Universidad Autónoma del Estado de México

Centro Universitario UAEM Zumpango

Seguridad en servidores DNS

TESIS

para obtener el título de

Ingeniero en Computación

presenta:

Elihu Hassani Hamud Guzman

Asesor

Dra. María de Lourdes López García

Zumpango, Estado de México Enero, 2014

Agradecimientos

Agradecimientos personales

Principalmente quiero agradecer en estas líneas la ayuda brindada por parte de mis abuelos ***María del Carmen Guzmán Prado*** y ***José Rubén Francisco Hamud Rivera***, ya han sido un ejemplo a seguir en todo momento y han compartido conmigo sus enseñanzas y experiencias para poder salir adelante, les agradezco por todo su apoyo en todos los aspectos como el moral y el económico, ya que sin ellos no estaría finalizando esta carrera.

Agradecimientos institucionales

Para comