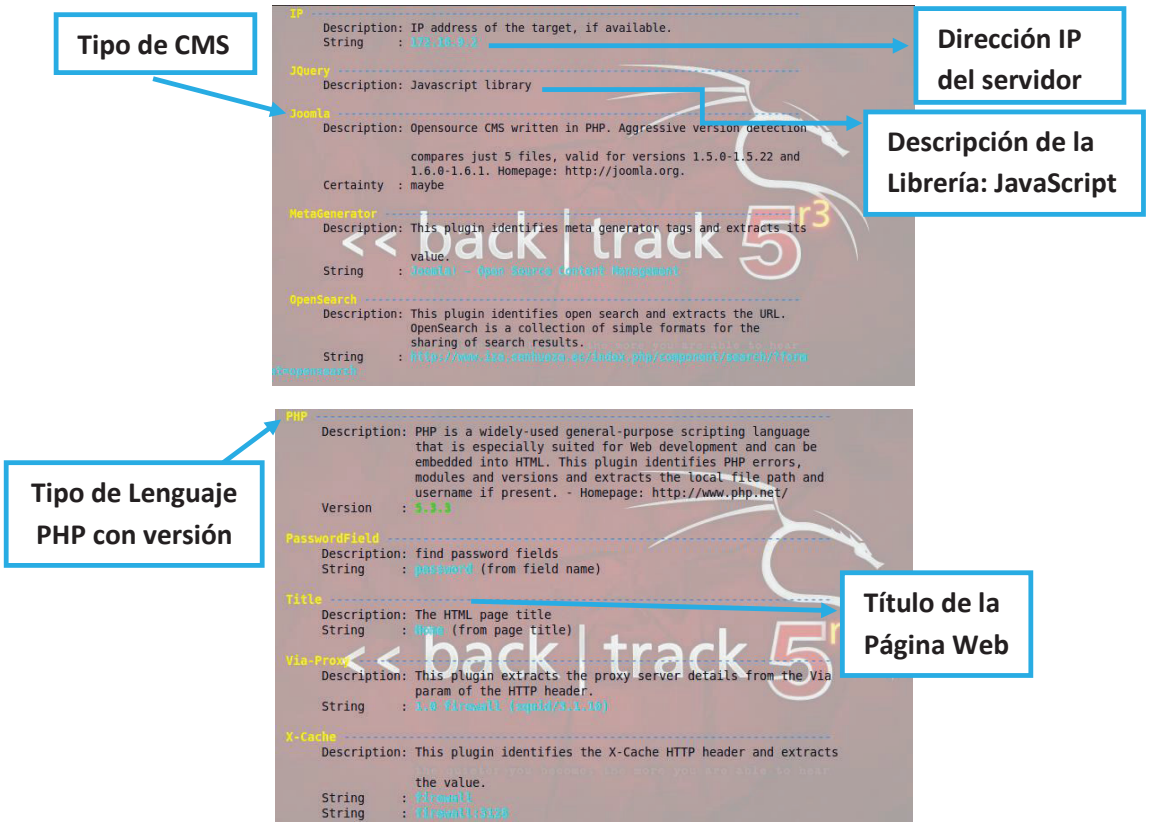


Figura 3-89 Fingerprinting con WhatWeb a la página Web Local

8. Pruebas de Fingerprinting a la página Web local con la herramienta cms-explorer

En la figura 3-90 utilizamos la herramienta cms-explorer para verificar los plugins instalados en el sitio web local. En base a esta información se puede buscar vulnerabilidades que comprometan la seguridad de un plugin instalado.

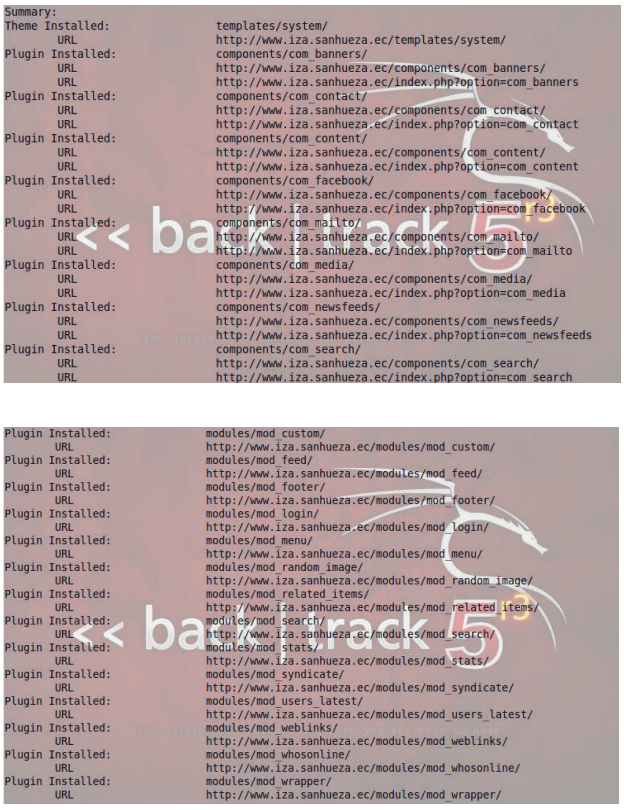


Figura 3-90 Fingerprinting con cms-explorer a la página web local

3.1.2.2 Módulo Verificación de Acceso

Dentro de este módulo se documenta las pruebas: Red, Servicios y Autenticación.

Pruebas: Red

a) Solicitar los servicios conocidos y comunes que utilizan UDP para las conexiones desde todas las direcciones

Se utiliza la herramienta Netcat y Dig para solicitar el servicio Dns a nivel local y obtener la respuesta desde la máquina de pruebas Backtrack. Los pasos se detallan a continuación:

1. Generar la petición DNS con la herramienta Netcat.

En la figura 3-91 solicitamos escuchar en un puerto arbitrario de la máquina de pruebas y el resultado se envía al archivo /tmp/dns-query.bin.

Comando: nc –ulp 5300 > /tmp/dns-query.bin

-u Activa el modo escaneo UDP
-l Activa el modo de escucha para conexiones entrantes
-p Define el puerto local -p

Resultado se envía al archivo /tmp/dns-query.bin

Figura 3-91 Escucha de peticiones dns en el puerto 5300

2. Generar la petición Dig al servidor DNS local

En la figura 3-92 solicitamos una petición Dig para consultar al servidor DNS local el sitio web "www.iza.sanhueza.ec" desde el puerto abierto para esta conexión (5300). La sintaxis del comando es:

Comando: dig@localhost –p 5300 www.iza.sanhueza.ec

Figura 3-92 Solicitud Dig para consultar al sitio web local

3. Comprobar la captura del archivo generado por consulta al sitio web local.

Una vez que Dig termina la prueba, se finaliza la escucha de la conexión del netcat inicial con Ctrl+C, y se comprueba que la captura del archivo se haya producido con el siguiente comando. El resultado se muestra en la Figura 3-93.

Comando: ls –l /tmp/dns-query.bin

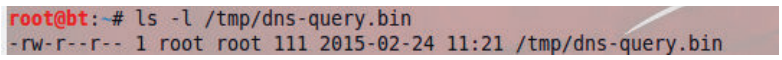


Figura 3-93 Verificación del archivo de petición al servidor Dns local

4. Capturar la respuesta del servidor DNS local

En la figura 3-94 se realiza la prueba, donde el resultado del contenido del archivo /tmp/dns-query.bin será la entrada standard para Netcat que se conecta al puerto UDP 53 del servidor Web local. Luego este resultado se envía al archivo /tmp/dns-reply.bin, validando que se establece la conexión hasta que automáticamente se cierra.

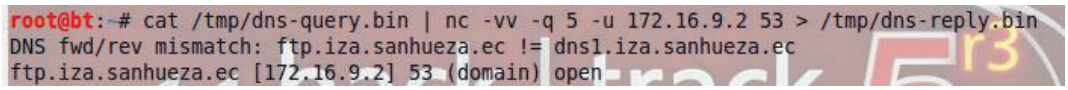


Figura 3-94 Envío de petición dns con netcat para conexión al servidor Dns.

5. Comprobar la captura del archivo generado en el paso 4

En la figura 3-95 se verifica el contenido del archivo /tmp/dns.reply.bin (como respuesta de la petición al servidor dns) con el siguiente **comando:**

Comando: ls –l /tmp/dns-reply.bin

Figura 3-95 Verificación del archivo de respuesta del servidor Dns local

Para ver el contenido de este archivo en formato hexadecimal y en ASCII utilizamos el comando hexdump[65] con la siguiente sintaxis (Figura 3-96).

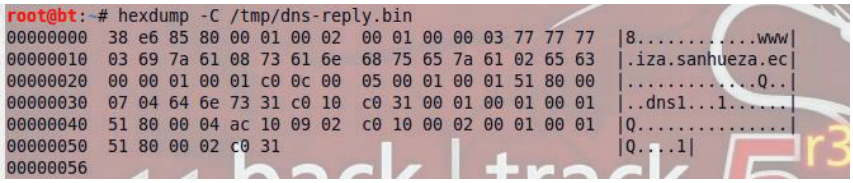


Figura 3-96 Visualización del archivo de respuesta /tmp/dns-reply.bin en formato Hex y Ascii

[65] Hexdump: Despliega una representación visible de todos los bites de un archivo. Fuente: http://francisconi.org/linux/comandos/hexdump.

b) Manipular los servicios de red y enrutamiento para el acceso a restricciones dentro del alcance.

Se utilizará la Herramienta Netcat opción –g para encaminar los paquetes cuando se establece la conexión al puerto 53 del servidor Dns/Ftp/Web. En la figura 3-97 se verifica esta prueba, cuyo resultado indica que se establece la conexión y se confirma que el firewall permite realizar encaminamiento de paquetes por origen desde cualquier parte de la red. La sintaxis del comando utilizado es:

Comando: nc –vv –u –g 172.16.10.3 –g 172.16.10.1 172.16.9.2 53

-vv Opción verbose para más detalle de la prueba.
-u Para activar modo Udp

-g Establece los gateway que atraviesa el paquete para llegar a su destino

Especifica el destino y el puerto

```
root@bt:~# nc -vv -u -g 172.16.10.3 -g 172.16.10.1 172.16.9.2 53
172.16.10.3: inverse host lookup failed: Unknown server error : Connection timed
 out
172.16.10.1: inverse host lookup failed: Unknown server error : Connection timed
 out
DNS fwd/rev mismatch: ftp.iza.sanhueza.ec != dns1.iza.sanhueza.ec
ftp.iza.sanhueza.ec [172.16.9.2] 53 (domain) open
```

Figura 3-97 Uso de netcat con opción de encaminamiento de origen hacia el Servidor Dns/Ftp/Web.

Pruebas: Servicios

c) Pedir todas las etiquetas de servicio (banners) de los puertos descubiertos, mediante interacciones con el servicio tanto con peticiones válidas y no válidas.

Se utiliza la **Herramienta Nmap** para conectarnos a los servicios del perímetro de red y verificar los banners que arrojan los sistemas tanto con peticiones válidas y no válidas. La sintaxis del comando se muestra a continuación.

Comando: nmap –vv (Servicio_Objetivo) (puerto_del_servicio)

1. Peticiones Válidas

En la figura 3-98 se muestra el resultado de aplicar Netcat para verificar el Banner del Servidor Web. Los resultados indican la siguiente información:

- Conexión establecida en el puerto 80
- Petición para obtener la cabecera del servidor web
- Respuesta del requerimiento código 200 OK
- Fecha
- Información del tipo y versión de servidor Web

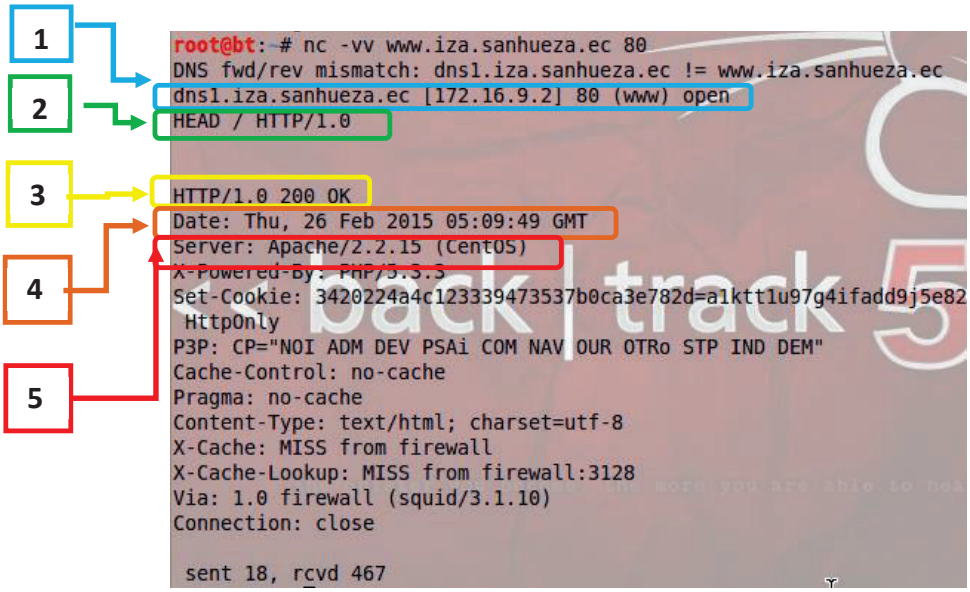


Figura 3-98 Uso de netcat para verificar Banner del Servidor Web

En la figura 3-99 se visualiza el resultado de aplicar Netcat para verificar el Banner del servicio FTP. El resultado indica solo el login de conexión al FTP.

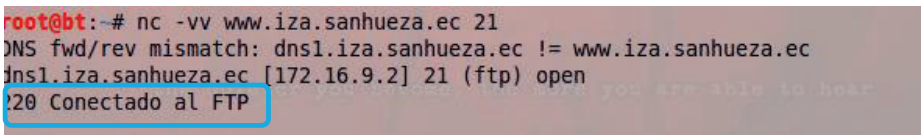


Figura 3-99 Uso de Netcat para verificar Banner del Servidor Ftp

En la figura 3-100 se verifica el uso de Netcat para indicar el Banner del servicio ssh corriendo en el servidor Dns/Ftp/Web. El resultado indica el tipo y versión de servicio SSH.

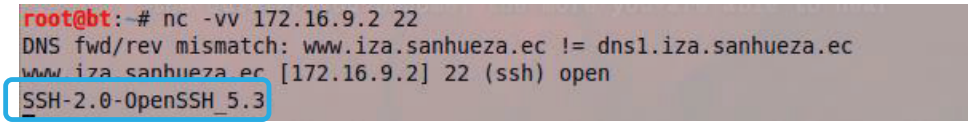


Figura 3-100 Uso del netcat para verificar Banner del servicio Ssh del Servidor Dns/Ftp/Web

En la figura 3-101 se muestra el uso de Netcat para verificar el banner del servicio SSH del servidor Firewall/Control de contenido. El resultado indica el tipo y versión de servicio SSH.

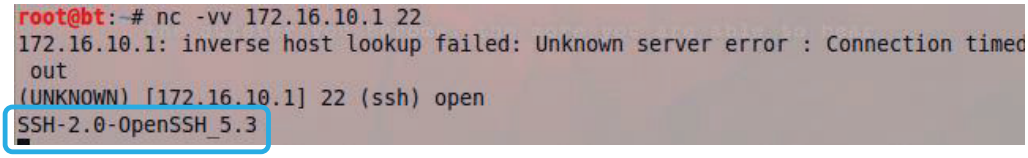


Figura 3-101 Uso del netcat para verificar Banner del servicio Ssh del Servidor Firewall/Control de Contenido

2. Peticiones no válidas

En la figura 3-102 se indica el uso de Netcat para verificar el Banner del servicio Squid Control del Servidor Proxy. El resultado indica que la conexión está iniciada en el puerto 3128, pero no da detalles del servicio como tal.

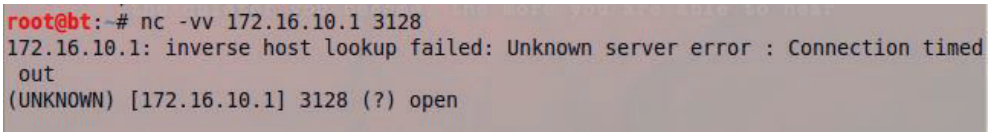


Figura 3-102 Uso del netcat para verificar Banner hacia el servicio de Squid Proxy

En la Figura 3-103 se muestra el uso de Netcat para verificar si existe algún Banner en la conexión hacia el servidor Firewall en el puerto dns 53. El resultado indica que la conexión se abre y no muestra detalles del servicio.

```
root@bt:~# nc -vv 172.16.10.1 53
172.16.10.1: inverse host lookup failed: Unknown server error : Connection timed
 out
(UNKNOWN) [172.16.10.1] 53 (domain) : Connection refused
 sent 0, rcvd 0
```

Figura 3-103 Uso de Netcat para verificar Banner en el servicio dns del Servidor Firewall/Control de Contenido.

En la figura 3-104 se muestra el uso de Netcat para probar si existe Banner en la conexión hacia el servidor Firewall en el puerto web 80. El resultado indica que el puerto está abierto, y que no está instalado un servidor Web por la respuesta “Código 403 Forbidden”. Además se constata la versión del Servidor Squid 3.1.10.

```
root@bt:~# nc -vv 172.16.10.1 80
172.16.10.1: inverse host lookup failed: Unknown server error : Connection timed
 out
(UNKNOWN) [172.16.10.1] 80 (www) open
HEAD / HTTP/1.0

HTTP/1.0 403 Forbidden
Server: squid/3.1.10
Mime-Version: 1.0
Date: Thu, 26 Feb 2015 03:13:23 GMT
Content-Type: text/html
Content-Length: 3208
X-Squid-Error: ERR_ACCESS_DENIED 0
Vary: Accept-Language
Content-Language: en
X-Cache: MISS from firewall
X-Cache-Lookup: NONE from firewall:3128
X-Cache: MISS from firewall
X-Cache-Lookup: MISS from firewall:3128
Via: 1.0 firewall (squid/3.1.10), 1.0 firewall (squid/3.1.10)
Connection: close

 sent 17, rcvd 454
```

Figura 3-104 Uso del netcat para verificar Banner hacia servicio web del Servidor Firewall.

d) Mapear cada puerto abierto a un demonio (servicio), aplicación (producto o código específico que usa el servicio) y protocolo (el medio para interactuar con ese servicio o aplicación.

En base a los resultados del literal anterior c, se asigna cada puerto abierto a su correspondiente servicio, aplicación y protocolo específico. En las tablas 3-1 y 3-2 se aprecian los resultados de esta asignación tanto para el Servidor Dns/Ftp/Web como para el Firewall/Control de Contenido.

Protocolo	Puerto	Servicio	Aplicación
TCP	21	FTP	No revela información
TCP	22	SSH	SSH-2.0-OpenSSH_5.3
TCP/UDP	53	DOMAIN	DNS
TCP	80	HTTP	Apache/2.2.15
TCP	443	HTTPS	No revela información

Tabla 3-1 Puertos abiertos para el Servidor Dns/Ftp/Web

Protocolo		Puerto	Servicio	Aplicación
TCP		22	SSH	SSH-2.0-OpenSSH_5.3
TCP		23	TELNET	TELNET
TCP		80	HTTP	No aplica
TCP		3128	SQUID-HTTP	Squid/3.1.10
TCP/UDP		11000	Desconocido	No revela información

Tabla 3-2 Puertos abiertos para el Servidor Firewall/Control de Contenido

e) Verificar el uptime del sistema con respecto a las últimas vulnerabilidades y liberación de parches.

Se uutiliza los **comandos uname –a y cat /etc/issue** para verificar el nivel de parcheado del sistema y la versión/tipo de distribución instalada respectivamente en los sistemas del perímetro de red. Luego analizamos las vulnerabilidades y liberación de parches respecto a la información obtenida.

1. Objetivo Servidor Firewall/Control de Contenido

En la figura 3-3 se muestra el resultado de aplicar estos comandos al servidor. Nos indica información del tipo, plataforma y versión del SO.

Característica	Resultado
Sistema Operativo	Linux 2.6.32-220.el6.i686
Versión	Centos 6.2

Tabla 3-3 Verificación del sistema operativo y versión del Servidor Firewall/Control de Contenido

2. Objetivo Servidor Dns/Ftp/Web

En la figura 3-4 se muestra el resultado de aplicar estos comandos al servidor. Nos indica información del tipo, plataforma y versión del SO.

Característica	Resultado
Sistema Operativo	Linux 2.6.32-220.el6.i686
Versión	Centos 6.3

Tabla 3-4 Verificación del sistema operativo y versión del Servidor Dns/Ftp/Web

3. Análisis de Vulnerabilidades y Liberación de Parches

En esta subsección detallamos las vulnerabilidades y liberación de parches de acuerdo a la información del tipo y versión del sistema operativo en los servidores del perímetro de red.

Vulnerabilidad Ghost código CVE-2015-0235:

Esta vulnerabilidad afecta a los sistemas Unix/Linux a la librería Glibc[66](GNU CLibrary). Permite que los atacantes adquieran el control total del sistema, sin conocer los accesos (usuarios y contraseñas) para el ingreso al mismo. Se logra mediante errores del tipo "desbordamiento de búfer" de la función básica de glibc llamada gethostbyname (), utilizada principalmente para la resolución DNS. En el Anexo N. 8 se detalla los últimos parches liberados para contrarrestar esta vulnerabilidad en los sistemas Centos 6.2. Para verificar si nuestro SO es vulnerable a Ghost, aplicamos las siguientes pruebas:

Wget https://gist.githubusercontent.com/pbruna/16c27df9ccb333f08f10/raw/30bcfd1da9cc250bf8c9bbf91eb4645780c77129/GHOST-test.sh
chmod +x GHOST-test.sh
./GHOST-test.sh

Tanto para el servidor Firewall/Control de Contenido como para el Dns/Ftp/Web se confirmó que son afectados por esta vulnerabilidad Ghost, obteniendo el siguiente resultado:

This system is vulnerable to CVE-2015-0235

<https://access.redhat.com/security/cve/CVE-2015-0235>

Please refer to 'https://access.redhat.com/articles/1332213' for more information

[66] Glibc es la biblioteca estándar de lenguaje C de GNU. Se distribuye bajo los términos de la licencia GNU LGPL, que proporciona y define las llamadas al sistema y otras funciones básicas. Fuente: http://es.wikipedia.org/wiki/Glibc.

Vulnerabilidad en el Kernel del Sistema Operativo

Los parámetros por defecto que maneja el sistema Kernel de Linux pueden derivar vulnerabilidades al sistema, tales como ataques de denegación de servicio, enrutamiento de paquetes por origen, etc. Con la utilidad sysctl de Linux es posible cambiar estos parámetros y mejorar la seguridad del sistema. En la tabla 3-5 se detalla las variables y valores recomendados en el sistema Kernel del servidor.

Variable y valor recomendado	Descripción
sysctl -w kernel.perf_event_paranoid=2	El valor por defecto en 1 permite a un usuario local aumentar sus privilegios en el sistema.
net.ipv4.icmp_echo_ignore_broadcasts = 1	Evitar ataques Smurf[67]
net.ipv4.tcp_syncookies = 1	Evitar ataques SynFlood[68]
kernel.exec-shield=1 kernel.randomize_va_space=1	Activar execshield[69] para reducir el riesgo de malware u otros ataques automatizados en sistemas Unix/Linux.
net.ipv4.conf.all.rp_filter=1	Habilitar protección ante ataques de Spoofing[70]
net.ipv4.conf.all.accept_source_route=0	Deshabilitar enrutamiento de paquetes por origen
net.ipv4.icmp_echo_ignore_broadcasts=1 net.ipv4.icmp_ignore_bogus_error_messages=1	Ignorar solicitudes de Broadcast

[67] El Ataque Smurf es una técnica de denegación de servicio que utiliza mensajes Spoofing de ping al broadcast para inundar (flood) el sistema objetivo. Fuente: http://es.wikipedia.org/wiki/Ataque_smurf.

[68] El Ataque SYN Flood es una técnica de denegación de servicio que envía un flujo de paquetes TCP/Syn con la dirección de origen falsificada. El servidor destino intentará establecer la conexión con un paquete TCP/SYN-ACK esperando el paquete de respuesta TCP/ACK (Parte del proceso de establecimiento de conexión TCP de 3 vías). Sin embargo, debido a que la dirección de origen es falsa o la dirección IP real no ha solicitado la conexión, nunca llega la respuesta, quedándose la conexión semi abierta y por tanto el servidor deja de contestar otras peticiones hasta poder completar la inicial. Fuente http://es.wikipedia.org/wiki/Ataque_de_denegaci%C3%B3n_de_servicio.

[69] Es un proyecto de Red Hat, Inc. A fin de reducir el riesgo de gusanos u otros ataques automatizados en sistemas Linux. Fuente: http://es.wikipedia.org/wiki/Exec_Shield.

[70] IP Spoofing consiste en sustituir la dirección IP origen de un paquete TCP/IP por otra dirección IP a la cual se desea suplantar. Esto se consigue gracias a programas destinados a ello y puede ser usado para cualquier protocolo TCP/IP como ICMP, UDP o TCP. Fuente: http://es.wikipedia.org/wiki/Spoofing.

net.ipv4.conf.all.log_martians = 1 net.ipv4.conf.default.log_martians = 1	Asegurarse que los paquetes spoofeados sean logueados
sysctl -w net.ipv4.conf.all.accept_redirects=0 sysctl -w net.ipv6.conf.all.accept_redirects=0 sysctl -w net.ipv4.conf.all.send_redirects=0 sysctl -w net.ipv6.conf.all.send_redirects=0	Deshabilitar redirección de enrutamiento ICMP71.
kernel.sysrq = 0	Deshabilitar la llave magic-sysrq72
net.ipv4.tcp_sack = 0	Deshabilitar Tcp Sack73
net.ipv4.tcp_timestamps = 0	Deshabilitar tcp_timestamps74
net.ipv4.tcp_syncookies = 1	Habilitar protección TCP Syn Cookie
net.ipv4.icmp_ignore_bogus_error_responses = 1	Habilitar mensajes de protección de errores

Tabla 3-5 Variables y valores recomendados al sistema Kernel de Linux

Vulnerabilidad Bourne again shell o Shellshock código CVE-2014-6271

Afecta al intérprete de comando Bash del Sistema Gnu/Linux, permitiendo la creación de variables de entorno arbitrarios antes de realizar la llamada al Shell mediante el Bash. Para verificar si nuestro sistema operativo es vulnerable a Shellshock aplicamos la siguiente prueba:

env x='() { :;}; echo vulnerable' bash -c "echo Si, hay que actualizar"

---

[71] La redirección de enrutamiento ICMP es un mecanismo para notificar a un host que una mejor ruta está disponible para un destino en particular y que actualice su tabla de enrutamiento. Esta habilitación de solicitudes ICMP Redirects podrían venir de un router no legítimo para desviar el tráfico a un.

[72] Llave Magic-sysrq es un método que mediante combinaciones de teclas entendidas por el kernel de Linux, permite al usuario realizar órdenes de bajo nivel sin tener en cuenta el estado del sistema. Fuente: http://es.knowledger.de/0625840/LlaveDeSysRqMagica.

[73] TcpSack es un algoritmo para detectar perdidas de paquetes. Fuente: http://www.linuxcolombia.com.co/?q=node/27 SACK.

[74] http://www.forensicswiki.org/wiki/TCP_timestamps. Mediante el análisis de las marcas de tiempo de TCP es posible calcular el tiempo Uptime del sistema y el tiempo de Arranque boot que pueden ayudar en la detección de Sistemas Operativos Fingerprinting.

Si la respuesta es el texto indicado “Si, hay que actualizar” entonces se confirma que esta vulnerabilidad existe en el sistema. En la figura 3-105 se valida esta prueba para el servidor Dns/Ftp/Web, confirmando que esta vulnerabilidad afecta al servidor. Para los sistemas afectados, la solución es actualizar el Bash del sistema operativo mediante el comando **yum –y update.**

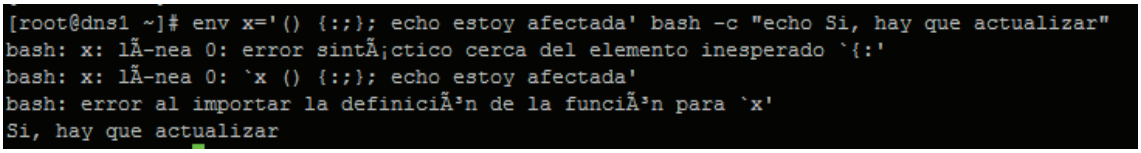


Figura 3-105 Verificación de Vulnerabilidad Bourne Again Shell al servidor Dns/Ftp/Web

f) Identificar los componentes del servicio en escucha.

Utilizamos la herramienta **Netcat** para realizar una conexión cliente-servidor, identificando el procedimiento y componentes tanto del servidor que escucha peticiones en un puerto, y el cliente que se conecta al servidor en dicho puerto. Posteriormente detallamos una prueba para identificar los componentes del Servidor Web en escucha de peticiones al puerto 80.

1. Habilitar el servidor para escuchar peticiones en un puerto arbitrario

En la figura 3-106 se muestra el resultado de habilitar el servidor (máquina de pruebas Backtrack) para escuchar peticiones en el puerto 1337. El resultado indica que la conexión empieza a escuchar peticiones en este puerto.

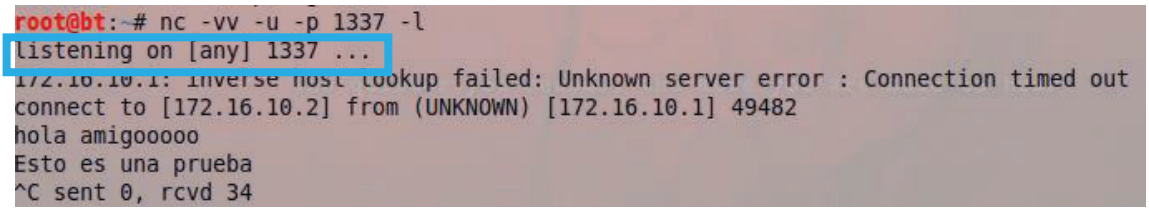


Figura 3-106 Uso de Netcat para habilitar el servidor con escucha de peticiones en el puerto 1337

2. Conexión de un cliente al servidor en el puerto específico

En la figura 3-107 se muestra el resultado de la conexión del cliente (Servidor Firewall) al servidor, y envío de un string de caracteres en la conexión establecida. Esta salida se visualiza en el servidor, tal y como lo muestra la figura 3.106.

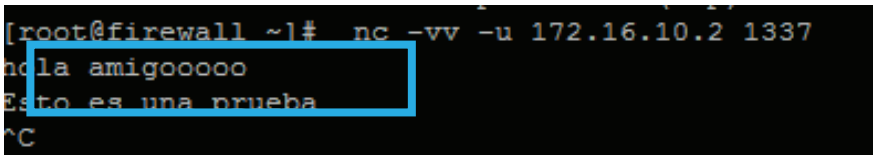


Figura 3-107 Uso de Netcat en la máquina cliente para conectarse al servidor en el puerto 1337

3. Prueba para identificar los componentes del Servicio Web en escucha de peticiones al puerto 80

Realizamos una prueba de conexión al servidor Web local "www.iza.sanhueza.ec" en el puerto 80, quien está escuchando constantemente peticiones en dicho puerto. En la figura 3-108 se muestra el uso de Netcat para conectarnos al servidor y verificamos su respuesta estableciendo la conexión.

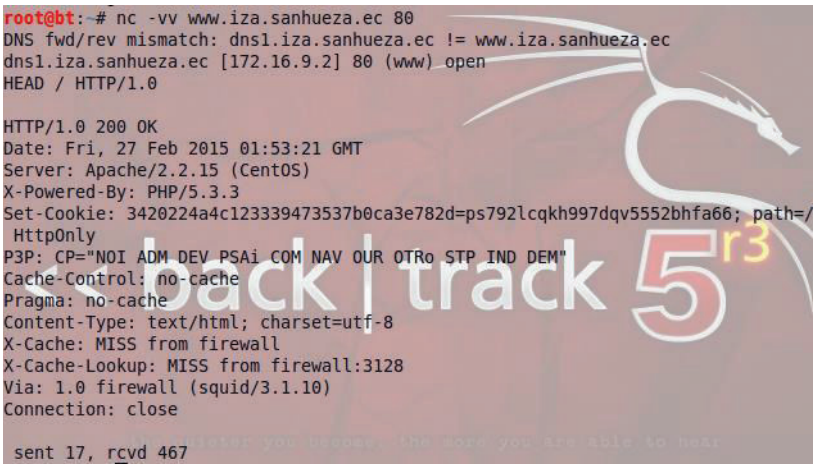


Figura 3-108 Uso de netcat para verificar conexión hacia el servicio Web local.

g) Verificar el uptime de los servicios comparado con las últimas vulnerabilidades y liberación de parches.

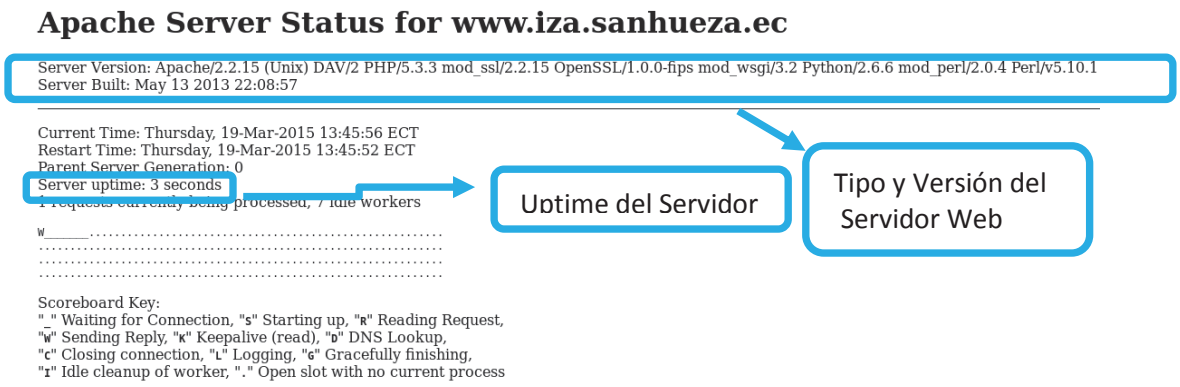
En base al valor del uptime, tipo o versión de los servicios verificamos las vulnerabilidades para cada uno de los servicios del perímetro de red.

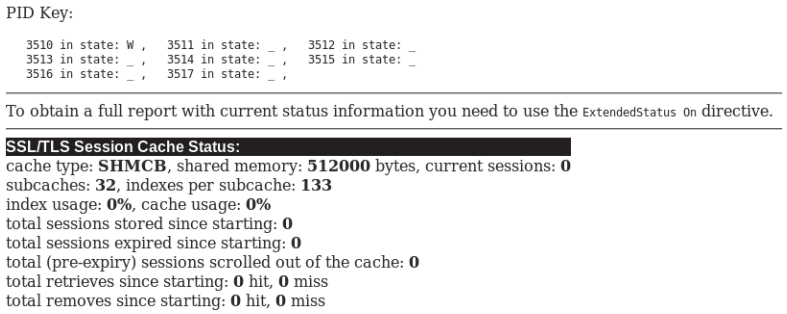
1. Objetivo Servidor Web

Utilizamos la **Herramienta Módulo server-status** del archivo apache del servidor Web para verificar el uptime, tipo y versión del servidor Web. En la figura 3-109 se muestra el resultado de activar este módulo en el archivo httpd.conf. Luego en la figura 3-110 se visualiza este módulo vía browser mediante el Url: http://www.iza.sanhueza.ec/server-status. El resultado indica el uptime del Servidor en 3 segundos con Versión Apache/2.2.15.

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 172.16.9.0/24
</Location>
```

Figura 3-109 Módulo Server-status del archivo "httpd.conf" del Servidor Web




PID Key:

3510 in state: W , 3511 in state: _ , 3512 in state: _
3513 in state: _ , 3514 in state: _ , 3515 in state: _
3516 in state: _ , 3517 in state: _ ,

To obtain a full report with current status information you need to use the ExtendedStatus On directive.

SSL/TLS Session Cache Status:
cache type: **SHMCB**, shared memory: **512000** bytes, current sessions: **0**
subcaches: **32**, indexes per subcache: **133**
index usage: **0%**, cache usage: **0%**
total sessions stored since starting: **0**
total sessions expired since starting: **0**
total (pre-expiry) sessions scrolled out of the cache: **0**
total retrieves since starting: **0** hit, **0** miss
total removes since starting: **0** hit, **0** miss

Figura 3-110 Uso del Módulo Server-status vía Browser

2. Análisis de vulnerabilidades en el Servidor Web

Vulnerabilidad Cross Site Scripting o Secuencia de Comandos en Sitios Cruzados XSS

Esta vulnerabilidad permite a un atacante inyectar código malicioso (JavaScript) debido a falta de control de filtrado en los campos de entrada de un Servidor Web. XSS puede ser usado para robo de información delicada, secuestro de sesiones de usuario[75], redirección del navegador a páginas Html malintencionadas y comprometer la integridad del sistema. Las medidas de precaución son: Limitar los caracteres de entrada, uso de filtros para limitar solo el ingreso de información requerida, eliminar etiquetas Html innecesarias, habilitar funciones para representar caracteres especiales insertados por el usuario, etc. Utilizamos la **Herramienta Nikto** para verificar si el Sitio Web local está afectado por esta vulnerabilidad. Para ejecutar la herramienta utilizamos el siguiente comando:

./nikto.pl –host http://www.iza.sanhueza.ec/mutillidae/

En la figura 3-111 se verifica el resultado de esta prueba, que refleja datos como versión del Servidor Apache, el banner que ha sido filtrado por el Servidor Squid

[75] Secuestro de sesiones de usuario también denominado SessionH ijacking hace referencia al robo de la identificación de un usuario para lograr acceso privilegiado a una aplicación Web, comprometiendo la confidencialidad e integridad de la información.

Proxy, Métodos Http permitidos, y el Osvdb[76] 27071 relacionado a la vulnerabilidad XSS activa en una imagen PHP del Sitio Web. Se concluye que directamente el sitio Web local no se ve afectado por esta vulnerabilidad.

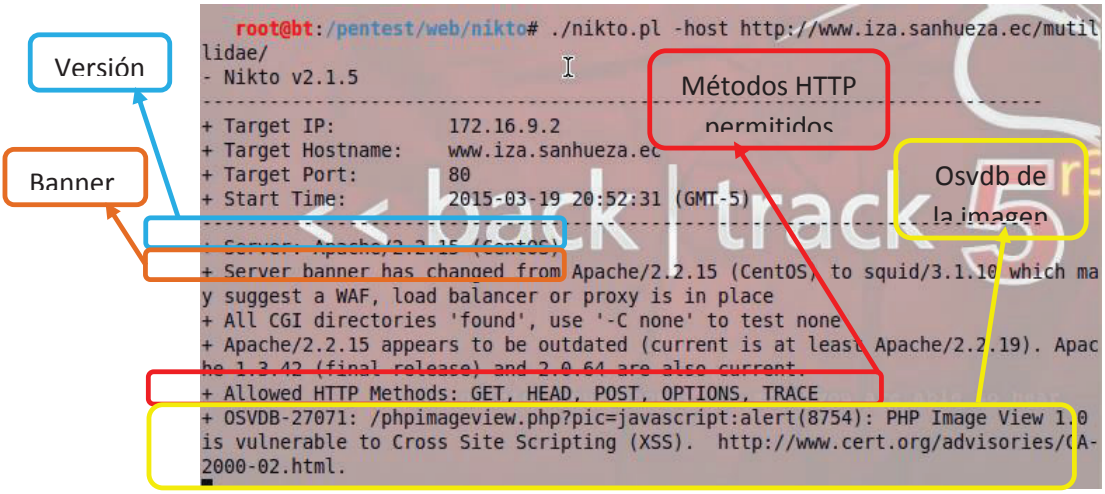


Figura 3-111 Herramienta Nikto aplicada al Servidor Web local

Vulnerabilidad XSSF

Se basa en el mismo principio de la vulnerabilidad XSS aplicada a un framework del sitio web. Permite administrar víctimas de ataques XSS genéricos y persiste una conexión con dichas victimas por medio de un "loop" en javascript, el cual se encarga de enviar peticiones reversas en intervalos definidos de tiempo a fin de ejecutar exploits contra la víctima[77]. Utilizamos la **Herramienta XSSF** para verificar víctimas de ataques XSS al servidor Web. En la figura 3-112 se muestra el resultado de este ataque con el comando "xssf_active_victims", el cual revela ninguna víctima asociada.

[76] OSVDB Es una base de datos independiente y de código abierto referente a las vulnerabilidades a nivel mundial. Fuente: http://en.wikipedia.org/wiki/Open_Source_Vulnerability_Database.

[77] Utilizando XSSF en Metasploit Framework. Fuente: http://thehackerway.com/2011/05/27/utilizando-xssf-en-metasploit-framework-explotando-xss/

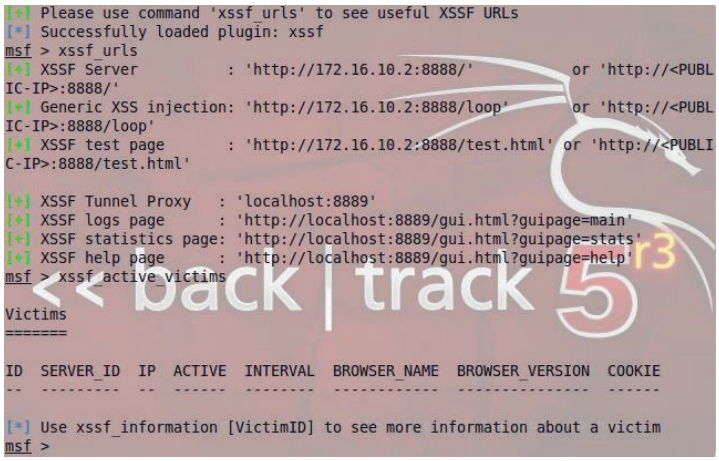


Figura 3-112 Herramienta Xssf de Metasploit.

Vulnerabilidad Inyección Sql

Es un método de infiltración de código Sql a nivel de la validación de las entradas en una consulta de base de datos. Esta vulnerabilidad se origina por el filtrado erróneo en las variables que genera código Sql, y una vez que toma el control puede leer los datos de la base de datos, modificarlos, ejecutar operaciones de administración, etc. Utilizamos la **Herramienta SqlMap** para verificar si el Servidor Web local está afectado por esta vulnerabilidad. En la figura 3-113 se muestra el resultado de esta prueba, que confirma que el sitio web no está afectado por esta vulnerabilidad.

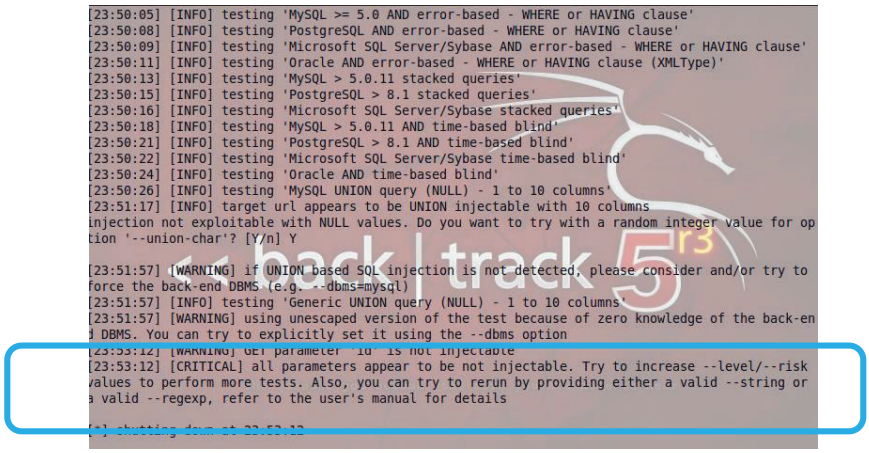


Figura 3-113 Herramienta Sqlmap aplicada al Servidor Web Local

Vulnerabilidad "Perdida de Autenticación y gestión de sesiones"

Las funciones de la aplicación Web relacionadas a la autenticación y gestión de sesiones son comúnmente implementadas con potentes agujeros de seguridad, comprometiendo la integridad del sitio, tal como la obtención ilícita de contraseñas, llaves, token de sesiones e incluso la implementación de fallas para asumir la identidad de otros usuarios.[78] Para validar si esta vulnerabilidad afecta el sitio Web local realizamos las siguientes pruebas:

a) Verificación del sistema de autenticación del usuario Administrador con acceso correcto e incorrecto.

Acceso Correcto

En la figura 3-114 se valida el acceso correcto del usuario administrador al Sitio Web local mediante la Url: http://www.iza.sanhueza.ec/administrator. En la figura 3-115 se muestra la captura Wireshark del paquete Http tipo POST[79] como resultado del envío de información de autenticación al servidor. Las credenciales viajan en texto plano

[78] A3 Perdida de Autenticación y Gestión de Sesiones. Fuente: https://www.owasp.org/images/f/ff/3.OWASP_Day_Costa_Rica_Mario.pdf.

[79] Indica al servidor que se prepare para recibir información del cliente. Se usa para enviar información desde formularios. Fuente: http://trevinca.ei.uvigo.es/~txapi/espanol/proyecto/superior/memoria/node46.html.

validando que no se utiliza mecanismo de cifrado en este proceso. En la figura 3-116 se muestra la captura Wireshark del paquete Http tipo GET como resultado del envío de la cookie almacenada. El timeout del cache es 3600 segundos que indica el tiempo de almacenamiento de la cookie en el navegador.

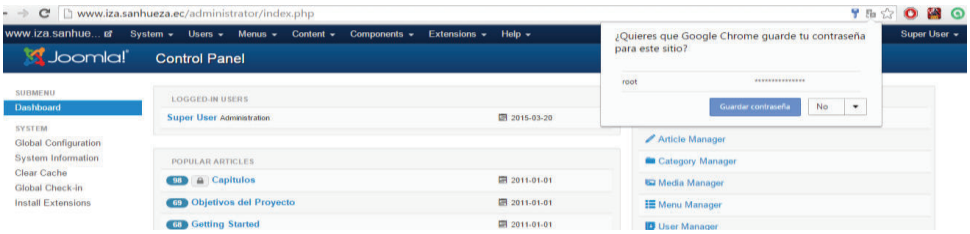


Figura 3-114 Acceso correcto del usuario administrador del Sitio Web

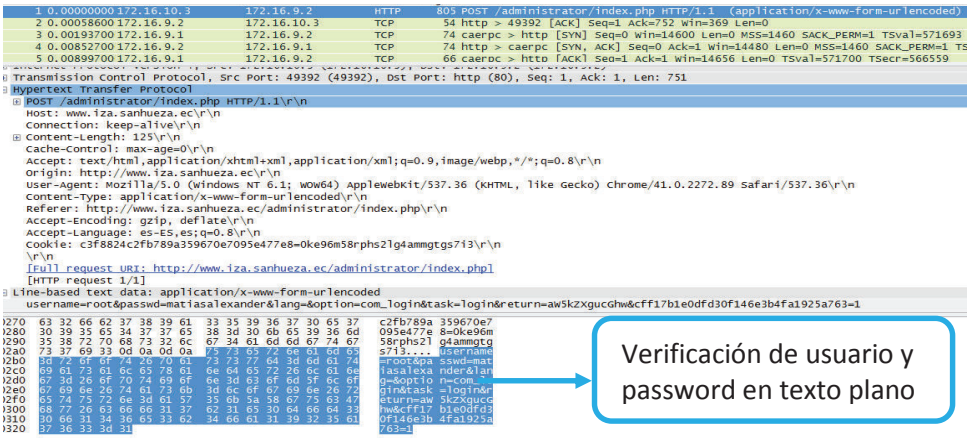


Figura 3-115 Captura Wireshark del paquete Http Post para enviar credenciales del administrador al Servidor Web

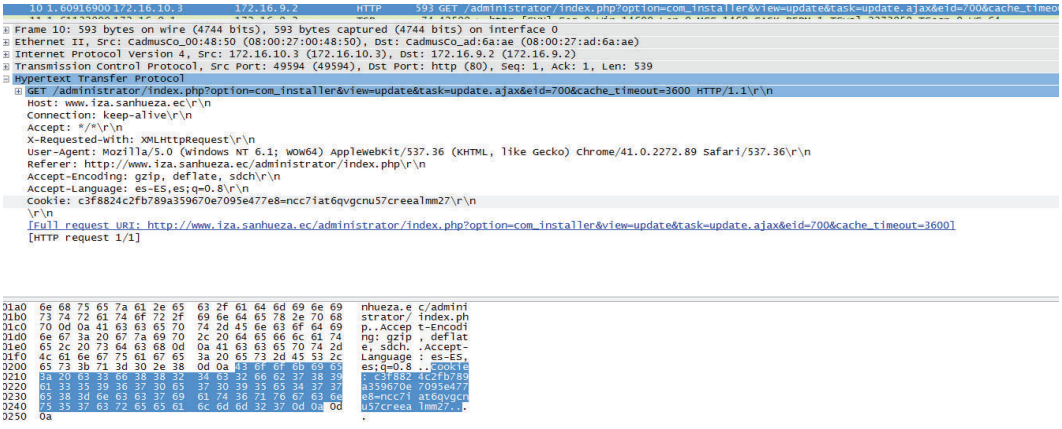


Figura 3-116 Validación del Cookie mediante la petición Http Get al Servidor Web local

Acceso Incorrecto

En la figura 3-117 se verifica el proceso de acceso incorrecto a la administración del Sitio Web local. El resultado es un Warning de usuario y password incorrecto.

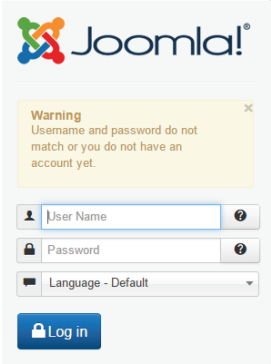


Figura 3-117 Validación de Acceso Incorrecto al sitio Web

b) Verificación de la caducidad de la sesión tras inactividad.

En la figura 3-118 realizamos un prueba de acceso al sitio Web, tras un tiempo de inactividad. El resultado es un Warning que valida que el sitio Web cuenta con mecanismos de caducidad de la sesión tras inactividad.

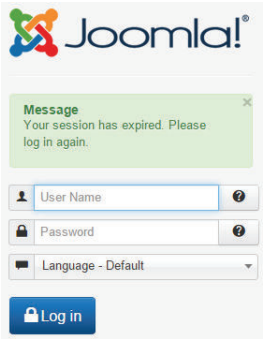


Figura 3-118 Validación de caducidad de la sesión tras inactividad

3. Objetivo Servidor Ftp

Se utiliza la **Herramienta Nmap** con la opción –Sv para identificar el tipo y versión del servicio Ftp. En la figura 3-119 se muestra esta prueba, que indica el nombre del servicio “Vsftpd” y la versión 2.0.8 correspondiente.

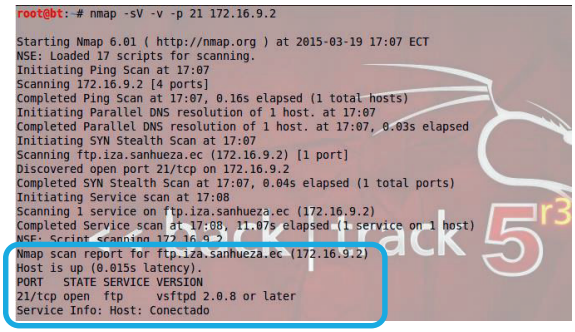


Figura 3-119 Uso de nmap para verificar uptime y versión del servidor Ftp

4. Análisis de Vulnerabilidades en el servidor Ftp

Vulnerabilidad "Backdoor Vsftpd 2.3.4"

Esta vulnerabilidad permite crear una puerta trasera por la cual se puede acceder al servidor FTP con permisos de administrador. Se utilizará la herramienta Metasploit Msf para verificar si el servidor Ftp local es vulnerable al Backdoor Vsftpd_234. En la figura 3-120 se verifica que el Servidor Ftp local no es afectado por este Backdoor ya que solicita el password de acceso. Los pasos se detallan a continuación:

a) Verificar la existencia del Backdoor con el comando "search vsftpd"

b) Aplicar los comandos "use exploit/unix/ftp/vsftpd_234_backdoor", "set RHOST 172.16.9.2" y "exploit" desde la máquina de pruebas Backtrack.

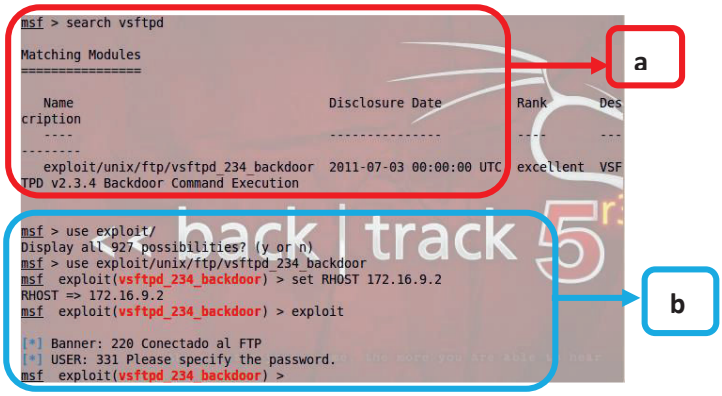


Figura 3-120 Verificación de vulnerabilidad "Backdoor_Vsftpd_234" en el Servidor Ftp

5. Objetivo Servidor Dns

Se utiliza el Comando “service named status” para validar la versión del Servidor Dns, tal y como se visualiza en la figura 3-121.

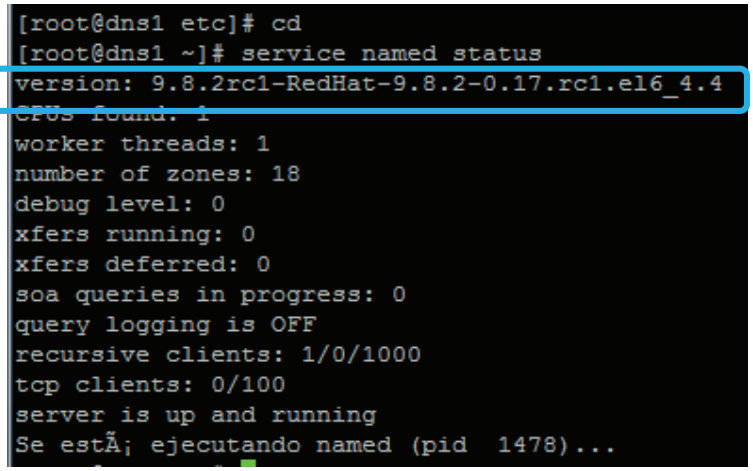


Figura 3-121 Verificación de versión del Servidor Dns

6. Análisis de Vulnerabilidades en el Servidor Dns

Vulnerabilidad de entropía de insuficiencia de zócalos (sockets) de DNS

Permite falsificar el tráfico Dns mediante técnicas de contaminación de Cache[80] en servidores Dns que tienen activa la opción de “Allow-recursion” para permitir la resolución recursiva a todo el mundo. Esta técnica puede re direccionar las peticiones de los clientes hacia direcciones falsas. La medida de protección es limitar la resolución recursiva solo a redes Lan internas y la Dirección Ip del propio servidor. En la figura 3-122 se verifican las secciones “allow query/recursión” del archivo “named.conf” para constatar si el servidor Dns está expuesto a esta vulnerabilidad. Se permiten las consultas y resolución recursiva solo a redes Lan Internas y la Ip del propio servidor, confirmando que no está expuesto a esta vulnerabilidad.

[80] Ataque de envenenamiento de caché Dns: El atacante explota una vulnerabilidad en el software de DNS que permite aceptar información incorrecta, cuando no valida correctamente las respuestas DNS para asegurarse de que provienen de una fuente autoritativa. El servidor puede terminar almacenando localmente información incorrecta y enviándola a los usuarios para que hagan la misma petición. Fuente http://es.wikipedia.org/wiki/DNS_cache_poisoning.

```
options {
listen-on port 53 { 172.16.9.2; };
directory       "/var/named/";
dump-file       "/var/named/data/cache_dump.db";
statistics-file "/var/named/data/named_stats.txt";
allow-query     { 172.16.9.0/24; 172.16.10.0/24; localhost; };
allow-recursion {
127.0.0.1;
172.16.9.0/24;
172.16.10.0/24;
};
```

Figura 3-122 Verificación de Vulnerabilidad "Entropía de insuficiencia de zócalos (sockets) de DNS"

Vulnerabilidad de transferencias de zona libre

Permite realizar transferencias de zona Dns desde cualquier parte, dando lugar a accesos malintencionados, tales como: Contaminación del Cache del Servidor, redirección de peticiones Dns a sitios falsos, consultas de subdominios del servidor, etc. Para evitar esta vulnerabilidad se permite la transferencia únicamente al propio servidor Dns. Esta vulnerabilidad también se la refiere con otro tipo de transferencias entre un servidor Dns primario y secundario, conocido como AXFR.[81] Verificamos la sección "allow-transfer" en los archivos de configuración de la zona directa e inversa para constatar si el servidor Dns local está afectado por esta vulnerabilidad. En la figura 3-123 se muestra esta prueba, que indica la sección "allow-transfer" habilitado solo para las direcciones Ip del propio servidor Dns 172.16.9.2 y localhost.

```
zone "iza.sanhueza.ec" IN {
        type master;
        file "iza.sanhueza.ec.zone";
        allow-transfer { 172.16.9.2; localhost; };
        allow-query { any; };
};
```

```
zone "9.16.172.in-addr.arpa" IN {
        type master;
        file "9.16.172.in-addr.arpa.zone";
        allow-query { any; };
};
```

Figura 3-123 Verificación de Vulnerabilidad de transferencia de zona libre en el servidor Dns

[81] Las siglas AXFR hace referencia a la transferencia por zonas de un DNS primario a un DNS secundario o de un DNS primario a un server maestro y de un server maestro a un DNS secundario, si llegara a existir algún problema de configuración o actualización del software de cualquiera de estos servidores se podrían explotar una serie de vulnerabilidades como por ejemplo un DoS y la integridad y confidencialidad de la base de datos del DNS primario se verían comprometidas, se estima que alrededor de un 60% de los servidores DNS en internet son vulnerables. http://blog.zerial.org/seguridad/axfr-una-vulnerabilidad-que-pasa-desapercibida/.

h) Verificar los hosting virtuales por HTTP y HTTPS

Verificamos en el archivo de configuración apache “httpd.conf” que la opción de Virtual Hosts esta desactivada (Figura 3-124).

```
### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
```

Figura 3-124 Verificación de Hosting Virtual en el archivo “httpd.conf”

Pruebas: Autenticación

i) Enumerar los accesos que requieren autenticación y documentar todos los privilegios descubiertos, que pueden ser usados para proveer acceso.

Las pruebas para verificar los accesos que requieren autenticación se detallan a continuación. Posteriormente se documentará todos los privilegios obtenidos.

1. Acceso a la página de administración del sitio Web local

Con la herramienta Wireshark verificamos el acceso del usuario administrador al sitio web local. En la figura 3-125 se muestra esta prueba que confirma que las credenciales no son cifradas y se transmiten en texto plano.

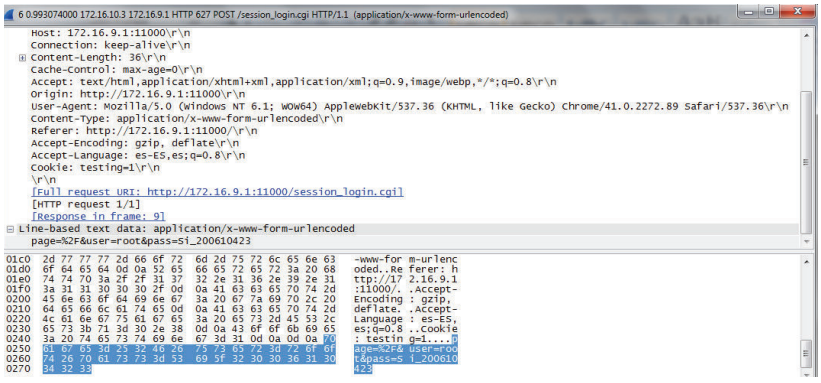


Figura 3-125 Captura Wireshark para acceso a la interfaz de administración del Servidor Web local

2. Acceso a la interfaz gráfica de administración “Webmin” del Servidor Firewall/Control de Contenido.

En la figura 3-126 se muestra la captura Wireshark cuando se accede a la interfaz Webmin del Servidor Firewall/Control de Contenido, la cual también confirma que las credenciales no son cifradas y se transmiten en texto plano.

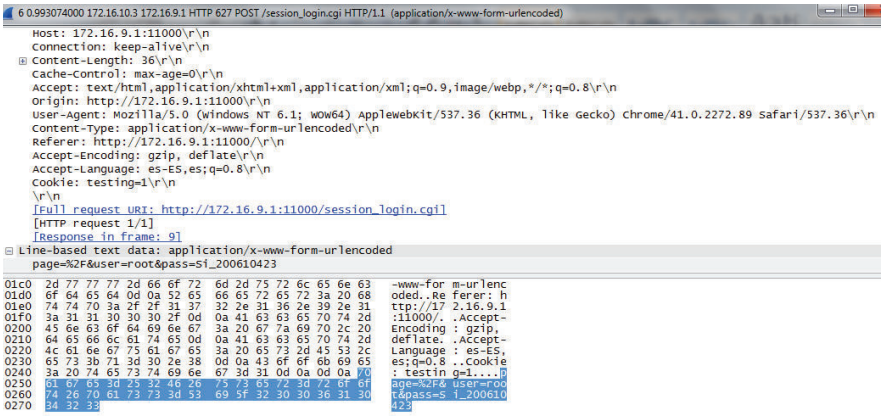


Figura 3-126 Captura Wireshark para acceso a la interfaz Webmin del Servidor Firewall/Control de Contenido

3. Acceso al servidor Ftp con el usuario de prueba creado

En la figura 3-127 se muestra la captura Wireshark cuando se accede al contenido del usuario “shirla” del servidor Ftp local, confirmando que las credenciales se transmiten en texto plano sin ningún método de cifrado.

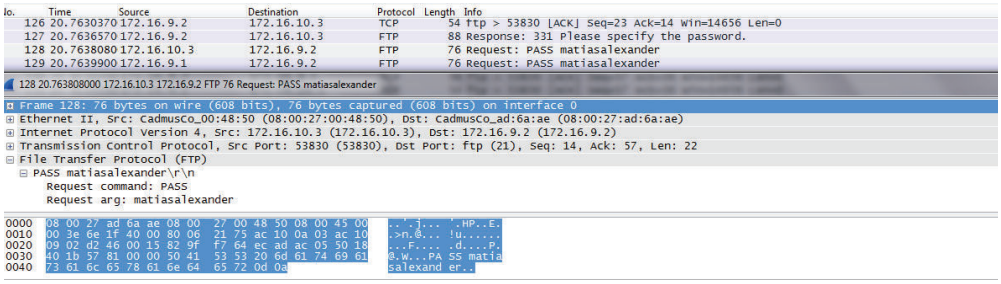


Figura 3-127 Captura Wireshark para el acceso a la carpeta local del usuario “shirla” del Servidor Ftp.

4. Documentación de privilegios obtenidos

En la tabla 3-6 se documenta todas las credenciales obtenidas tras el proceso de acceso a los servicios de las pruebas 1,2 y **3** de la sección actual.

Servicio	Credenciales	
	User	Password
Administración del sitio Web	root	matiasalexander
Webmin	root	Si_200610423
Ftp	shirla	matiasalexander

Tabla 3-6 Credenciales de los servicios del Perímetro de Red

j) Verificar la fortaleza de la autenticación a través del cracking de claves y re-aplicación de claves descubiertas para todos los puntos de acceso que requieren autenticación.

Utilizamos la **Herramienta John the Ripper** para constatar las contraseñas de los usuarios en los servidores del perímetro de red. Luego se documenta todos los privilegios obtenidos.

1. Objetivo: Servidor Web/Ftp/Dns

Comenzamos la prueba combinando el archivo /etc/passwd[82] y el archivo /etc/shadow[83] para obtener el archivo /var/tmp/mypasswd sobre el cual se ejecutará la herramienta Jhon de Ripper y verificar las credenciales del servidor. La sintaxis del comando utilizado es:

Unshadow /etc/passwd /etc/shadow > /var/tmp/mypasswd

En la figura 3-128 se verifica la aplicación de la herramienta Jhon sobre este archivo con el comando John /var/tmp/mypasswd. Los resultados indican las contraseñas con sus respectivos usuarios para el servidor Web/Ftp/Dns.

[82] Archivo /etc/passwd contiene las contraseñas hasheadas en modo lectura para todo el mundo.
[83] Archivo /etc/shadow contiene las contraseñas del SO y está protegido para que solo root pueda leerlo

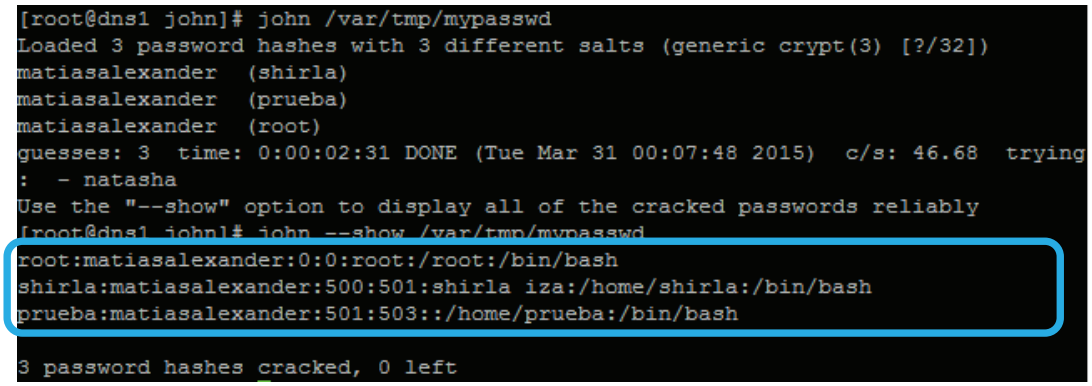


Figura 3-128 Prueba de cracking con John the Ripper al Servidor Web/Ftp/Dns

2. Objetivo: Servidor Firewall/Control de Contenido

En la figura 3-129 se visualiza el resultado de aplicar la herramienta Jhon sobre el servidor Firewall, verificando las credenciales respectivas.

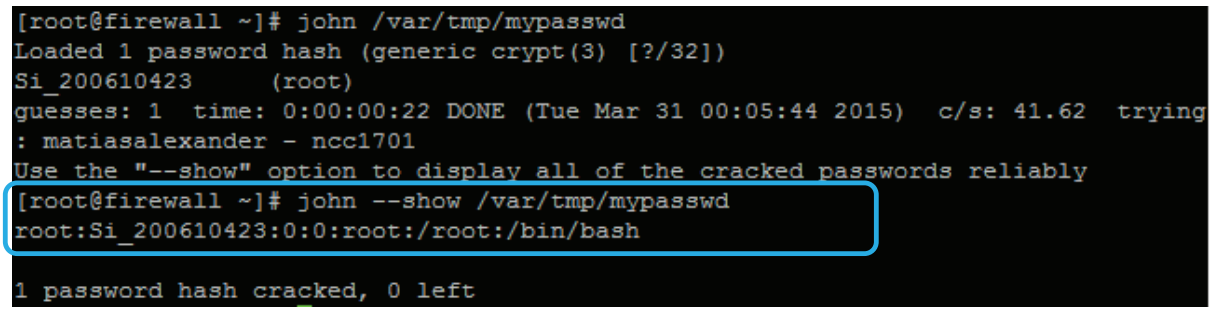


Figura 3-129 Prueba de cracking con John the Ripper al Servidor Firewall/Control de Contenido

3. Documentación de credenciales obtenidas mediante cracking de contraseñas.

En las tablas 3-7 y 3-8 se documentan las credenciales obtenidas mediante la técnica de cracking de contraseña a los servidores del perímetro de red.

Credenciales	
User	**Password**
root	matiasalexander
shirla	matiasalexander
prueba	matiasalexander

Tabla 3-7 Credenciales encontradas con técnicas de cracking al Servidor Web/Ftp/Dns

Credenciales	
User	**Password**
root	Si_200610423

Tabla 3-8 Credenciales encontradas con técnicas de cracking al servidor Firewall/Control de Contenido

k) Verificar el proceso para recibir una autenticación.

Realizaremos un análisis para el proceso de autenticación en el acceso a la administración de la Página Web local, el acceso al servidor Ftp y el acceso a la administración Webmin del Servidor Firewall/Control de Contenido.

1. Proceso de Autenticación para el acceso a la administración de la Página Web local

Este proceso de autenticación se detalla a continuación:

- La autenticación propia utilizada para desplegar la página de administración del Servidor Web, usa una autenticación http básica.
- Luego el Servidor Web valida la autenticación con su propio esquema de usuarios del sistema operativo Linux/Unix.

Autenticación http básica

Esta autenticación permite al navegador web proveer credenciales en la forma de usuario y contraseña cuando se solicita una página al servidor[84]. Se aplica desde el protocolo http 1.0, y se caracteriza por su simplicidad y poca seguridad ya que los datos viajan codificados con el algoritmo base 64 bits[85], por lo cual las credenciales se obtienen mediante un proceso de reversa, sin ningún mecanismo de cifrado. En la figura 3-130 se verifica el proceso para esta autenticación. El primer paso es la petición GET desde el navegador Web para solicitar la página del Servidor. Luego el servidor responde con un error 401 y comienza el proceso de autenticación, en la que se muestra un diálogo al usuario pidiendo el nombre de usuario y contraseña. Entonces el cliente envía la información al servidor con una petición GET anexando una cabecera extra con información de la autenticación. El servidor recibe la nueva petición de página con información de la autenticación, y permite o no el acceso al recurso según la validación de las credenciales suministradas.

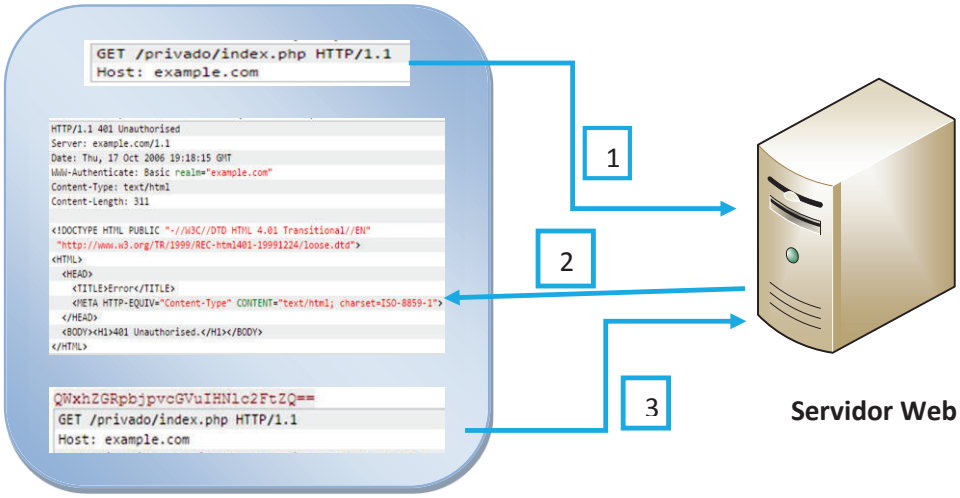


Figura 3-130 Proceso de autenticación básica para acceso a la página de administración del Servidor Web

Autenticación de usuarios en Linux/Unix

En los sistemas Unix los usuarios a nivel local se identifican mediante el UID o

[84] http://es.wikipedia.org/wiki/Autenticaci%C3%B3n_de_acceso_b%C3%A1sica.
[85] Algoritmo Base 64 bits se basa en el uso de los caracteres US-ASCII (no acentuados) para codificar cualquier tipo de información con un código de 8 bits. Fuente: http://es.kioskea.net/contents/56-codificacion-base-64.

identificador de usuario al que se le asocia un nombre y una contraseña. Estos datos se almacenan de forma cifrada en el archivo etc/passwd. El proceso de cifrado de las contraseñas, utilizan un criptosistema irreversible a base de la función C Crypt basada en el algoritmo DES. En la figura 3-131 se verifica los tipos de autenticación disponibles para el sistema Linux, y los que utiliza el modelo clásico de usuarios en Linux, resaltados con el carácter asterisco.

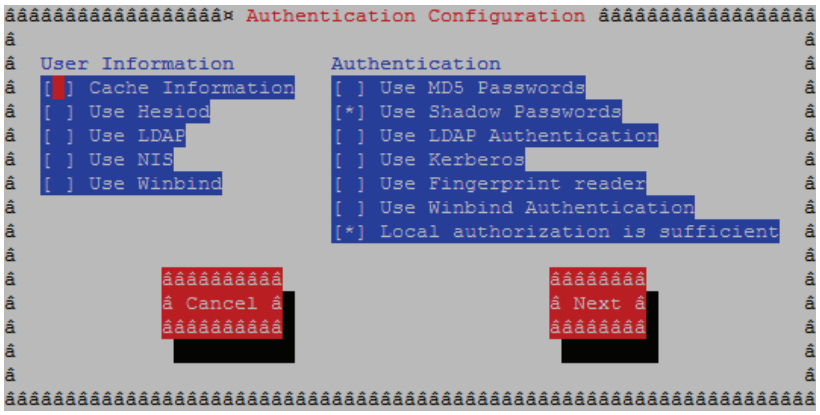


Figura 3-131 Autenticación de usuarios en Linux.

2. Proceso de Autenticación para el acceso al servidor Ftp

En la figura 3-132 se esquematiza los mecanismos de autenticación que utiliza el servidor Ftp.

Anónimo

En este esquema, cualquier cliente remoto se conecta al servidor Ftp mediante la cuenta “anonymous”. Los datos no están encriptados ni sujetos a ningún mecanismo de autenticación, por lo cual es menos seguro.

Autenticado

En este esquema, los usuarios Ftp poseen una cuenta y contraseña específicas para el acceso, aunque las credenciales se transmitan en texto plano sin ningún

mecanismo de cifrado. El acceso del usuario a los directorios y ficheros depende del nivel de permisos para la cuenta utilizada. Para el servidor Ftp local se usa este esquema de autenticación donde el usuario local (shirla), solo tiene acceso a su directorio personal.

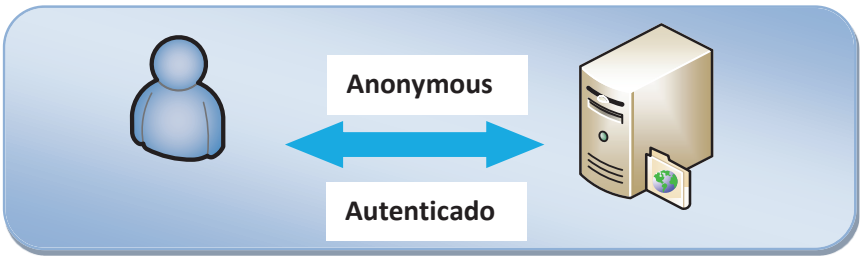


Figura 3-132 Mecanismos de Autenticación en el Servidor FTP

3. Proceso de Autenticación para el acceso a la administración Webmin del Servidor Firewall/Control de Contenido.

El proceso de autenticación se detalla a continuación:

- Autenticación básica: Utiliza un mecanismo sin cifrar, en el cual las credenciales se transmiten en texto plano.

- Sistema de autenticación propio del sistema Linux: Valida las credenciales enviadas por el cliente, tras la autenticación básica.

I) Evaluar en busca de errores lógicos en la aplicación de la autenticación.

Utilizamos la **herramienta Web Scarab** para validar los errores lógicos más importantes en el sistema de autenticación de la aplicación Web.

1. Pruebas de validación para saltarse el método de autenticación

Probaremos tanto el acceso a un recurso interno saltándose la página de autenticación, e intentar saltarse este sistema, mediante manipulación de peticiones http, formularios o falsificación de sesiones.

Verificación de acceso a un recurso interno saltándose la página de registro.

En la figura 3-133 se muestra el resultado de intentar acceder a un recurso interno llamado "Administrator Components" de la administración del sitio Web Primero cerramos la sesión y al intentar pedir este recurso nos muestra un Warning de que no es permitido el acceso directo al recurso. Entonces se confirma que no es posible acceder a páginas internas del Sitio Web tanto con mecanismos de almacenamiento de contraseña como sin ellos en el navegador, lo que denota un punto de defensa ante sistemas que permitan saltarse la autenticación.

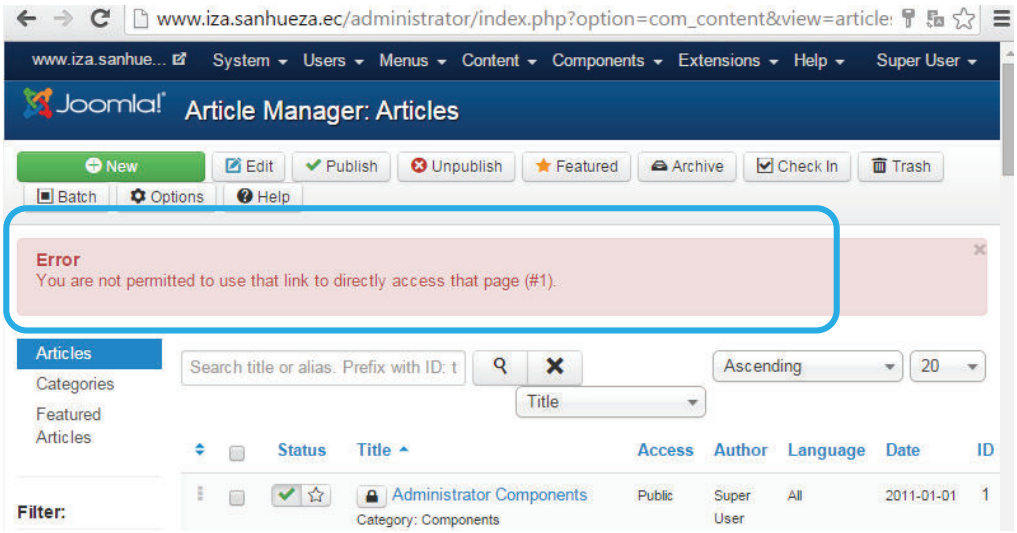


Figura 3-133 Acceso Denegado tras reintento de acceso a un recurso interno del Sitio Web local.

Verificación para saltarse el sistema de autenticación mediante modificación de parámetros en las peticiones http.

Para filtrar las peticiones http con Web Scarab utilizamos la pestaña Intercept y se activa los casilleros Intercept Request y Responses con los métodos Get y Post seleccionados, tal y como se muestra en la Figura 3-134

Figura 3-134 Filtro de Web Scarab para interceptar solicitudes y respuestas tipo GET y POST

Analizamos la petición POST en la figura 3-135 que envía el navegador del cliente al servidor Web una vez que ingresa al portal de administración. Concluimos que no es posible alterar esta petición, ya que no incluye en el código disponible parámetros relacionados a la autenticación, tal como lo es el parámetro “authenticated”.

Figura 3-135 Verificación de la petición POST al realizar la autenticación a la administración del sitio Web local

2. Pruebas de validación que permitan recolectar un conjunto válido de usuarios, interactuando con el sistema de autenticación.

Con frecuencia, las aplicaciones web revelan cuando un usuario existe en el sistema, debido a una mala configuración o error en el diseño[86]. Estas pruebas tratan de explotar esta vulnerabilidad y recolectar usuarios válidos. Los nombres de usuarios usados con frecuencia entre los administradores de sistemas son: "admin",

[86] Enumeración de Usuarios Sección Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf.

"administrator", "root", "system", "super", "qa", "test", "test1", "testing". Si la aplicación es vulnerable a estas pruebas de enumeración de usuarios, y se logra identificar alguno de estos nombres con éxito, entonces se recomienda probar también con las contraseñas de modo parecido.

Prueba de acceso tanto con usuario válido como inválido y sin contraseña.

En la figura 3-136 se muestra esta prueba, con un warning que indica que el campo de password sin valor no es permitido. Esta prueba no nos proporciona información de los usuarios, ya que siempre indicará el mismo warning de que el campo de password es obligatorio.

Figura 3-136 Prueba de validación de usuario sin contraseña.

Prueba de acceso tanto con usuario válido como inválido con el campo de contraseña incorrecto.

En la figura 3-137 se muestra esta prueba, cuyo warning indica que el usuario y contraseña no coinciden o no se tiene aún creada la cuenta. Esta prueba no proporciona datos de los usuarios ya que independiente de que si el usuario sea correcto o no, siempre indica el mismo warning. Realizamos esta prueba indefinidas veces, confirmando que no existe un mecanismo para limitar el número de intentos de log-in con bloqueo de cuenta tras n intentos fallidos.

Figura 3-137 Prueba de validación de usuario y contraseña incorrecta

3. Pruebas de validación para verificar si se gestiona correctamente el almacenamiento y reseteo de la contraseña en el navegador.

En esta subsección realizamos las siguientes pruebas:

Verificación del sistema de reseteo de la contraseña

Mediante pruebas de interacción con la Aplicación Web, comprobamos que esta no cuenta con ningún mecanismo de reseteo de contraseña, en caso de pérdida u olvido de la misma. Estos mecanismos se recomiendan en aplicaciones Web con información sensible, donde se maneja un gran número de usuarios y perfiles de acceso en constante interacción con el sitio Web.

Verificación del sistema de caching de contraseña en el navegador del cliente

Utilizamos la herramienta Web Scarab para analizar e interceptar las peticiones/respuestas http del cliente cuando usa la opción Guardar contraseña en el navegador. Este mecanismo de “recordar mi contraseña" puede ser implementada mediante los siguientes métodos[87]:

- Permitir la característica "cache password" en navegadores web. A pesar de no ser directamente un mecanismo de la aplicación, este método puede y debería ser deshabilitado.

[87] Sección 4.4.6: Comprobar sistemas de recordatorio/reset de contraseñas vulnerables. Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf.

-Almacenar la contraseña en una cookie permanente. La contraseña debe estar cifrada/en formato hash y no ser en texto plano.

En la figura 3-138 realizamos la prueba de acceso al sitio de administración Web, con la opción de “Recordar mi contraseña”. Empieza el intercambio de mensajes POST, GET para la gestión de la contraseña almacenada en una cookie de forma permanente. En la figura 3-139 se aprecia el intercambio de mensajes, en la cual la cookie cifrada se transmite en formato hash. Las credenciales cacheadas en el navegador son enviadas tanto en la fase de registro, como en el acceso a recursos internos del portal Web, lo que denota una falla de seguridad, por traspasar la misma cookie en todas las sesiones activas.

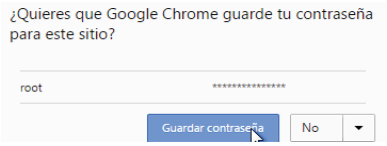


Figura 3-138 Opción de Recordar Contraseña en el sistema Web

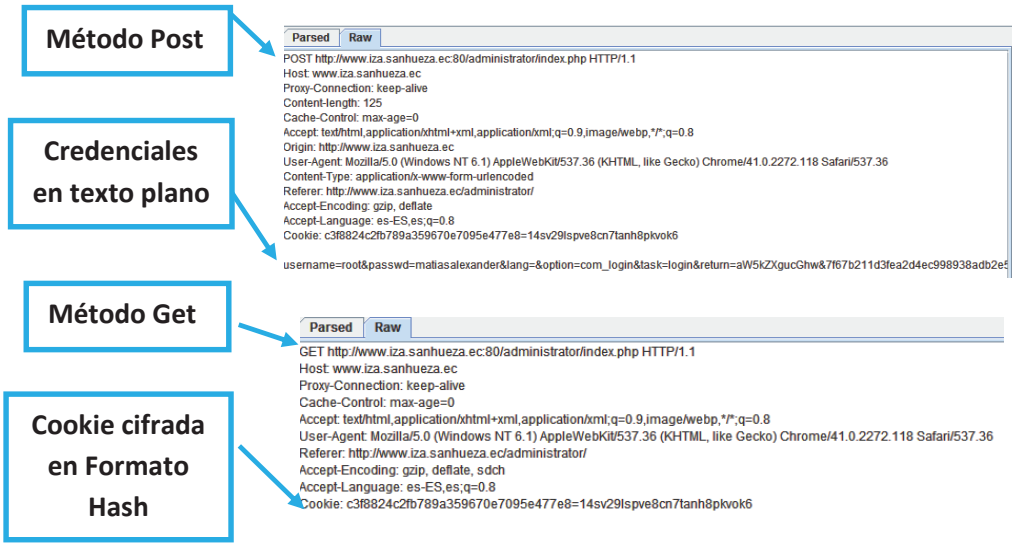


Figura 3-139 Intercambio de mensajes POST y GET para la gestión de contraseña almacenada en una cookie cifrada

4. Pruebas para comprobar la gestión del Caché de navegación y salida de sesión

Los eventos que desencadenan un cierre de sesión son:

Cierre de sesión intencional con la opción Logout del Portal de administración Web.

Es importante verificar que la función de cierre de sesión esté disponible en todas las páginas internas de la aplicación para reforzar la seguridad, y evitar que el usuario olvide esta función al cerrar la aplicación. (Figura 3-140)

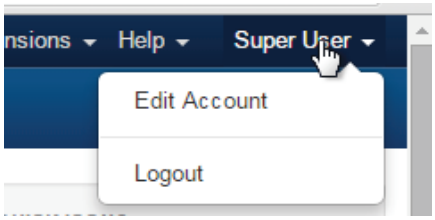


Figura 3-140 Proceso de logout de la administración del Sitio Web.

Inactividad del usuario durante un lapso de tiempo

Esta acción conduce a que la aplicación automáticamente cierre la sesión. Es importante comprobar que ocurre con los testigos de sesión una vez que se invoca la función de cierre de sesión. Realizamos la prueba en la que tras un cierre de sesión, intentamos retroceder en el navegador para acceder al recurso anterior desplegado. En la figura 3-141 se muestra este resultado en el cual no es posible este acceso y se implementa un cierre de sesión efectivo. Esto se realiza mediante la eliminación efectiva de los IDs de sesión en las cookies que obligan a cumplir los controles adecuados en el lado del servidor.

Figura 3-141 Verificación de cierre de sesión efectiva en el Sitio Web.

3.1.2.3 Módulo Verificación de Confianza

Dentro de este módulo documentamos las pruebas: Spoofing, Phishing y Abuso de Recursos.

Pruebas: Spoofing

a) Verificar las formas de acceder a la propiedad dentro del alcance haciendo spoofing de la dirección de red como uno de los sistemas confiados.

Realizamos el ataque denominado Main in the Midle (MITM) hacia el Servidor Web/Ftp/Dns desde una máquina víctima, que en este caso es la máquina de pruebas Windows, consiguiendo interceptar el tráfico proveniente desde la máquina Víctima hacia el Servidor Web. Con el uso de la **Herramienta Cain y Abel** empleamos la técnica de spoofing para capturar el tráfico proveniente de la máquina víctima y capturar las contraseñas para el acceso al sistema Web local y al Servidor Ftp. Los pasos se describen a continuación:

1. Se activa la opción Sniffer y Host del programa, para seleccionar la herramienta Scan Mac Address y escanear todos los equipos de la subred. En la figura 3-142 se muestra el resultado, indicando los hosts disponibles. Escogemos la máquina víctima al host 172.16.10.3 máquina de pruebas Windows.

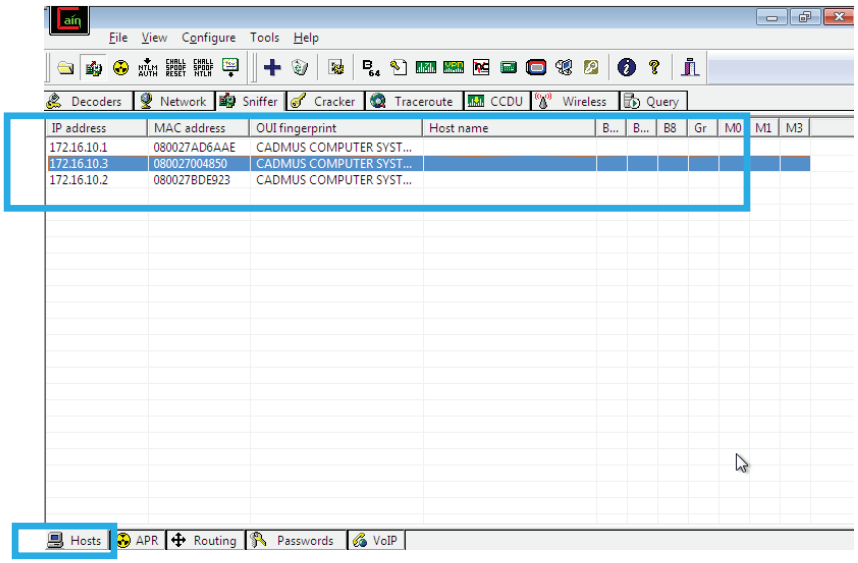


Figura 3-142 Sniffer activado para mostrar hosts disponibles en la subred

2. En la figura 3-143 interceptamos el tráfico generado por la máquina víctima, activando la opción “APR” Poison Routing para seleccionar la máquina víctima en el panel izquierdo y su gateway en el lado derecho. De esta forma, se intercepta el tráfico de la máquina víctima hacia nuestra máquina, sin interrumpir la comunicación entre la víctima y su gateway, tal y como se muestra en la Figura 3-144.

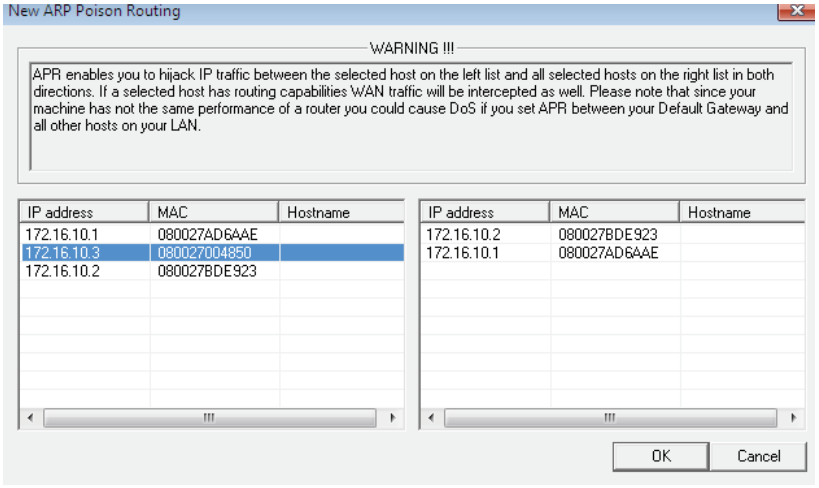


Figura 3-143 Activación de ARP Poisoning de Cain

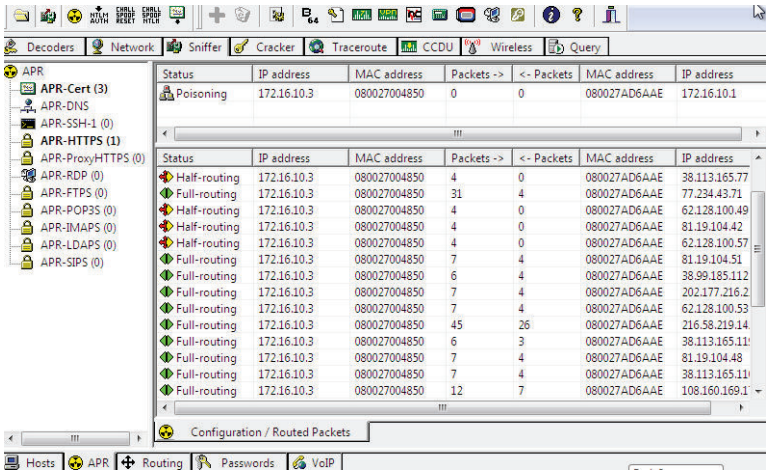


Figura 3-144 Intercepción del tráfico mediante Arp Poisoning

3. En la sección Password en el apartado HTTP, FTP se puede validar las contraseñas y sitios web que accede la máquina víctima. En esta prueba se loguea a la Página de administración del Sitio Web local, y al servidor Ftp con el usuario "shirla". En las figuras 3-145 y 3-146 se muestra los resultados de capturar las contraseñas al realizar los accesos Web y Ftp con el usuario local "shirla".

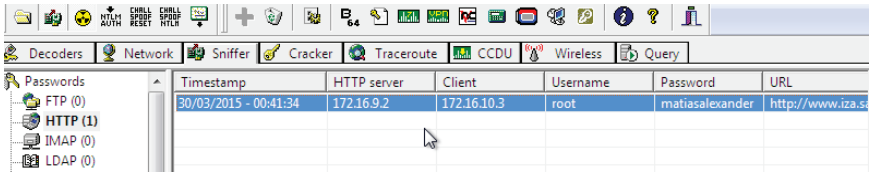


Figura 3-145 Captura de contraseña en el acceso a la administración del Sitio Web local

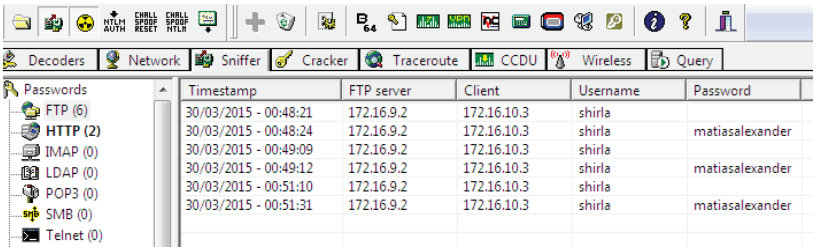


Figura 3-146 Captura de contraseña en el acceso Ftp con el usuario local "shirla

b) Verificar sí los mecanismos de caching pueden ser envenenados

Utilizamos la **Herramienta Cain&Abel** con el módulo ARP-Dns para suplantar la tabla Dns y redirigir la petición de google.com hacia nuestra Página Web local. El objetivo es verificar si el Servidor Web local puede ser contaminado localmente con una entrada en su tabla Dns que conduzca a un Sitio Web de forma intencionada. Los pasos se detallan a continuación:

1. Activamos el dispositivo controlador del Módulo Sniffer de Cain. En la figura 3-147 se muestra esta configuración en el Menú “Configuration Dialog” del Menú Tools de Cain.

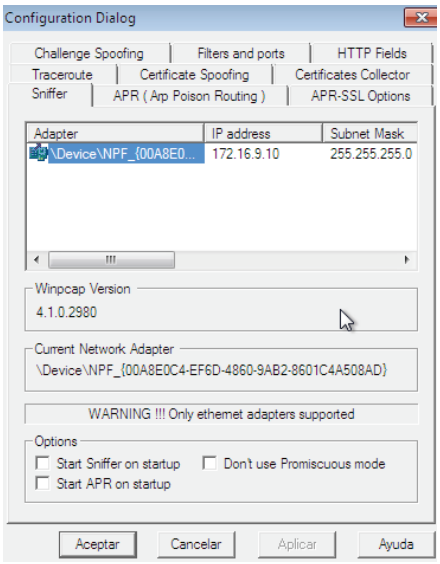


Figura 3-147 Configuración del controlador en Cain

2. En la pestaña APR (Arp Poison Routing) escogemos la opción “usar la Dirección Ip y Mac Address Real de nuestra máquina de pruebas” para realizar el escaneo. En la figura 3-148 se muestra esta configuración.

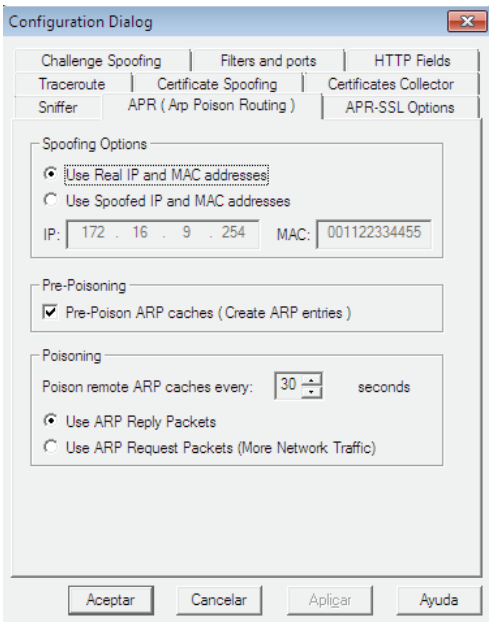


Figura 3-148 Configuración de opciones de Spoofing en el Diálogo de configuración en Cain

3. Activamos la opción "ARP Poison Routing" para interceptar todo el tráfico proveniente de la máquina víctima, técnica denominada Mail in the Midle. Una vez activo el sniffer, configuramos la máquina víctima (Firewall 172.16.9.1) en el panel izquierdo y los Sistemas Objetivos (Segmento de red del Servidor Dns 172.16.9.0/24) en el panel Derecho, tal y como se muestra en la figura 3-149

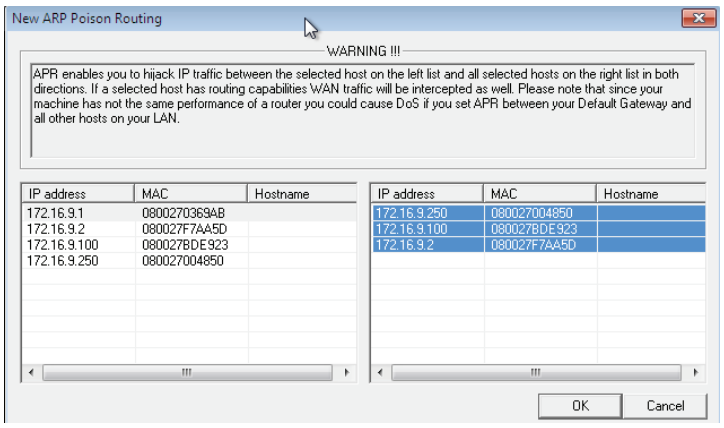


Figura 3-149 Configuración de Arp Poison Routing para interceptar tráfico de la máquina Victima

4. En la figura 3-150 se procede a envenenar la tabla Dns, usando el módulo Arp-Dns, dentro del cual se configura la Web a suplantar (www.google.com), y la Dirección Ip a la cual se redirigirá la petición (Sitio Web 172.16.9.2).

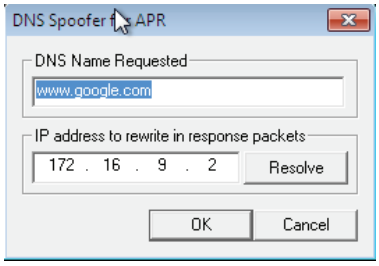


Figura 3-150 Configuración de Spoof Dns para envenenar la tabla Dns

5. Validamos desde la máquina de pruebas Backtrack el acceso a la página www.google.com. En la figura 3-151 se muestra el resultado, verificando la redirección de google.com hacia el sitio Web local. Este resultado confirma que el Servidor Web es vulnerable a este tipo de ataques que permiten envenenar la caché Dns, ya que se originó desde el segmento de red de pruebas, acción que no resultaría si el origen del ataque hubiera sido externo por las políticas de seguridad configuradas en el servidor.

Figura 3-151 Validación de acceso a google.com para redirección hacia el Sitio Web local

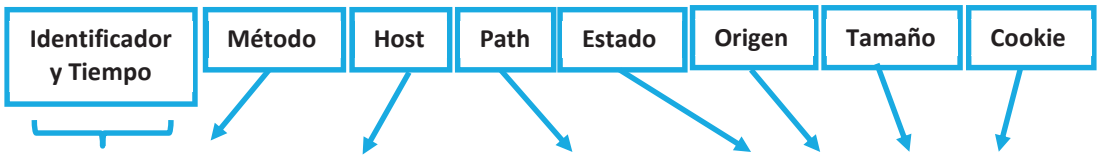
Pruebas: Phishing

c) Verificar que las URL's para el envío y consultas en el objetivo sean concisas, dentro del mismo dominio, usando únicamente el método POST y una marca consistente.

Utilizamos la **Herramienta Web Scarab** para interceptar las URL's en la comunicación cliente (Máquina de pruebas Windows) al servidor Web local y verificar el tipo de peticiones transmitidas. Los pasos se detallan a continuación:

1. Conexión al Sitio Web local "www.iza.sanhueza.ec"

Usamos Web Scarab para capturar las solicitudes y respuestas http originadas por la máquina cliente al conectarse al sitio web local. En la figura 3-152 se muestra el resultado que indica que todas las peticiones son del tipo GET con la siguiente información: tiempo, host, path, estado, Origen, tamaño y cookies.



3	09:30:17	GET	http://www.iza.sanhueza.ec:...	/	200 OK	Proxy	17321	...	
5	09:30:27	GET	http://www.iza.sanhueza.ec:...	/templates/protostar/cs...	200 OK	Proxy	138485		34202...
6	09:30:27	GET	http://www.iza.sanhueza.ec:...	/images/headers/maple....	200 OK	Proxy	35277		34202...
7	09:30:27	GET	http://www.iza.sanhueza.ec:...	/images/safe.png	200 OK	Proxy	37995		34202...
8	09:30:27	GET	http://www.iza.sanhueza.ec:...	/templates/protostar/fa...	200 OK	Proxy	1150		34202...
9	09:31:12	GET	http://www.iza.sanhueza.ec:...	/media/system/js/captio...	200 OK	Proxy	729		34202...
10	09:31:13	GET	http://www.iza.sanhueza.ec:...	/media/system/js/mooto...	200 OK	Proxy	83893		34202...
11	09:31:13	GET	http://www.iza.sanhueza.ec:...	/media/jui/js/bootstrap....	200 OK	Proxy	26454		34202...
12	09:31:13	GET	http://www.iza.sanhueza.ec:...	/media/jui/js/jquery-noco...	200 OK	Proxy	21		34202...
13	09:31:13	GET	http://www.iza.sanhueza.ec:...	/media/system/js/core.js	200 OK	Proxy	3813		34202...
14	09:31:13	GET	http://www.iza.sanhueza.ec:...	/media/jui/js/jquery.min.js	200 OK	Proxy	93637		34202...
15	09:31:18	GET	http://www.iza.sanhueza.ec:...	/media/jui/fonts/IcoMoo...	200 OK	Proxy	18140		34202...

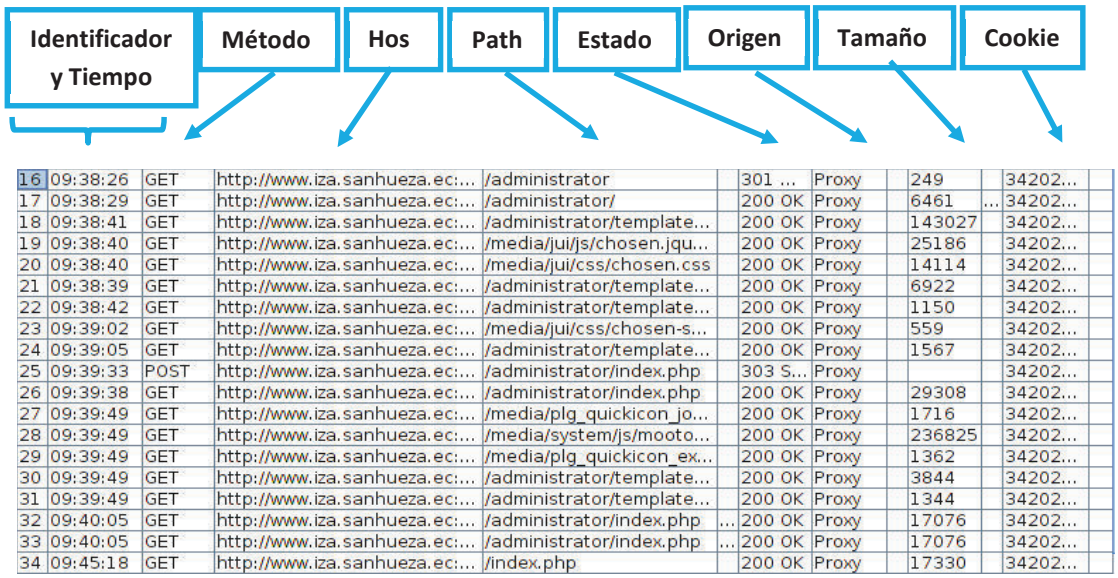
Figura 3-152 Web Scarab para captura de paquetes http en el acceso al Sitio Web local

En la figura 3-153 se visualiza el detalle del paquete http tipo GET, que detalla la información del sitio Web, el puerto y tipo de protocolo Http. El estado del paquete tiene el valor de 200 como respuesta OK de la solicitud aprobada.

Figura 3-153 Web Scarab para detalle del paquete tipo GET

2. Conexión a la Página de administración del sitio Web local

En la figura 3-154 se muestra el resultado de las peticiones http cuando se conecta a la Página de administración del Sitio Web local www.iza.sanhueza.ec/administrator. Los resultados indican que todas las peticiones son del tipo GET, a excepción de un único paquete tipo POST debido al envío de credenciales del usuario administrador al servidor.



16	09:38:26	GET	http://www.iza.sanhueza.ec:...	/administrator		301 ...	Proxy		249	34202...
17	09:38:29	GET	http://www.iza.sanhueza.ec:...	/administrator/		200 OK	Proxy	...	6461	34202...
18	09:38:41	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		143027	34202...
19	09:38:40	GET	http://www.iza.sanhueza.ec:...	/media/jui/js/chosen.jqu...		200 OK	Proxy		25186	34202...
20	09:38:40	GET	http://www.iza.sanhueza.ec:...	/media/jui/css/chosen.css		200 OK	Proxy		14114	34202...
21	09:38:39	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		6922	34202...
22	09:38:42	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		1150	34202...
23	09:39:02	GET	http://www.iza.sanhueza.ec:...	/media/jui/css/chosen-s...		200 OK	Proxy		559	34202...
24	09:39:05	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		1567	34202...
25	09:39:33	POST	http://www.iza.sanhueza.ec:...	/administrator/index.php		303 S...	Proxy			34202...
26	09:39:38	GET	http://www.iza.sanhueza.ec:...	/administrator/index.php		200 OK	Proxy		29308	34202...
27	09:39:49	GET	http://www.iza.sanhueza.ec:...	/media/plg_quickicon_jo...		200 OK	Proxy		1716	34202...
28	09:39:49	GET	http://www.iza.sanhueza.ec:...	/media/system/js/mooto...		200 OK	Proxy		236825	34202...
29	09:39:49	GET	http://www.iza.sanhueza.ec:...	/media/plg_quickicon_ex...		200 OK	Proxy		1362	34202...
30	09:39:49	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		3844	34202...
31	09:39:49	GET	http://www.iza.sanhueza.ec:...	/administrator/template...		200 OK	Proxy		1344	34202...
32	09:40:05	GET	http://www.iza.sanhueza.ec:...	/administrator/index.php	...	200 OK	Proxy		17076	34202...
33	09:40:05	GET	http://www.iza.sanhueza.ec:...	/administrator/index.php	...	200 OK	Proxy		17076	34202...
34	09:45:18	GET	http://www.iza.sanhueza.ec:...	/index.php		200 OK	Proxy		17330	34202...

Figura 3-154 Web Scarab para captura de paquetes http en el acceso a la página de ad

En la figura 3-155 se muestra el detalle del paquete http tipo POST, cuyo encabezado viene acompañado por el path del sitio web y el tipo de protocolo utilizado. Dentro del contenido del paquete se incluye las credenciales del usuario, que no están cifradas y se transmiten en texto plano.

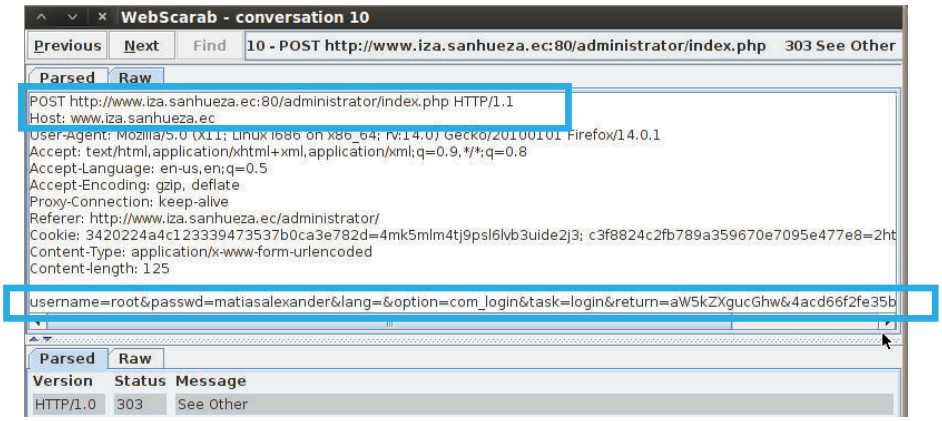


Figura 3-155 Web Scarab. Detalle del paquete http tipo POST

d) Examinar los registros de dominio de alto nivel para dominios similares a los identificados dentro del alcance.

Utilizamos la **herramienta Nic.ec** para examinar si existen registros de dominio de alto nivel gTLD similares al dominio local. En la figura 3-156 realizamos esta prueba validando que el dominio local “iza.sanhueza.ec” no tiene registro a nivel mundial.

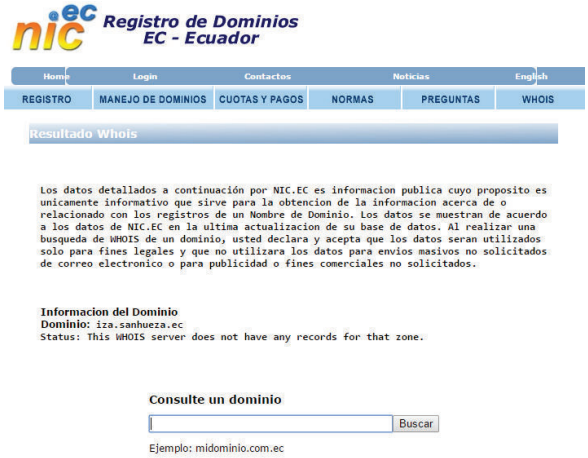


Figura 3-156 Verificación del dominio local “iza.sanhueza.ec” en nic.ec

Realizamos otra prueba para constatar si existen dominios similares registrados, verificando que no existen registros similares. Solo los dominio de segundo nivel “iza.ec” y “sanhueza.ec” se encuentran disponibles, tal y como se muestran en las figuras 3-157 y 3-158 respectivamente.

Figura 3-157 Verificación del dominio iza.ec en Nic.ec

Figura 3-158 Verificación del dominio Sanhueza.ec en Nic.ec

Pruebas: Abuso de Recursos

d) Evaluar la profundidad del acceso a la información de negocios o confidencial disponible en servidores web sin credenciales establecidas.

En esta sección aplicamos un análisis detallado para determinar los puntos críticos de los datos presentados por el sitio Web y analizar cuál información debería o no ser revelada por la misma. Es importante medir la relación Exposición de Información sensible Vs. Seguridad de la información, a fin de garantizar el funcionamiento robusto de la aplicación y disminuir las amenazas de seguridad a la misma. Analizamos la interfaz Home del Sitio Web local, validando la siguiente información:

1. Revela información general del proyecto actual.
2. Muestra el diagrama de funcionalidad de los elementos y/o sistemas que intervienen en el proyecto.
3. Indica las pruebas de seguridad que se implementan en el proyecto.
4. Revela información de los objetivos del mismo.
5. Incluye información de las Fases de la metodología usada por el proyecto.

3.1.3 RESULTADO DE PRUEBAS: FASE DE INVESTIGACIÓN

En esta sección se documentan los resultados de aplicar las pruebas de la fase de investigación, detalladas por la metodología OSSTMM. En esta fase se desarrollará el módulo Verificación de Configuración.

3.1.3.1 Módulo Verificación de Configuración

Dentro de este módulo se documentan las pruebas: Controles de configuración, Errores de configuración comunes y Mapeo de Limitaciones.

Pruebas: Controles de Configuración

a) Examinar los controles para verificar las configuraciones y líneas base de los sistemas, equipos y aplicaciones para cumplir con la intención de la organización y reflejar una justificación de negocio.

En esta sección analizamos los controles de privacidad aplicado a los servicios del Perímetro de red (Servicios Web, Ftp y Dns), a fin de verificar las configuraciones y líneas base de las mismas, conforme a la intención del proyecto y que refleje la justificación del mismo.

1. Control para verificar configuración del Servidor Web/Ftp y limitar sus respuestas

Los controles de privacidad que implementamos para el servidor Web/Ftp se detallan a continuación:

- **Control de acceso y conectividad**

  Este control lo implementamos mediante políticas de acceso y restricción en el Servidor Firewall para permitir el flujo de tráfico entrante y saliente de los servicios utilizados (Web y Ftp) y conexión a internet. Además se realizó configuraciones de seguridad en el sistema Kernel del Servidor Firewall para evitar ataques de desbordamiento Syn de Tcp contra el servidor Web/Ftp.

- **Control para limitar las respuestas del servidor**

  Limitamos las respuestas del servidor mediante configuraciones de seguridad en el archivo apache “http.conf” para el servidor Web y en el archivo “vsftpd.conf” para el servidor Ftp. Con esta limitación evitamos mostrar información tal como versión y tipo de servicio, que pueda conducir a un atacante reunir información suficiente para realizar pruebas de intrusión sobre el objetivo.

2. Control para verificar configuración del Servidor DNS y limitar las consultas Dns

Los controles de privacidad que implementamos para el servidor Dns se detalla a continuación:

- **Control para limitar la transferencia de zona Dns**

  Este control limita la transferencia de zona Dns únicamente al propio servidor DNS a fin de limitar ataques externos como: Contaminación del Cache del Servidor, redirección de peticiones Dns a sitios falsos, consultas de subdominios del servidor, etc. Utilizamos configuraciones de seguridad en las zonas directa e inversa del servidor.

- **Control para restringir las consultas Dns**

  Este control permite las consultas Dns solo desde la red interna, a fin de bloquear el forward de consultas desde el Internet y evitar ataques de envenenamiento de caché Dns que puedan re direccionar las peticiones hacia servidores falsos.

b) Examinar Access Control Lists (ACL) y funciones de negocios configurados en redes, sistemas, servicios y aplicaciones en el alcance para garantizar que cumplan la intención de la organización y reflejen una justificación de negocio.

Estas pruebas consisten en verificar las ACLs representadas por las políticas de iptables en el Servidor Firewall y las reglas de control de acceso Http definidas en el Servidor Squid Control.

1. Revisión de Políticas de Iptables en el Servidor Firewall

En el iptables del servidor, configuramos las reglas para permitir el tráfico entrante y saliente del tipo http, https, correo, ftp, telnet y dns que cumplen con las políticas e intención del proyecto al implementar los servicios del Perímetro de Red. Para

permitir este tráfico configuramos las secciones Filter y Nat, mediante cadenas “Input, Forward y Postrouting” respectivamente en el iptables del servidor.

2. Reglas de control de acceso Http en el Servidor Squid Control

En el servidor Squid Control configuramos reglas de restricción para el tráfico http no permitido. Se prohibió el contenido pornográfico y redes sociales. Estas configuraciones las implementamos mediante ACLs de tipo Client Address para definir los segmentos de red Lan que se les permite la navegación (Segmento de red de pruebas y Segmento del Servidor Web/Ftp/Dns) y del tipo Url Regex para definir las palabras asociadas a la regla de restricción del tráfico pornográfico y redes sociales.

Pruebas: Errores de Configuración Comunes

c) Verificar servicios disponibles que no son redundantes innecesariamente y que coinciden con el rol definido para la intención de los sistemas.

Los servicios disponibles en el perímetro de red cumplen con los objetivos e intención del proyecto actual. Conforme a los objetivos definidos en la sección 1.3, planteamos la entrega de un documento formal de pruebas de intrusión en la red perimetral, diseñada bajo los lineamientos de la Metodología Safe de Cisco. Esta metodología estableció los siguientes servicios y sus roles para implementar el perímetro de la red.

- **Servidor Web/Ftp:** Publica un Portal Web con información general del proyecto actual.
- **Filtro Url**: Establece controles de acceso en la navegación http.
- **Servidor Dns:** Establece el servicio de traducción de nombres de dominio a direcciones Ip y viceversa a nivel de la red local.

- **Servidor Firewall:** Implementa políticas de control de acceso a los servicios/aplicaciones dentro del Perímetro de Red.

d) Verificar que los parámetros default han sido cambiados. Algunos dispositivos o aplicaciones se entregan con una cuenta administrativa default u oculta. Estas cuentas deberían ser cambiadas o, sí es posible, inhabilitarlas o eliminarlas y reemplazarlas con una nueva cuenta administrativa.

A continuación detallamos si los servicios del Perímetro de red están expuestos a configuraciones y cuentas administrativas por defecto.

1. Servidor Web:

No entrega ninguna cuenta administrativa por defecto u oculta. En cuanto a configuraciones, implementamos el gestor de contenido con Joomla, el cual mantiene su presentación default. También se cambiaron parámetros del archivo apache “httpd.conf” para limitar las respuestas del Servidor a los clientes y en el sistema Kernel del Servidor Firewall para evitar ataques de desbordamiento Tcp Syn.

2. Filtro Url:

No entrega ninguna cuenta administrativa por defecto u oculta. En cuanto a configuraciones, se mantiene parámetros por defecto, tales como la característica de Proxy Transparente para no tener que definir ningún proxy en los navegadores cliente y puedan acceder directamente a los servicios Http, basándose en las reglas de acceso de este servidor. Con frecuencia esta configuración no es recomendable, ya que puede existir un mal uso o abusos de acceso a Internet por parte del personal interno no autorizado.

3. Servidor Dns:

No entrega ninguna cuenta administrativa por defecto u oculta. En cuanto a configuraciones se cambiaron parámetros en el archivo de configuración named.conf

para limitar las consultas dns que se permitan solo desde la red interna y en los archivos de configuración de zona directa e inversa para limitar la transferencia de zona únicamente al propio servidor a fin de evitar ataques externos.

4. Servidor Ftp:

Este servidor si entrega una cuenta por defecto llamada “anonymous” para que pueda ser utilizado por cualquier usuario. En cuanto a configuraciones se limitó el banner para que no muestre información de la versión, mediante la directriz ftp_banner del archivo de configuración “vsftpd.conf”.

e) Verificar que la administración se realice localmente o con controles para limitar quien o que puede acceder por interfaces remotas de administración del equipo

Para la administración de los servicios Dns/Ftp/Web del perímetro de red, se dispone del protocolo ssh. Mientras que para el Servidor Firewall/Control de contenido también manejamos los protocolos telnet y Webmin. De estos protocolos, el más seguro es ssh al proporcionar un canal cifrado para la transmisión de los datos, en tanto que telnet y Webmin no implementan mecanismos de encriptación en el canal de comunicación, por lo cual los datos viajan en texto plano. El control para limitar el acceso a la administración remota viene dada por el proceso de autenticación. Solo el usuario con mayores privilegios (root) se le permite administrar y realizar cualquier cambio en los servicios del perímetro de red.

Pruebas: Mapeo de Limitaciones

f) Chequear por servicios/características disponibles no necesarias en el alcance.

En esta sección se detallan los servicios disponibles en el perímetro de red y que no cumplen una función necesaria para los objetivos e intención del proyecto.

1. Objetivo: Servidor Web/Ftp/Dns

Previamente en la tabla 3-1 detallamos los servicios disponibles en el perímetro de red para este servidor. De esta información se confirma que todos los servicios son necesarios, ya que cumplen su función independiente de acuerdo a las directrices de diseño del módulo Internet Corporativo de Cisco Safe. Los dispositivos que plantea esta metodología son: Servidor Dns para la traducción de las solicitudes dns, Web/Ftp para proporcionar información pública del proyecto actual y transferencia de archivos a nivel local. También habilitamos el servicio ssh para el cifrado de los datos en el canal de comunicación durante el proceso de administración, y el protocolo Https para la navegación Web segura utilizando el protocolo de seguridad SSL.

2. Objetivo: Servidor Firewall/Control de Contenido

Previamente en la tabla 3-2 detallamos los servicios disponibles en el perímetro de red para este servidor. Todos los servicios son necesarios, a excepción de Telnet, el cual introduce una amenaza de seguridad por enviar los datos de la conexión cliente-servidor en texto plano sin ningún mecanismo de cifrado. Los otros servicios cumplen con las directrices de diseño de Cisco Safe, tales como el Servidor Firewall para brindar protección a nivel de recursos y filtro con estado del tráfico, Servidor Filtro de Direcciones Url para limitar las conexiones http, Servicio Http para permitir la navegación Web, Servicio Ssh para permitir las conexiones remotas seguras y el Servicio Webmin para administración a los servicios del Servidor.

3.1.4 RESULTADO DE PRUEBAS: FASE DE INTERVENCIÓN

En esta sección documentamos los resultados de aplicar las pruebas de la fase de intervención, detalladas por la metodología OSSTMM. Los módulos dentro de esta fase son: Auditoría de Privilegios y Validación de Supervivencia.

3.1.4.1 Módulo Auditoria de Privilegios

En este módulo documentamos las pruebas: Autorización y Escalación de privilegios.

Pruebas: Autorización

a) Examinar y verificar cualquier forma de obtener una autorización fraudulenta para ganar privilegios similares a las del otro personal.

Las pruebas que realizamos en esta sección son la enumeración de Vectores de Entrada, que permiten verificar las partes de la aplicación que son vulnerables ante las entradas del usuario, tales como consultas Http Get y Post, opciones de carga de archivos y formularios Html.

En la figura 3-159 se muestra el resultado de utilizar la herramienta Web Scarab para verificar la petición Get, una vez que el usuario root se loguea al sitio de administración web. Validamos que la variable index.php es asignada a un valor específico y que las cookies usadas por la aplicación para la generación dinámica de páginas se encolan conforme a las sesiones mantenidas con el sitio web. Estos resultados indican que la variable index.php no admite valores dinámicos que puedan alterarse en la captura de los paquetes.

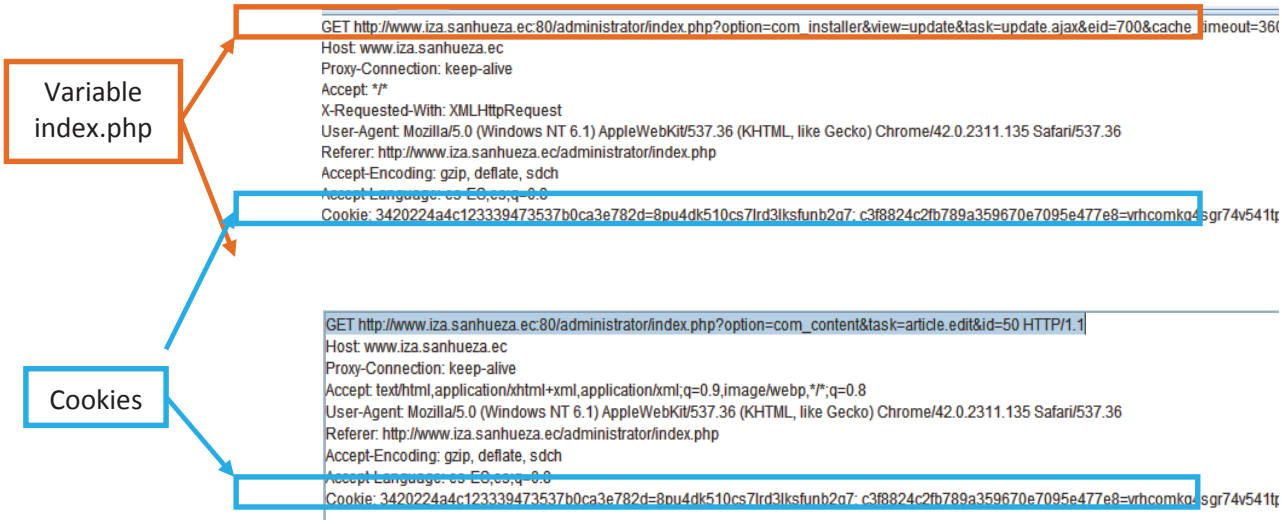


Figura 3-159 Petición get para identificación de variable index.php y cookie

b) Pruebas para evadir el esquema de autenticación y acceder a información reservada/sensible en el sistema.

En esta sección se realizan las siguientes pruebas:

1. Llamadas directas a páginas internas, saltándose la página de registro.

En la figura 3-160 se muestra el resultado de esta prueba, al tratar de acceder a un recurso interno del aplicativo web mediante la Url, y automáticamente nos redirige a la página de Loguin. Se confirma que el sistema no permite saltarse la página de registro, lo que proporciona mayor seguridad en el sistema de acceso.

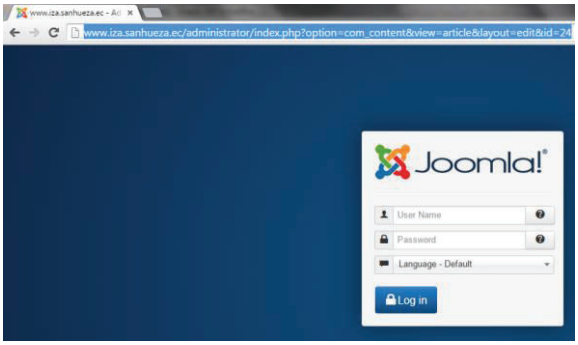


Figura 3-160 Llamada directa a Página Interna del sitio web saltándose la página de registro

2. Modificación de las peticiones para engañar al sistema de autenticación mediante técnicas de manipulación de formularios, URL's o falsificando sesiones.

La prueba consiste en llamar a un recurso interno, el fichero /etc/passwd a través de la Url del aplicativo Web. El resultado indicado en la Figura 3-161 confirma que el método de validación de entrada con la consulta Http Get se ha implementado con protección ya que no permite la llamada a lectura/escritura de archivos en el sistema.

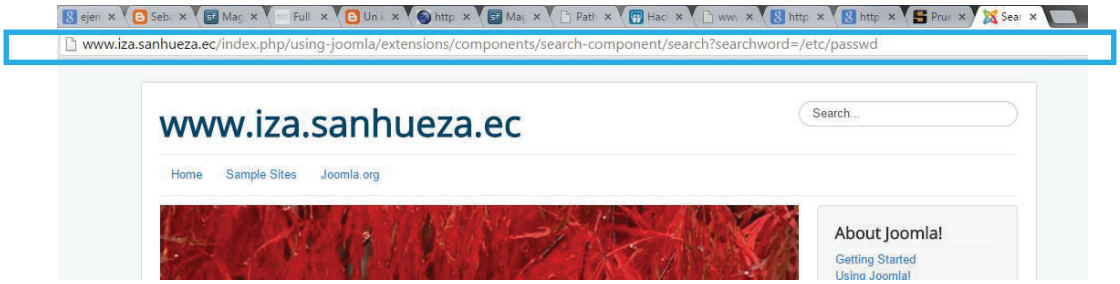


Figura 3- 161 Llamada del recurso /etc/passwd en la Url del aplicativo Web

3. Predicción de los IDs de sesión.

Utilizamos la herramienta Web Scarab, para analizar los IDs de sesión cuando el usuario administrador accede al aplicativo web e interactúa con la misma. En la figura 3-162 se visualiza los paquetes interceptados. Los resultados indican una sola cookie asociada a esta acción.

ate	Meth...	Host	Path	P...	Status	...	...	Cookie	Comments	Sc
)5/21...	GET	http://www.i...	/ad...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/ad...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/ad...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/ad...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/m...		200 OK	...		c3f8824c2fb789a359670e7095e477e8=f52ml1hlffunq8ueravt66ij23	☐	
)5/21...	GET	http://www.i...	/ad...		200 OK	...	...		☑	
)5/21...	GET	http://www.i...	/ad...		301 Mo...	...			☐	

Figura 3-162 Cookie asociada al login de administración del aplicativo web.

En la figura 3-163 se muestra la cookie generada cuando se navega desde la página de administración hacia el link del Sitio web. El resultado indica una nueva cookie diferente que incluye de igual forma caracteres entre números y letras.

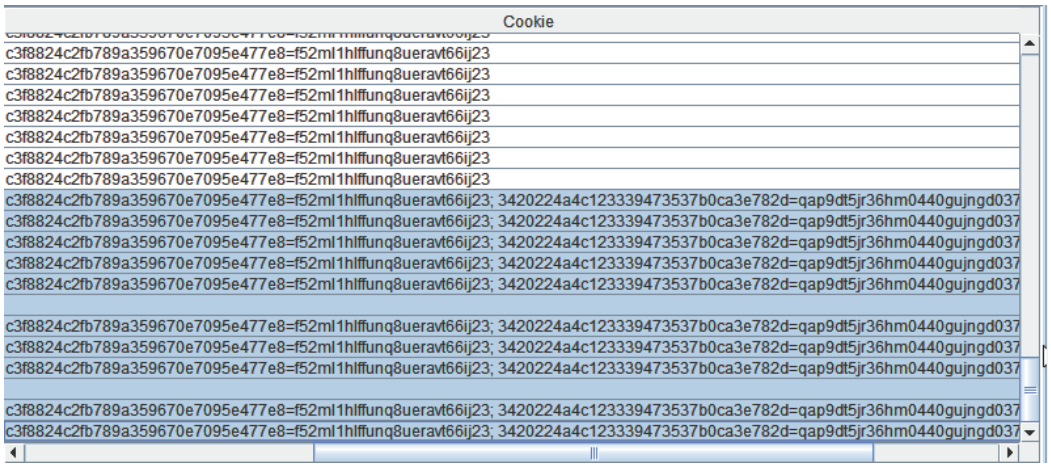


Figura 3-163 Cookie asociada al navegar al sitio web desde la página de administración.

En la figura 3-164 se muestra el resultado del logout de la página de administración. Como efecto se elimina la primera cookie y se mantiene la segunda cookie generada cuando se navegó al sitio web. Se agrega una nueva cookie por la acción del deslogueo (esta última cookie se crea por la opción de guardar contraseña que se activó inicialmente). Una vez que nos volvemos a loguear a la página de administración con el usuario guardado, sigue esta última cookie. Estos resultados afirman que no existe aleatoriedad en la generación de cookies ante la interacción de las sesiones con el aplicativo web. Esto conlleva a dificultar el trabajo de predicción de los Ids de sesión, y refuerza el sistema de autenticación.

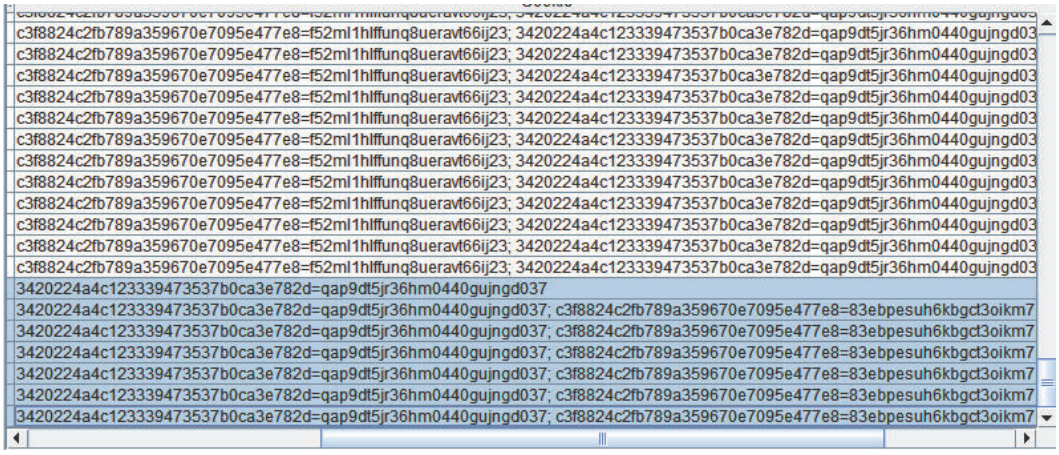


Figura 3-164 Resultados de cookies cuando se desloguea del sitio Web

Pruebas: Escalación de privilegios

c) Coleccionar información acerca de personas con altos privilegios. Buscar por roles o posiciones de confianza, gateways de acceso para personas de confianza y cualquier medio físico tales como tokens o tarjetas inteligentes.

Para el sitio web del proyecto actual definimos los siguientes roles de usuario:

- **Administrador**: Es el usuario que tiene todos los privilegios en la administración del aplicativo web.
- **Otros:** Usuarios con privilegios en modo lectura para visualización del aplicativo web.

Además de definir los tipos de roles, también es importante especificar los privilegios asignados para determinar quién o qué puede realizar en el sistema y el grupo al que pertenecen. En la tabla 3-9 se detalla esta información.

Privilegios	Quién puede hacerlo	Grupo de usuario
Añadir, modificar artículos	root	SuperUsuario
Administrar menús, usuarios y medios	root	SuperUsuario
Administrar el lenguaje, extensiones y módulos	root	SuperUsuario
Editar el perfil y configuración global	root	SuperUsuario
Visualizar en modo lectura el sitio web	otro	ninguno

Tabla 3-9 Tipos de roles con sus privilegios y grupos de usuarios.

d) Verificar los límites de los privilegios en los objetivos y si los medios existen para escalar estos privilegios

Utilizamos el **Exploit Mempodipper** para validar si es posible la escalada de privilegios del usuario “shirla” al super usuario “root”. En la figura 3-165 se muestra la

ejecución de este exploit en el servidor Dns/Web/Ftp. El resultado confirma que el servidor no es vulnerable a este CVE: 2012-0056, por lo cual no es posible realizar escalada de privilegios en el sistema.

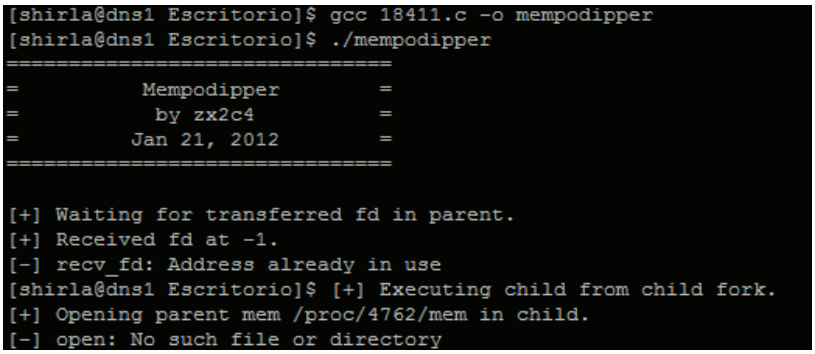


Figura 3-165 Ejecución del exploit Mempodipper sobre el servidor Web

3.1.4.2 Módulo Validación de Supervivencia

En este módulo se documentan las pruebas de Resiliencia detalladas a continuación:

Pruebas: Resiliencia

a) Verificar los puntos únicos de fallo (cuellos de botella) en la infraestructura donde el cambio o el fracaso puede causar una interrupción del servicio.

Los puntos únicos de fallo que pueden interrumpir los servicios son:

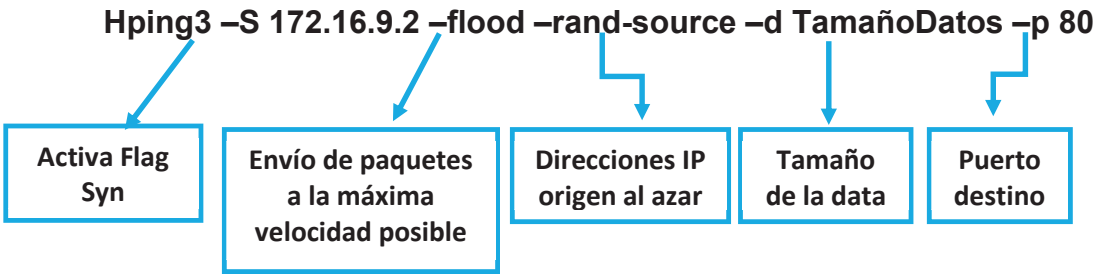
- **Servidor Web/Ftp:** Si este servicio llega a ser vulnerable, impedirá a los clientes acceder al portal del aplicativo Web, así como permitir la subida y bajada de archivos en el Servidor FTP.
- **Servidor Dns:** Si este servicio llega a ser vulnerable, bloqueará al Servidor de responder las solicitudes Dns de los clientes que deseen utilizar los servicios de

navegación Web y todo tipo de traducciones de nombres calificados a direcciones IP.

b) Verificar el impacto de acceso a los objetivos en la cual se puede causar fallas del sistema o servicio.

Utilizamos la **Herramienta: hping3** para realizar ataques de denegación de servicio sobre los sistemas objetivos Servidor Web/Ftp.

Hping3 en combinación de las opciones –S y –flood nos permite enviar un flood de peticiones con el bit Syn activado. Las opciones –rand source, -d, y –p permiten aleatoriedad en las IPs origen falsificado, definir el tamaño de la data y el puerto objetivo. El comando utilizado es:



1. Objetivo Servidor Web/Ftp

El escenario de pruebas se detalla en la figura 3-166. Utilizamos las 3 máquinas del segmento Lan de pruebas, para ejecutar simultáneamente los ataques de denegación de servicio con hping3 al servidor Web. En el primer escenario contemplamos el ataque con la limitación del parámetro del kernel "net.ipv4.tcp_syncookies" a 1, en la que el servidor no espera peticiones de respuesta ACK ante las peticiones Syn que envía el remitente. En el segundo escenario se cambia el valor de este parámetro a 0 para indicar que el servidor espere la confirmación ACK del lado del remitente, y por tanto no puede abrir nuevas conexiones hasta que se concluya la conexión inicial de 3 vías de TCP.

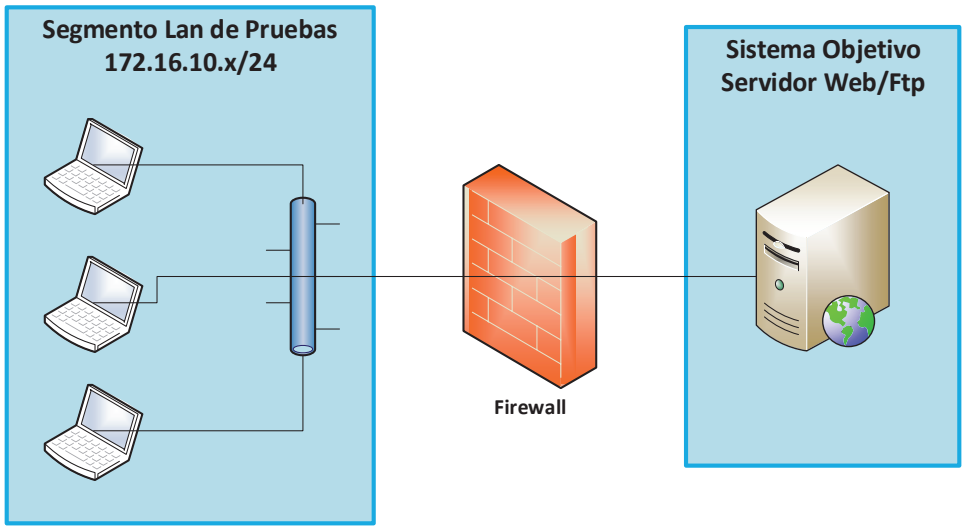


Figura 3-166 Diagrama de pruebas para ataque de Denegación de Servicio al Servidor Web/Ftp

En el primer escenario se envía las solicitudes detalladas en la figura 3-167 desde las máquinas de prueba hacia el servidor Web. Los resultados indican que se puede acceder al sitio Web local sin ningún inconveniente. En el segundo escenario, también se envían las solicitudes desde la red Lan del segmento de pruebas y se confirma que el Sitio Web local está indisponible (Figura 3-168).

```
[root@localhost ~]# hping3 -S 172.16.9.2 --flood --rand-source -d 2000000000 -p 80
HPING 172.16.9.2 (eth0 172.16.9.2): S set, 40 headers + 37888 data bytes
hping in flood mode, no replies will be shown
```

```
[root@localhost ~]# hping3 -S 172.16.9.2 --flood --rand-source -d 1000000000 -p
80
HPING 172.16.9.2 (eth0 172.16.9.2): S set, 40 headers + 51712 data bytes
hping in flood mode, no replies will be shown
```

```
root@bt:~# hping3 -S 172.16.9.2 --flood --rand-source -d 1000000000 -p 80
HPING 172.16.9.2 (eth0 172.16.9.2): S set, 40 headers + 51712 data bytes
hping in flood mode, no replies will be shown
```

Figura 3-167 Pruebas de Denegación de Servicio con hping3 al Servidor Web

Figura 3-168 Indisposición del sitio Web local ante pruebas de DoS

3.2 ANÁLISIS DE RESULTADOS

Una vez finalizada la etapa de documentación de pruebas y resultados descritos en este proyecto, se realiza el análisis de resultados que permiten al analista de seguridad reducir los falsos positivos, negativos y errores humanos en las pruebas descritas del actual proyecto.

3.2.1 REDUCIR FALSOS POSITIVOS Y NEGATIVOS

Las pruebas realizadas permiten constatar el estado de un sistema/servicio dinámico en períodos de tiempo variables. El analista de seguridad intenta probar los sistemas del perímetro de red, poniéndolos al límite, y simulando situaciones que permitan validar sus funciones de control. Los errores que con mayor frecuencia se presentan en estas pruebas son:

Falso positivo: Situación en que se detecta un error que no existe

Falso negativo: Situación en que no se detecta un error que si existe

Error humano: Presentado por la falta de experiencia en manejo y/o conocimiento de herramientas de testeo para la auditoría de seguridad.

3.2.2 REDUCIR ERRORES HUMANOS

Para reducir los errores humanos se recomienda que el personal asignado para las pruebas de seguridad, tenga experiencia y conocimientos avanzados en el manejo de las herramientas que van hacer utilizadas, para minimizar en lo posible los riesgos de errores humanos.

3.2.3 RESULTADOS ESPERADOS

Con el objetivo de cumplir esta sección se desarrolló las pruebas descritas en la sección 3.1.1 “Documentación de pruebas y Resultados” del proyecto actual. El análisis de resultados esperados, según los resultados obtenidos se describe en la Tabla 3-10.

Prueba	Resultado Esperado	Resultado Obtenido
Verificación de la información de registro de dominio.	El analista de seguridad debe contar con un listado detallado de la información de registro de dominio, tales como Dns primarios, secundarios, entidad registrante, nombre del administrador, fecha de creación/caducidad, y registros correspondientes al dominio.	Se obtuvo información de los registros tipo A, SOA, NS, ttl con los cuales se verificó el host asociado al dominio, el servidor DNS primario, los servidores autoritativos de la zona dns y el tiempo de propagación cuando existen cambios con el dominio.
Calidad de la red.	El analista de seguridad debe contar con un listado detallado de las pruebas de calidad de la red, que incluyen uso del	Se obtuvo los valores de la tasa de transferencia, el tiempo, y porcentaje de peticiones servidas ante solicitudes Tcp e Icmp.

	ancho de banda utilizado por la red del perímetro, así como los paquetes TCP e ICMP perdidos, y el promedio tanto para las solicitudes de servicio como respuesta de paquetes para cada protocolo.	
Tiempo.	El analista de seguridad debe contar con un listado detallado de las pruebas de tiempo, que incluyen la documentación de la zona horaria actual de las pruebas, así como los valores de TTL al gateway y los sistemas dentro del perímetro de red.	Se detalló la zona horaria de las pruebas, así como los valores de TTL a los sistemas objetivos, validando los nodos intermedios para alcanzarlos.
Filtrado	El analista de seguridad debe contar con una lista detallada de las pruebas de filtrado que permitan verificar si existen filtros a nivel de las comunicaciones en los sistemas auditados.	Se detalló los filtros en cuanto a navegación web del protocolo http con el servidor Squid Proxy Control y el firewall del perímetro de red con el Software Iptables. Los filtros de la navegación web definen bloqueos a nivel de tráfico no permitido (redes sociales, pornografía). Los filtros del Firewall permiten solo el tráfico para los servicios del perímetro de red, la redirección de las peticiones web al servidor proxy, mientras que el resto de tráfico se bloquea.

Detección Activa	El analista de seguridad debe contar con una lista detallada de las pruebas de detección activa, que permita verificar las respuestas activas a las pruebas de los sistemas auditados.	Se obtuvo las respuestas de los servicios activos en el perímetro de red, mostrando los puertos abiertos, latencia, plataforma del Sistema Operativo, tiempo y detalle de paquetes enviados y recibidos.
Survey de la Red	El analista de seguridad debe contar con una lista detallada de las pruebas de reconocimiento de la red, que permita identificar los sistemas auditados, mediante escaneos a nivel de tráfico, protocolos, puertos abiertos, filtrados/cerrados, trazado de rutas e identificación de establecimiento de las conexiones en las interacciones con el perímetro de red.	Se obtuvo información de los protocolos y puertos utilizados en el perímetro de red (http, dns, Icmp, tcp, udp, ftp, ssh, telnet) validando los servicios activos. También se validó las trazas de rutas a los objetivos verificando los nodos intermedios y se analizó en detalle el protocolo de comunicación de 3 vías que Tcp establece para la conexión con los sistemas auditados.
Enumeración	El analista de seguridad debe contar con una lista detallada de las pruebas de enumeración, que permita detallar las respuestas de los sistemas ante solicitudes TCP, UDP para enumerar todos los sistemas dentro de la red. Así como el registro de logs y	Se obtuvo las respuestas de los sistemas activos en el perímetro de red ante las solicitudes TCP con distintas banderas del protocolo, así como de las solicitudes UDP. Los resultados validan puertos abiertos, filtrados/cerrados para cada servicio auditado, así como el

	validación del código fuente de la aplicación web.	registro de log y el código fuente de la aplicación web.
Identificación	El analista de seguridad debe contar con una lista detallada de las pruebas de identificación, que permita detallar las respuestas de los sistemas para validar pruebas de Fingerprinting y obtener información como el TTL, uptime, servicios, aplicaciones, y las vulnerabilidades correlacionadas.	Se obtuvo información de los TTLs a los sistemas auditados, así como el uptime, tipo de sistema operativo y pruebas de Fingerprinting al sitio web local que identificó su versión, SO, Dirección IP, tipo de CMS, etc.
Red	El analista de seguridad debe contar con una lista detallada de las pruebas de Acceso a la red, que permita detallar los accesos ante las solicitudes de servicios TCP o UDP, e incluso manipular el enrutamiento para saltarse restricciones en el acceso a los sistemas auditados.	Se obtuvo información de la respuesta Dns ante consultas al servidor, validando el establecimiento de conexión, la traducción y la respuesta del servidor Dns. Además se validó que es posible el enrutamiento de paquetes cuando se estableció la conexión al puerto 53 del servidor DNS.
Servicios	El analista de seguridad debe contar con una lista detallada de las pruebas de Acceso a los servicios, que permita detallar información de los banners y lograr identificar el	Se obtuvo información de los banners de los sistemas del perímetro de red. Se identificó el tipo y versión del servicio Web, Ssh, Squid. Mientras que para el servidor FTP únicamente se

	uptime, tipo y versión de los servicios auditados para correlacionar con posibles vulnerabilidades asociadas.	muestra el login de conexión. En base a estos resultados se analizó las vulnerabilidades en los sistemas auditados
Autenticación	El analista de seguridad debe contar con una lista detallada de las pruebas de Acceso para autenticación, que permita detallar el proceso de acceso a los sistemas auditados a fin de obtener credenciales de acceso, definir el canal de autenticación utilizado, verificar si es posible realizar cracking de claves e identificar errores lógicos en el sistema de autenticación.	Los resultados obtenidos indican que el canal de autenticación no es seguro ya que las credenciales de acceso se transmiten en texto plano y es posible obtenerlas mediante técnicas de cracking de claves. Además se constató el proceso de tipo http básica y basada en sistema Linux para recibir la autenticación en el acceso a los sistemas auditados. Se validó también los errores en el sistema de autenticación, como falta de mecanismos que limiten el número de intentos de log-in con bloqueo de cuenta tras n intentos fallidos, falta de un sistema de reseteo de contraseña en caso de pérdida u olvido, transporte de la misma cookie originada por la autenticación en todas las sesiones del usuario.
Spoofing	El analista de seguridad debe contar con una lista detallada de las pruebas de Spoofing a fin de constatar el acceso a	Los resultados reflejan que el sistema web local y el servidor FTP pueden ser atacados mediante la técnica de “Main in the

	los sistemas auditados desde fuentes spoofeadas haciéndose pasar como sistemas confiados.	Middle" para interceptar el tráfico destinado a estos sistemas y obtener las credenciales de acceso. Además se comprobó la posibilidad de envenenar la cache del servidor DNS para re direccionar peticiones desde la Red interna.
Phishing	El analista de seguridad debe contar con una lista detallada de las pruebas de Phishing a fin de determinar información sensible del sistema auditado, valiéndose de vulnerabilidades en las transacciones electrónicas, mail, redes sociales, malware infiltrado, etc.	Los resultados confirmaron que el acceso al sistema web local utilizan métodos para el envío y consultas con peticiones GET, y una vez que el usuario administrador se loguea utiliza solo una petición POST para el envío de credenciales en el cuerpo de la petición. También se evaluó que no existen dominios de alto nivel Gtld similares al dominio local.
Abuso de Recursos	El analista de seguridad debe contar con una lista detallada de las pruebas de Abuso de Confianza a fin de evaluar si es posible el acceso a los sistemas e información sensible sin credenciales establecidas	Los resultados detallan la información desplegada por el sitio web local a fin de constatar la relación Exposición de Información sensible con la Seguridad de la Información.
Controles de Configuración	El analista de seguridad debe contar con una lista detallada de las pruebas de Controles	Los resultados detallan las configuraciones y líneas base de los sistemas del perímetro de red.

	de Configuración a fin de examinar y evaluar las configuraciones y líneas base de los sistemas auditados y validar si cumplen bajo la intención y políticas planteadas en la organización.	Se constató control de acceso, conectividad y límite en las respuestas proporcionadas por el servidor Web/Ftp. Se validó controles para limitar la transferencia de zona y restringir las consultas en el servidor Dns. Se verificó control en el tráfico entrante y saliente del perímetro de red mediante las políticas de iptables del servidor Firewall, así como control en la navegación web en el servidor Squid Control Proxy.
Errores de Configuración	El analista de seguridad debe contar con una lista detallada de las pruebas de Errores de Configuración que permita validar la funcionalidad de los servicios auditados que cumplan con la intención de la organización, así como constatar que los parámetros por defecto hayan sido cambiados.	Los resultados validan el rol de funcionamiento que cumple cada servicio en el perímetro de red. Se detalló si en los sistemas auditados se cambiaron los parámetros por defecto. Para el caso del servidor Web/Ftp se limitó las respuestas a los clientes, así como configuraciones de seguridad para limitar ataques de desbordamiento Syn de Tcp. Para el servidor Squid Proxy se mantuvo las configuraciones por defecto haciéndolo vulnerable en cuanto al abuso del acceso a internet por la característica de

		Proxy Transparente. En el servidor Dns se limitó la transferencia de zona y las consultas Dns. También se verificó los controles de acceso a los servicios validando vulnerabilidades de seguridad con los protocolos telnet y Webmin.
Mapeo de Limitaciones	El analista de seguridad debe contar con una lista detallada de las pruebas de Mapeo de Limitaciones para constatar los servicios/características disponibles en el alcance y que no son necesarios.	Los resultados indican que para el servidor Web/Ftp/Dns todos los servicios son necesarios para cumplir las funcionalidades del perímetro de red planteado en el capítulo 1 sección 1.4.1.3. Para el servidor Firewall/Control de Contenido todos los servicios son necesarios a excepción de telnet que introduce amenazas en el acceso al sistema por el envío de las credenciales en texto plano.
Autorización	El analista de seguridad debe contar con una lista detallada de las pruebas de Autorización para verificar si es posible obtener una autorización fraudulenta y acceder a información sensible de los sistemas auditados	Los resultados indicaron que no es posible obtener una autorización fraudulenta para el acceso al sitio web local, ya que las variables de las peticiones tipo Get enviadas al servidor no envían valores dinámicos que puedan alterarse en la captura de los paquetes. También se realizaron pruebas para evadir el sistema de autenticación, tales como:

		Restricción al no permitir saltarse la página de registro, métodos de validación de entrada http get robusto al no permitir la llamada a lectura/escritura de archivos en el sistema, y aleatoriedad en los IDs de sesión generados al interactuar con el sitio web local.
Escalación de Privilegios	El analista de seguridad debe contar con una lista detallada de las pruebas de Escalación de Privilegios para constatar si los límites en los privilegios pueden ser escalados.	Los resultados detallan los diferentes roles y privilegios asignados en los sistemas del perímetro de red. También se constató que no es posible escalar privilegios en el servidor Dns bajo la vulnerabilidad CVE: 2012-0056.
Resiliencia	El analista de seguridad debe contar con una lista detallada de las pruebas de Resiliencia para verificar la disponibilidad y consistencia de los servicios auditados cuando se los expone a interrupciones prolongadas del servicio.	Los resultados exponen pruebas de denegación de servicio mediante hping3 al servidor web local validando que este se indispone únicamente cuando el firewall espera peticiones de respuesta ACK ante las peticiones Syn con la opción “net.ipv4.tcp_syncookies” del Kernel del servidor.

Tabla 3-10 Tabla de Resultados Esperados según los resultados obtenidos

3.3 GUÍA RESUMEN CAPÍTULO 3

En esta sección se detalla brevemente una guía resumen para los procedimientos desarrollados en el capítulo actual.

3.3.1 HERRAMIENTAS Y COMANDOS UTILIZADOS PARA EL DESARROLLO DEL TOOLKIT DE PRUEBAS DE INTRUSIÓN

En las Tablas 3-11, 3-12, 3-13 y 3-14 se detallan las pruebas y las Herramientas/Comandos utilizados para el desarrollo del Toolkit de Pruebas de Intrusión según la fase recomendada por la metodología OSSTMM V3.

Fase	Prueba	Herramientas /Comandos
Fase de Inducción	Framework	1. Comando: Dig (Dominio auditado) 2. Comando: Host –a4 (Dominio auditado)
	Calidad de la Red y Tiempo	3. Comando: ab –n (#consultas) –c (nivel de concurrencia) (Servidor Web) 4. Comando: ping –c ((#consultas) (Host Objetivo)
	Filtrado	5. Procedimientos manuales para verificación de filtros en los sistemas auditados
	Detección Activa	6. Herramienta: Nmap –T (tiempo de la prueba) –A (Información detallada) –v(Incrementa el nivel de detalle) Objetivo

Tabla 3-11 Pruebas y Herramientas/Comandos para la Fase de Inducción

Fase	Prueba	Herramientas/Comandos
Fase de	Survey	1. Herramienta: Wireshark para filtrar tráfico de los servicios

Interacción	de la Red	auditados y verificar los números de secuencia TCP ISN. 2. Herramienta: Nmap –sO (Host Objetivo) Realiza exploración de Protocolos IP 3. Herramienta: Nmap –PE (Host Objetivo) Escaneo de solicitudes con el tipo de paquete ICMP #15 4. Herramienta: Nmap –PP (Host Objetivo) Escaneo de solicitudes con el tipo de paquete ICMP #13 para huella de tiempo 5. Herramienta: Nmap –PM (Host Objetivo) Escaneo de solicitudes con el tipo de paquete ICMP #18 para Host que aparentemente están indisponibles 6. Herramienta: Nmap –sU (Host Objetivo) Activa el escaneo para puertos UDP 7. Comando: Ping –t (#ttl) (Host Objetivo) 8. Comando: Tracert (Host Objetivo)Traza la ruta de los paquetes a su destino 9. Comando: Traceroute –p (#puerto a probar) (Host Objetivo) Traza la ruta de los paquetes TCP con un puerto específico 10. Comando: Traceroute –P udp –p (#puerto) (Host Objetivo) Traza la ruta de los paquetes UDP para un puerto específico
	Enumeración e Identificación	11. Herramienta: Firebug de Firefox analiza el código fuente del sitio web local. 12. Herramienta: Tail –f (Archivo del servicio) Verifica los logs en tiempo real 13. Herramienta: Nmap –sU –p- (Host Objetivo) Escaneo ante solicitudes Udp a los puertos 0 – 65535 14. Herramienta: Nmap –sU --source-port (#puerto origen) –p U:(# puerto destino) (Host Objetivo) Escaneo UDP con

		puerto origen y destino 15. Herramienta: Nmap –sU –p U:(# puerto destino) –badsum (Objetivo) Escaneo Udp con la opción de checksum incorrecto. 16. Herramienta: Nmap –sU -sV –p U:(# puerto destino) (Objetivo) Activa el escaneo a los puertos malware de acceso remoto UDP 17. Herramienta: Nmap –sS –p- (Objetivo) Escaneo Tcp Syn a los puertos 1-65535 18. Herramienta: Nmap –p (# puerto) Escaneo de solicitudes TCP en los puertos indicados. 19. Herramienta: Nmap –sA –source-port 80 –p (rango de puertos destino) (Objetivo) Escaneo de tipo TCP Ack con puerto origen y destino. 20. Herramienta: Nmap –f –p (puertos destino) (Objetivo) Escaneo con paquetes IP fragmentados. 21. Herramienta: Nmap –sN –p(# puerto destino) (Objetivo) Escaneo con todas las banderas TCP activadas 22. Herramienta: Nmap –sF –p(# puerto destino) (Objetivo) Escaneo con la bandera FIN activada 23. Herramienta: Nmap –sX –p(# puerto destino) (Objetivo) Escaneo con las banderas FIN, PUSH y URG activadas 24. Herramienta: Hping2 (Host Objetivo) Verifica parámetros como longitud del paquete, ttl, secuencia, tamaño de ventana, identificación del paquete ID, etc. 25. Herramienta: Nmap –Pn –PO –p- -Si (Host Zombie) (Host Objetivo). Activa escaneo Idle Scan 26. Comando: Ping (Objetivo) Identificar respuestas TTLs de los objetivos.

		27. Herramienta: Nmap –v –O (Objetivo) Activa el escaneo Fingerprinting 28. Comando: ./whatweb –v (Sitio Web) 29. Comando: ./cms-explorer.pl –url (Sitio Web) –type (Tipo de CMS) Verifica los plugins instalados en el sitio web local
	Red, Servicios y Autenticación en la Red	30. Comando: nc –ulp 5300 > /tmp/dns-query.bin Genera la petición DNS con la herramienta Netcat para escuchar en un puerto arbitrario 31. Comando: dig@localhost –p 5300 (sitio web) Genera la petición Dig al servidor DNS local 32. Comando: ls –l /tmp/dns-query.bin Comprueba la captura del archivo generado por consulta al sitio web local. 33. Comando: Cat /tmp/dns-query.bin ! nc –vv –q 5 –u (Objetivo) (puerto) > /tmp/dns-reply.bin Captura la respuesta del servidor DNS local 34. Comando: ls –l /tmp/dns-reply.bin Verifica el archivo de respuesta del servidor Dns local 35. Comando: nc –vv –u –g (gateway) (Objetivo) (Puerto Destino) Manipula el enrutamiento de paquetes hacia un host Objetivo. 36. Comando: Nmap –vv (Servicio_Objetivo) (# puerto) Verifica los banners de los servicios auditados 37. Comando: nc –vv –u –p 1337 –l Habilita la escucha de conexiones en el puerto 1337 38. Comando: nc –vv –u (Ip_Servidor) (# puerto) Conexión de un cliente a un servidor en el puerto habilitado 39. Comando: ./nikto.pl –host http://www.iza.sanhueza.ec/mutillidae/ Permite verificar si el sitio web local está afectado por la vulnerabilidad XSS

		40. Comando: xssf_active_victims para verificar si el sistema auditado es vulnerable a XSSF 41. Comando ./sqlmap.py –u http://www.iza.sanhueza.ec/index.php?id=1 Verifica si el sitio web local es vulnerable a Inyecciones Sql 42. Herramienta Wireshark para verificar credenciales de acceso al sitio web local 43. Procedimientos manuales para verificar el sistema de autenticación del sitio web local 44. Herramienta: Nmap –Sv –v –p ((# puerto) (Objetivo) Identifica el tipo y versión del servicio auditado. 45. Comandos: use exploit/unix/ftp/vsftpd_234_backdoor", "set RHOST 172.16.9.2" y "exploit" para verificar si el servidor Ftp es vulnerable a Backdoor Vsftpd_234. 46. Comando: Service named status para validar versión del Servidor Dns 47. Procedimientos manuales para validar vulnerabilidades en el servidor Dns y Web. 48. Comando: John --show /var/tmp/mypasswd Permite obtener las credenciales de acceso de un sistema mediante cracking de contraseñas. 49. Herramienta Web Scarab para validar errores lógicos en la autenticación al sitio web local
	Spoofing, Phishing y Abuso de Recursos	50. Herramienta Cain/Abel para realizar ataque "Main in the Midle" a los sistemas auditados y para comprobar si los mecanismos de caching Dns pueden ser envenenados. 51. Herramienta Web Scarab para verificar el tipo de peticiones http en la conexión del usuario administrador al sitio web local.

		52. Herramienta nic.ec para examinar si existen registros de dominio de alto nivel gTLD similares al dominio local. 53. Procedimiento manual para evaluar la exposición de información sensible en el perímetro de la red.

Tabla 3-12 Pruebas y Herramientas/Comandos para la Fase de Interacción

Fase	Prueba	Herramientas /Comandos
Fase de Investigación	Controles y Errores de Configuración	Procedimientos manuales para verificar los controles y errores de configuración en los sistemas auditados.
	Mapeo de Limitaciones	Procedimientos manuales para verificar los servicios disponibles en el perímetro de red y que no cumplen funciones necesarias para la intención del proyecto

Tabla 3-13 Pruebas y Herramientas/Comandos para la Fase de Investigación

Fase	Prueba	Herramientas /Comandos
Fase de Intervención	Autorización y Escalación de privilegios	1. Herramienta: Web Scarab para verificar los componentes de la aplicación que pueden ser vulnerables ante accesos maliciosos. 2. Procedimientos manuales para evadir el sistema de autenticación y acceso a información sensible. 3. Exploit Mempodiper para realizar pruebas de Escalación de privilegios
	Resiliencia	4. Herramienta Hping3 –S 172.16.9.2 –flood –rand-source –d TamañoDatos –p 80 para realizar ataques de denegación de servicios sobre los sistemas auditados. Las opciones –S y –flood nos permite enviar un flood de peticiones con el bit Syn activado.

		Las opciones –rand source, -d, y –p permiten aleatoriedad en las IPs origen falsificado, definir el tamaño de la data y el puerto objetivo.

Tabla 3-14 Pruebas y Herramientas/Comandos para la Fase de Intervención

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

En la presente sección se detallan las conclusiones y recomendaciones derivadas del desarrollo de los capítulos del actual proyecto. Las conclusiones se las define en la sección 4.1 y las recomendaciones en la sección 4.2.

4.1 CONCLUSIONES

- El diseño del toolkit de pruebas de intrusión realizado en el proyecto actual aseguran procedimientos y buenas prácticas de seguridad estándar descritas por la metodología de seguridad OSSTMM V3 en el canal de Redes de datos. En la sección 2.1 de análisis de requerimientos para el desarrollo del toolkit de pruebas de intrusión se detalló cada uno de los módulos necesarios para cumplir las fases de la metodología. Las pruebas escogidas para cumplir los objetivos de cada módulo, permitieron validar los controles establecidos para los sistemas en el perímetro de red.

- El esquema del perímetro de red utilizado en el proyecto actual asegura cumplir con las recomendaciones para el diseño de infraestructura, servicios, aplicaciones y redes seguras, bajo los lineamientos de la metodología de seguridad Safe de Cisco en los contornos de las pequeñas y medianas empresas. En la sección 2.3 se detalló el análisis de requerimientos para el desarrollo del perímetro de la red, que incluye los elementos tanto en software como hardware. Las políticas de seguridad de estos elementos, permitieron evaluar las funcionalidades y controles de los sistemas/servicios en el perímetro de la red.

- En base a los resultados obtenidos al aplicar las pruebas de intrusión sobre la red perimetral, se determinó que la metodología Cisco Safe cumple con directrices de seguridad robustas, que minimizan en gran parte las amenazas y vulnerabilidades de los sistemas y servicios auditados.

- El presente toolkit de pruebas de intrusión propone una guía pública de procedimientos formales de pruebas de seguridad, que un analista en la materia pueda tomar de referencia para evaluar la seguridad de las redes perimetrales a nivel de pequeñas y medianas empresas.

- El toolkit de pruebas de intrusión empezó con una verificación de los controles, el alcance y las limitaciones en el perímetro de la red. Las pruebas de intrusión detalladas en la sección 3.1.1.1 del documento actual, establecieron las pruebas necesarias para la preparación del entorno del canal y evitar resultados imprecisos en las pruebas, tales como falsos positivos y negativos. Las pruebas de intrusión detalladas en la sección 3.1.1.2, permitieron detectar los diferentes filtros para los controles activos y/o pasivos en el perímetro de red.

- En las secciones 3.1.2.1 y 3.1.2.2 se implementó una fase de interacción con los sistemas/servicios del perímetro de la red, comenzando con una auditoría de visibilidad y acceso a los sistemas descubiertos dentro del alcance. Posteriormente en la sección 3.1.2.3 se realizó pruebas de confianza para validar el acceso a los servicios sin la necesidad de autenticarse. En la sección 3.1.3 se implementó una fase de investigación en la que se verificó la forma en que trabajan los sistemas, sus configuraciones de control y servicios, así como la exposición de información sensible publicada al internet.

- En la sección 3.1.4.1 se desarrolló las fases de auditoría y Escalación de privilegios, para verificar y probar los sistemas de autenticación, autorización y

escalamiento de privilegios del aplicativo web. Se comprobó la robustez del control de autenticación tanto en la gestión de sesiones, con IDs de sesión generados aleatoriamente y la restricción en la escalada de privilegios para acceder a roles con mayores privilegios de los proporcionados. Finalmente en la sección 3.1.4.2 se desarrolló la sección de validación de supervivencia en la cual se probó la resistencia y continuidad del Servidor Web mediante pruebas de stress. Se validó la indisponibilidad del Aplicativo web cuando en el firewall no se configuran apropiadas medidas de seguridad para evitar los ataques tipo Syn Flood.

- El toolkit de pruebas de intrusión basado en los lineamientos de la metodología OSSTMM abarca las pruebas de seguridad necesarias desde sus inicios con la fase de recolección de información hasta la intervención en los sistemas, para garantizar una auditoria de seguridad fiable.

4.2 RECOMENDACIONES

- Se recomienda tomar como referencia los lineamientos de la metodología OSSTMM V3 en el análisis del canal de las redes de datos. OSSTMM incluye módulos de pruebas que permiten a un analista de seguridad incluir las fases apropiadas para el diseño de auditorías de seguridad, así como procesos y recomendaciones estándar basadas en metodologías como OWASP, ISO 17999-2000, entre otras.

- Se recomienda tomar como referencia en el diseño de redes seguras los lineamientos de la metodología Safe de Cisco sección Contorno de la Empresa. Esta metodología además de incluir las directrices de diseño basadas en arquitecturas de redes seguras, también maneja un enfoque

modular donde el crecimiento y seguridad es diseñado, implementado y evaluado en bloques funcionales.

- Se recomienda una documentación detallada en el proceso de análisis de seguridad, que permita definir una evaluación sistemática de riesgos, amenazas y fortalezas en el perímetro de la red, para finalmente tomar las decisiones gerenciales apropiadas. Todo esto a fin de mejorar las políticas, procedimientos internos y seguridad en las redes perimetrales.

- Se recomienda que los analistas de seguridad, previo a realizar auditorías de seguridad tengan una capacitación intensiva en el conocimiento y manejo de las herramientas de seguridad para el desarrollo de las pruebas de intrusión. Esto con el fin de minimizar en lo posible los errores humanos que puedan causar interpretaciones incorrectas en el análisis de resultados.

REFERENCIAS BIBLIOGRÁFICAS

LIBROS Y MANUALES

[L1] Cisco System, (2000). Cisco Safe “Un modelo de seguridad para las redes de las empresas”. [Online]. Disponible en: http: //www.cisco.com/web/ES/assets/docs/safe_wp1.pdf [2013, 25 de Enero].

[L2] ISECOM, (2010). OSSTMM “Manual de la Metodología Abierta de Testeo de Seguridad”. [Online], Versión 3. Disponible en: http://www.isecom.org/mirror/OSSTMM.3.pdf [2012, 25 de Agosto].

[L3] Fundación OWASP, (2008). Guía de Pruebas OWASP “Proyecto Abierto de Seguridad de Aplicaciones Web”. [Online], Versión 3.0. Disponible en: https://www.owasp.org/images/8/80/Guía_de_pruebas_de_OWASP _ver_3.0.pdf [2014, 6 de Febrero].

PROYECTOS DE TITULACIÓN

[T1] COELLO, Maria Gabriela. Procedimiento Formal de Ethical Hacking para la infraestructura tecnológica de los servicios por internet de la Banca Ecuatoriana, Facultad de Sistemas, Escuela Politécnica Nacional, Ecuador, 2012.

[T2] IZA, Leyla Jennifer. VILLAREAL, Jorge. Diagnóstico de vulnerabilidades del entorno informático del Hospital de niños “Baca Ortiz” utilizando Ethical Hacking, Facultad de Sistemas, Escuela Politécnica del Ejército, Ecuador, 2011.

DIRECCIONES ELECTRÓNICAS

[W1] Instalación DNS Centos 6.2 (2012, Septiembre). [Online]. Disponible en: http://jogagiro.wordpress.com/2012/09/02/instalacion-dns-centos-6-2/ [2012, 20 Diciembre].

[W2] Instalación FTP Centos 6.2 (2012, Septiembre). [Online]. Disponible en: http://jogagiro.wordpress.com/2012/09/15/instalacion-ftp-centos-6-2/ [2012, 20 Diciembre].

[W2] Instalación Web Centos 6.2 (2012, Septiembre). [Online]. Disponible en: http://jogagiro.wordpress.com/2012/09/13/instalacion-web-centos-6-2/ [2012, 24 Diciembre].

[W3] Técnicas de sondeo de puertos. [Online]. Disponible en: http://nmap.org/man/es/man-port-scanning-techniques.html [2014, 10 Enero]

[W4] Information Gathering: Maltego (2013, Enero). [Online]. Disponible en: http://hacking-etico.com/2013/01/09/information-gathering-maltego/ [2014, 23 Marzo].

[W5] Proyecto Web Scarab OWASP (2011, Junio). [Online]. Disponible en: https://www.owasp.org/index.php/Proyecto_WebScarab_OWASP [2014, 15 Abril].

[W6] Jason A. Donenfield (2007). Linux Local Privilege Escalation via SUID /PROC/PID/MEM WRITE. [Online]. Disponible en: http://blog.zx2c4.com/749 [2014, 15 Abril].

[W7] Alvaro Paz (2009, Noviembre). Herramienta de OS Fingerprinting. [Online]. Disponible en: http://www.gurudelainformatica.es/2009/11/ herramienta-de-os-fingerprinting.html [2014, 5 Abril].

[W8] BlitzKrieg (2007, Agosto). El sendero del Hacker “Ataques de Denegación de Servicio”. [Online]. Disponible en: http: //issuu.com/griillo7/docs/el-sendero-del-hacker, pp. 138-139 [2014, 10 Abril].

[W9] Arriols Eduardo (2013, Octubre). MITM, DNS Spoofing y Set. [Online]. Disponible en: http://highsec.es/wp-content/uploads/2013/10/

Mitm-Dns-Spoof-y-Set.pdf [2014, 10 Marzo].

[W10] John The Ripper – Cracking Passwords (2013, Julio). [Online]. Disponible en: http://freak4zoid.blogspot.com/2013/07/john-ripper-optimizacion.html [2014, 20 Abril].

ANEXOS

ANEXO N.1 Configuración del archivo “iptables.conf” del Servidor Firewall Iptables

ANEXO N.2: Configuración del archivo “squid.conf” del Servidor control de contenidos Squid Proxy

ANEXO N.3: Instalación y Configuración del Servicio Dns

ANEXO N.4: Instalación y Configuración del Servidor Web

ANEXO N.5: Instalación y Configuración del Servidor Ftp

ANEXO N.6: Puertos UDP comunes y contemporáneos en relación a malware de acceso remoto

ANEXO N.7: Operadores Básicos de Google Hacking (GH)

ANEXO N.8: Parches de actualización para el sistema operativo Centos 6.2

ANEXO N.9: Glosario de Términos

ANEXO N.1: CONFIGURACIÓN DEL ARCHIVO “iptables.conf” del SERVIDOR FIREWALL IPTABLES

/etc/sysconfig/iptables

```
# Firewall configuration written by lokkit

# Manual customization of this file is not recommended.

# Note: ifup-post will punch the current name servers through the
#       firewall; such entries will *not* be listed here.

*filter

:FORWARD ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:FILTER_FIREWALL - [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -j FILTER_FIREWALL
-A FORWARD -j FILTER_FIREWALL
-A FILTER_FIREWALL -m state --state ESTABLISHED,RELATED -j ACCEPT
-A FILTER_FIREWALL -p tcp -m tcp -m multiport -j ACCEPT –dports 25, 20:21, 80, 110, 22, 11000, 9090, 53  --syn
-A FILTER_FIREWALL -m udp -p udp --dport 53 -j ACCEPT
-A FILTER_FIREWALL -i lo -j ACCEPT
-A FILTER_FIREWALL -i eth1 -j ACCEPT
-A FILTER_FIREWALL -i eth2 -j ACCEPT
-A FILTER_FIREWALL -p tcp -m tcp -j REJECT  --syn
-A FILTER_FIREWALL -p udp -m udp -j REJECT
```

```
COMMIT
# Generated by webmin

*mangle
:FORWARD ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:PREROUTING ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]

COMMIT
# Completed
# Generated by webmin

*nat
:OUTPUT ACCEPT [0:0]
:PREROUTING ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]

#Cuando llegue las peticiones al puerto 80 lo redireccione al puerto 3128
-A PREROUTING -p tcp -m tcp -i eth2 --dport 80 -j REDIRECT --to-ports 3128

#Enmascara los puertos 22, 110, 20:21, 443, 11000, 53 para salida al internet.
-A POSTROUTING -p tcp -m tcp -m multiport -j MASQUERADE --dports 22, 110,
20:21, 443, 11000, 53
#Enmascara el protocolo icmp cuando el destino sea la interfaz eth0
-A POSTROUTING -p icmp -o eth0 -j MASQUERADE
```

ANEXO 2: CONFIGURACIÓN DEL ARCHIVO “Squid.conf” DEL SERVIDOR CONTROL DE CONTENIDOS SQUID PROXY

/etc/squid/squid.conf

```
#
# Recommended minimum configuration:
#
acl manager proto cache_object
acl localhost src 127.0.0.1/32 ::1
aclto_localhostdst 127.0.0.0/8 0.0.0.0/32 ::1

# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
#acllocalnetsrc 10.0.0.0/8  # RFC1918 possible internal network
acllocalnetsrc 172.16.9.0/24 # RFC1918 possible internal network
acllocalnetsrc 172.16.10.0/24
#acllocalnetsrc 192.168.0.0/16     # RFC1918 possible internal network
#acllocalnetsrc fc00::/7       # RFC 4193 local private network range
#acllocalnetsrc fe80::/10      # RFC 4291 link-local (directly plugged) machines

aclSSL_ports port 443
aclSafe_ports port 80 # http
aclSafe_ports port 11000
aclSafe_ports port 21 # ftp
aclSafe_ports port 443 # https
aclSafe_ports port 70 # gopher
aclSafe_ports port 210 # wais
aclSafe_ports port 1025-65535 # unregistered ports
```

```
aclSafe_ports port 280 # http-mgmt
aclSafe_ports port 488 # gss-http
aclSafe_ports port 591 # filemaker
aclSafe_ports port 777 # multiling http
acl CONNECT method CONNECT
aclRedes_Socialesurl_regex "/etc/squid/Redes_Sociales"
aclpermitidosrc 172.16.9.2 172.16.10.1-172.16.10.254

# Recommended minimum Access Permission configuration
# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access deny manager

# Deny requests to certain unsafe ports
http_access deny !Safe_ports

# Deny CONNECT to other than secure SSL ports
http_access deny CONNECT !SSL_ports

# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
```

```
# from where browsing should be allowed
http_access allow localnet

# And finally deny all other access to this proxy
http_access allow localhost
http_access deny all

# Squid normally listens to port 3128
http_port 3128 transparent

# We recommend you to use at least the following line.
hierarchy_stoplistcgi-bin ?

# Uncomment and adjust the following to add a disk cache directory.
cache_dirufs /var/spool/squid 100 16 256

# Leave coredumps in the first cache dir
coredump_dir /var/spool/squid

# Add any of your own refresh_pattern entries above these.
refresh_pattern ^ftp:                1440  20%   10080
refresh_pattern ^gopher:   1440  0%    1440
refresh_pattern -i (/cgi-bin/|\?) 0  0%    0
refresh_pattern .            0       20%   4320
cache_mem 50 MB
```

ANEXO N. 3: INSTALACIÓN Y CONFIGURACIÓN DEL SERVICIO DNS

1. Instalar el servicio dns para lo cual se hace uso del paquete bind

```
[root@dns1 sysconfig]# yum install bind
```

2. Confirmar que el servicio se halla instalado correctamente

```
[root@dns1 sysconfig]# rpm -q bind
bind-9.8.2-0.17.rc1.el6_4.4.i686
```

3. Configurar el archivo principal del servicio DNS localizado en el directorio /var/named/chroot/etc/named.conf con los siguientes parámetros:

a) Escuchar el puerto 53 para el servicio DNS y configurar los parámetros citados a continuación:

```
options {
listen-on port 53 { 172.16.9.2; };
directory       "/var/named/";
dump-file       "/var/named/data/cache_dump.db";
statistics-file "/var/named/data/named_stats.txt";
allow-query     { 172.16.9.0/24; 172.16.10.0/24; localhost; };
allow-recursion {
127.0.0.1;
172.16.9.0/24;
172.16.10.0/24;
};
forwarders {
200.105.225.2;
200.105.225.4;
};
};
```

b) Definir la zona directa que permite resolver nombres de dominios a direcciones IP. El nombre del dominio es iza.sanhueza.ec y la dirección IP estática asignada al

servidor es 172.16.9.2. El directorio de la zona directa es: /var/named/chroot/var/named/iza.sanhueza.ec.zone

En este apartado también se define el permitir la transferencia de zona solo al host 192.168.1.1 que corresponde al server primario dns.

```
zone "iza.sanhueza.ec" IN {
        type master;
        file "iza.sanhueza.ec.zone";
        allow-transfer { 192.168.1.1; localhost; };
        allow-query { any; };
};
```

c) Definir la zona inversa que permite resolver direcciones IP a nombres de dominios. El directorio de la zona inversa es: var/named/chroot/var/named/9.16.172.in-addr.arpa.zone

```
zone "9.16.172.in-addr.arpa" IN {
        type master;
        file "9.16.172.in-addr.arpa.zone";
        allow-query { any; };
};
```

d) Editar el archivo de la zona directa iza.sanhueza.ec.zone de la siguiente forma:

```
$TTL 1D
@       IN      SOA     dns1.iza.sanhueza.ec. admin.iza.sanhueza.ec. (
                                        10      ;serial
                                        1D      ;refresh
                                        1H      ;retry
                                        1W      ;expire
                                        3H )    ;minimun

        IN      NS      dns1.iza.sanhueza.ec.
dns1    IN      A       172.16.9.2
www     IN      CNAME   dns1
ftp     IN      CNAME   dns1
```

e) Editar el archivo de la zona inversa .16.172.in-addr.arpa.zone de la siguiente forma:

```
$TTL 3H
@       IN      SOA     dns1.iza.sanhueza.ec. root.iza.sanhueza.ec. (
                                                        1       ;serial
                                                        1D      ;refresh
                                                        1H      ;retry
                                                        1W      ;expire
                                                        3H )    ;minimum

@       IN      NS      dns1.iza.sanhueza.ec.
2       IN      PTR     dns1.iza.sanhueza.ec.
2       IN      PTR     www.iza.sanhueza.ec.
2       IN      PTR     ftp.iza.sanhueza.ec.
```

f) Editar el archivo /etc/resolv.conf para definir el dominio DNS y la dirección IP del DNS.

```
[root@dns1 named]# vim /etc/resolv.conf
```

```
# Generated by NetworkManager
search iza.sanhueza.ec
nameserver 172.16.9.2
```

4. Reiniciar el servicio named

```
[root@dns1 named]# service named restart
Deteniendo named:                                          [  OK  ]
Iniciando named:                                           [  OK  ]
```

5. Verificar que el iptables de Linux permita el tráfico del puerto 53 para DNS, tanto en TCP como UDP.

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 53 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 53 -j ACCEPT
```

6. Se verifica la zona directa con el comando nslookup www.iza.sanhueza.ec

```
[root@dns1 named]# nslookup www.iza.sanhueza.ec
Server:         172.16.9.2
Address:        172.16.9.2#53

www.iza.sanhueza.ec     canonical name = dns1.iza.sanhueza.ec.
Name:   dns1.iza.sanhueza.ec
Address: 172.16.9.2
```

1. **Se verifica la zona inversa con el comando nslookup172.16.9.2**

```
[root@dns1 named]# nslookup 172.16.9.2
Server:         172.16.9.2
Address:        172.16.9.2#53

2.9.16.172.in-addr.arpa name = www.iza.sanhueza.ec.
2.9.16.172.in-addr.arpa name = ftp.iza.sanhueza.ec.
2.9.16.172.in-addr.arpa name = dns1.iza.sanhueza.ec.
```

ANEXO N. 4: INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR WEB

1. Instalación del servicio httpd, con el comando yum install httpd

```
[root@dns1 ~]# yum install httpd
Loaded plugins: fastestmirror, refresh-packagekit, security
Loading mirror speeds from cached hostfile
 * base: centos.secrel.com.br
 * extras: centos.secrel.com.br
 * updates: centos.secrel.com.br
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package httpd.i686 0:2.2.15-15.el6.centos.1 will be updated
--> Processing Dependency: httpd = 2.2.15-15.el6.centos.1 for package: httpd-man
ual-2.2.15-15.el6.centos.1.noarch
--> Processing Dependency: httpd = 2.2.15-15.el6.centos.1 for package: 1:mod_ssl
-2.2.15-15.el6.centos.1.i686
---> Package httpd.i686 0:2.2.15-28.el6.centos will be an update
--> Processing Dependency: httpd-tools = 2.2.15-28.el6.centos for package: httpd
-2.2.15-28.el6.centos.i686
--> Running transaction check
---> Package httpd-manual.noarch 0:2.2.15-15.el6.centos.1 will be updated
```

2. Verificar que el servicio se haya instalado correctamente con el comando rpm –q httpd

```
[root@dns1 ~]# rpm -q httpd
httpd-2.2.15-28.el6.centos.i686
```

3. Configurar el archivo httpd.conf localizado en /etc/httpd/conf/httpd.conf.

Modificar las siguientes líneas:

```
Listen 80
```

```
ServerName www.iza.sanhueza.ec:80
```

4. El servicio Web identificará el archivo del diseño Web en la ruta /var/www/html/index.html

```
<html>
<head>
<title>Mi pagina Web: Prueba</title>
</head>
<body>
<h1>pagina en construccion<br/>
regrese pronto</h1>
</body>
</html>
```

5. Reiniciar el servicio httpd con el comando servicehttpdrestart

```
root@dns1 conf]# service httpd restart
arando httpd:                                              [  OK  ]
niciando httpd:                                            [  OK  ]
```

6. Configurar el nombre del servidor y la dirección ip estática asignada al mismo. Se lo configura en el archivo /etc/hosts. El nombre de la máquina corresponde a dns1.iza.sanhueza.ec y la ip correspondiente es la 172.16.9.2

```
127.0.0.1   localhost.localdomain localhost
172.16.9.2      dns1.iza.sanhueza.ec    dns1    dns1
::1     localhost.localdomain   localhost6      localhost
~
```

7. Para desplegar la página Web, se digita en el explorador http://172.16.9.2

8. Con el comando chkconfig nos aseguramos que el servicio httpd se active siempre al reiniciar el servidor

```
root@dns1 ~]# chkconfig httpd on
```

ANEXO N. 5: INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR FTP

1. Instalar el servicio vsftpd

```
[root@dns1 ~]# yum install vsftpd
```

2. Verificar que el servicio vsftpd haya sido intalado correctamente.

```
[root@dns1 ~]# rpm -q vsftpd
vsftpd-2.2.2-11.el6_4.1.i686
```

3. Creamos un directorio en el cual se almacenarán los archivos para compartir por el servicio ftp.

```
[root@dns1 shirla1]# mkdir /var/ftp/directorioejemplo
```

4. Creamos un archivo de ejemplo dentro del directorio a compartir por ftp.

```
[root@dns1 directorioejemplo]# cd /var/ftp/directorioejemplo/
[root@dns1 directorioejemplo]# touch ejemplo
```

5. Configurar el archivo principal para el servicio ftp, el cual se encuentra en el directorio /etc/vsftpd/vsftpd.conf

```
[root@dns1 ~]# vim /etc/vsftpd/vsftpd.conf
```

6. El archivo vsftpd.conf debería presentar la siguiente configuración:

```
anonymous_enable=YES
local_enable=YES
write_enable=YES
local_umask=022
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
xferlog_std_format=YES
chroot_local_user=YES
listen=YES

pam_service_name=vsftpd
userlist_enable=YES
tcp_wrappers=YES
```

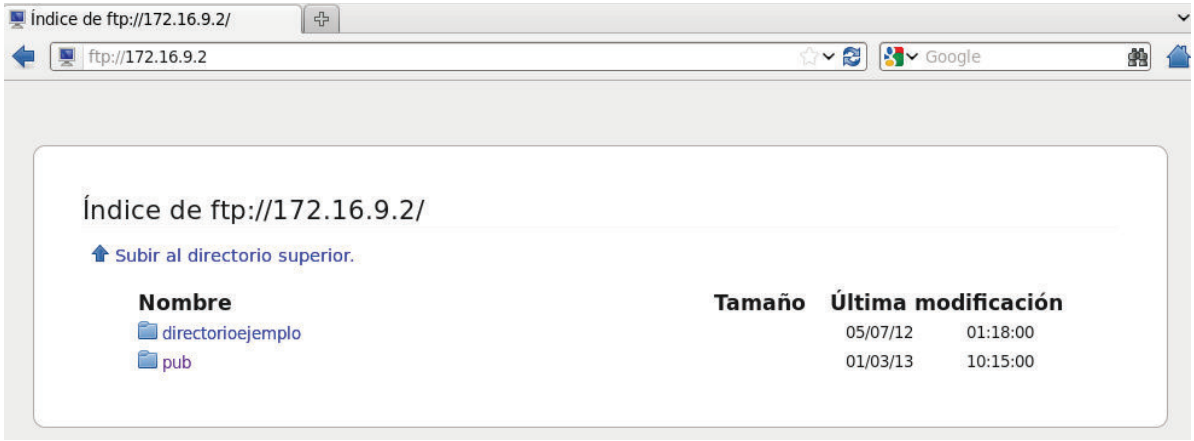
7. Reiniciamos el servicio vsftpd con el siguiente comando:

```
[root@dns1 ~]# service vsftpd restart
Apagando vsftpd:                                           [  OK  ]
Iniciando vsftpd para vsftpd:                              [  OK  ]
```

8. Con el comando chkconfig nos aseguramos que el servicio vsftpd se active siempre al reiniciar el servidor

```
[root@dns1 ~]# chkconfig vsftpd on
```

9. Para verificar el funcionamiento del servicio, escribimos en el navegador lo siguiente: ftp://172.16.9.2 y se despliega la siguiente pantalla.



10. Para incrementar la seguridad a nivel de los usuarios que pueden ingresar al servidor se define un usuario para que por medio de una contraseña específica tenga acceso correcto al servidor ftp. Para ello nos dirigimos al directorio cd /etc/vsftpd

```
[root@dns1 ftp]# cd /etc/vsftpd
```

11. Creamos el usuario con el comando user adds hirla y le asignamos una contraseña.

```
[root@dns1 vsftpd]# useradd shirla
```

12. Deshabilitar el firewall propio de Linux SELinux, para ello editamos el archivo anaconda-ks.cfg y verificar que esté activa la línea selinux –disabled.

```
[root@dns1 ~]# vim anaconda-ks.cfg
selinux --disabled
```

13. Habilitar el servicio ftp a nivel del firewall iptables en el servidor Linux, para ello debe permitir el tráfico de los puertos 20 y 21.

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 53 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 53 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 20 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 30300:30309 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

14. Reiniciamos el iptables del servidor

```
[root@dns1 sysconfig]# service iptables restart
iptables: Guardando las reglas del cortafuegos:            [  OK  ]
iptables: Poniendo las cadenas de la política ACCEPT: filte[  OK  ]
iptables: Descargando módulos:                             [  OK  ]
iptables: Aplicando reglas del cortafuegos:                [  OK  ]
iptables: Cargando módulos adicionales:nf_conntrack_ftp    [  OK  ]
```

15. Verificamos el acceso al servidor ftp con el usuario creado shirla.

ANEXO N. 6: PUERTOS UDP COMUNES Y CONTEMPORÁNEOS EN RELACIÓN A MALWARE DE ACCESO REMOTO

Número de Puerto	Tipo	Servicio
121	tcp/udp	Erpc, llamada de procedimiento remoto EncoreExpedited
193	tcp/udp	Spider Remote Monitoring Protocol SNMP
384	tcp/udp	ARNS Sistema de servidor de red remota
556	tcp/udp	Servidor RFS Remote File Sharing
623	udp	Alert Standard Format ASF RMCP Protocolo de administración y Control remoto
664	udp	Alert Standard Format ASF Secure RMCP Protocolo de administración y Control remoto
1046	tcp/udp	WebFilterRemote Monitor
1053	tcp/udp	Remote Assistant (RA)
9418	tcp/udp	Xcode Server (git remoto)

Puertos UDP conocidos en IANA

Número de Puerto	Tipo	Servicio
1701	Tcp/Udp	Acceso Remoto para VPN con L2TP
111	Tcp/Udp	Protocolo de Llamadas de procedimientos remotos (RPC) para la ejecución de comandos remotos, utilizado por Sistemas de archivos de red (Network Filesystem, NFS)
5	Tcp/Udp	Entrada de trabajo remota
50	Tcp/Udp	Protocolo de verificación de correo remoto
71, 72, 73	Tcp/Udp	Servicio de trabajos remotos

107	Tcp/Udp	telnet remoto rtelnet
517-518	Udp	Servicio y cliente de llamada remota Talk y de Red
556	Tcp/Udp	Remote FS Sistema de archivos Brunhoff (RFS)
4321	Tcp/Udp	ServicioWhoisremoto (rwhois)
7009	Tcp/Udp	Servicio gestor de caché remoto AFS
2105	Tcp/Udp	Inicio de sesión remota encriptada Kerberos v5 (rlogin)
953	Tcp/Udp	Herramienta de configuración remota del Berkeley Internet NameDomain versión 9 (BIND 9) rndc
514	Tcp/Udp	RSH RemoteSHell, es un servicio que permite ejecutar comandos de otros usuarios a través de Internet. Ataque malware RPC Backdoor
1099	Tcp/Udp	RMI Java RemoteMethodInvocation, es el equivalente de los RPC pero en Java. Ataque malware BFevolution, Rat (TCP)
2773	Udp	REMOTE BACKUP Servicio que permite hacer respaldos remotos via red de información BackDoor-G, SubSeven, Sub7

Puertos UDP conocidos ante malware en Sistemas Unix/Linux

Número de Puerto	Tipo	Servicio
137-138	Udp	Netbios de Windows
1900	Udp	Servicios de descubrimientos SSDP

Puertos UDP conocidos ante malware en Sistemas Windows

ANEXO N. 7: OPERADORES BÁSICOS DE GOOGLE HACKING (GH)

Operadores Básicos		
Símbolo	**Tipo**	**Nombre y función**
|	OR	Búsqueda de cualquiera de los términos de la consulta
+	AND	Incluye más de 1 palabra clave
-	NOT	Elimina los resultados que contengan el término que sigue al operador
*	Comodín	Añade cualquier término entre dos palabras de búsqueda
" "	Búsqueda exacta	Busca frases o expresiones exactas

Tabla 2.4.3.2.1 Operadores básicos con GH

Operadores de Búsqueda Avanzados	
Operador	**Propósito**
intitle	Buscapáginas con ese título.
Allintitle	Buscapáginas que solamente tenga ese título en específico.
Inurl	Busca direcciones en Internet que tengan ese URL.
Allinurl	Busca direcciones en Internet solamente las páginas que tengan ese URL.
Filetype	Especifica un tipo de extensión de archivo.
Allintext	Busca solo ese texto en específico.
Site	Busca una específicapágina en Internet.
Link	Busca las paginas relacionadas con esta frase.

inanchor	Busca el texto anchor de la página relacionada.
daterange	Busca un rango de datos.
Autor	Busca un grupo de autores.
Group	Busca el nombre de algún grupo.
insubject	Busca un tema en particular.
Msgid	Busca el grupo msgid.

Tabla 2.4.3.2.2 Operadores de búsqueda avanzados con GH

Palabras Clave (Keywords)			
Ingles		**Ingles**	
dictionary	**diccionario**	**cheatsheet**	**apuntes de repaso**
manual	manual	template	plantilla
template	formulario	rubric	rubrica
glossary	glosario	tutorial	tutorial
course	curso	introduction	introducción
poster	afiche	board	cartel
how to	como hacer	studyguide	guía de estudio
internship	internado	jobs	trabajos
university	universidad	letter	carta

Tabla 2.4.3.2.3 Palabras clave con GH

Ejemplos de palabras clave en combinación con operadores	
Comandos de Google	**Resultados Aproximados**
Como hacer	44,200,000
"Como hacer" site:www.youtube.com	46,400,000
"Como hacer" filetype:pdf	1,260,000
"Como hacer" inurl:edu	500
"Como hacer" inurl:edufiletype:pdf	363
"Como hacer" filetype:doc	49,6
"Como hacer" inurl:edufiletype:doc	40,5
"Como hacer" filetype:html	5,120,000
"Como hacer" -filetype:html	68,800,000
"Como hacer" inurl:edu -filetype:html	212

Tabla 2.4.3.2.4 Ejemplos de palabras clave con GH

ANEXO N.8: PARCHES DE ACTUALIZACIÓN PARA EL SISTEMA OPERATIVO CENTOS 6.2[88]

- http://mirror.centos.org/centos/6/updates/i386/Packages/glibc-2.12-1.149.el6_6.4.i686.rpm
- http://mirror.centos.org/centos/6/os/i386/Packages/libgcc-4.4.7-11.el6.i686.rpm
- http://mirror.centos.org/centos/6/updates/i386/Packages/nss-softokn-freebl-3.14.3-18.el6_6.i686.rpm
- http://mirror.centos.org/centos/6/updates/i386/Packages/tzdata-2014h-1.el6.noarch.rpm
- http://mirror.centos.org/centos/6/updates/i386/Packages/glibc-common-2.12-1.149.el6_6.4.i686.rpm

[88] Parches para Vulnerabilidad Ghost de sistemas Centos6 i386 desde Enero 2015. Fuente: http://blog.itlinux.cl/blog/2015/01/30/ghost-nueva-vulnerabilidad-critica-de-linux/.

ANEXO N.9: GLOSARIO DE TÉRMINOS

Testigo: término escogido para la traducción: testigo. Puede ser interpretado como identificador, elemento.

Framework: término escogido para la traducción: entorno de trabajo. En ocasiones se ha mantenido el término original, cuando forma parte del nombre de un producto. Puede ser interpretado como entorno de desarrollo, entorno de pruebas, marco de pruebas.

Backend: se ha optado por no traducir el término. Puede ser interpretado como servicios internos, de infraestructura.

Auditor: término escogido para la traducción: persona que realiza las pruebas, o auditor. Puede ser interpretado como verificador, comprobador.

Housing: Es una modalidad de alojamiento web destinado principalmente a grandes empresas y a empresas de servicios web.

Survey es un término aplicado a una encuesta o recopilación de información a base de preguntas que se formulan a un conjunto o muestra de personas previamente escogidas según las necesidades del estudio.

Gusano es un programa independiente que se replica a través de una red.

Troyano es un código malicioso que, a diferencia de los virus y gusanos, no puede reproducirse por sí mismo e infectar archivos.

Consultas recursivas Dns: Establece el reenvío de solicitudes Dns a una zona autoritativa no conocida, es decir a la zona Dns del ISP.

ICANN (Corporación de Internet para la Asignación de Nombres y Números) es una organización que opera a nivel multinacional/internacional, responsable de asignar las direcciones del protocolo IP, los identificadores de protocolo, las

funciones de gestión del sistema de dominio y la administración del sistema de servidores raíz.

IANA Internet Assigned Numbers Authority es la entidad que supervisa la asignación global de direcciones IP, sistemas autónomos, servidores raíz de nombres de dominio DNS y otros recursos relativos a los protocolos de Internet.

Zombie es una computadora que tras haber sido infectado por algún tipo de malware, puede ser usado por una tercera persona para ejecutar actividades hostiles. Este uso se produce sin la autorización o el conocimiento del usuario del equipo.

Firewall (con estado): el dispositivo de filtro de paquetes con estado que mantienen las tablas de los estados de los protocolos basados en IP. El tráfico sólo puede atravesar el firewall si se ajusta a los filtros de control de acceso definidos o si forma parte de una sesión ya establecida de la tabla de estados.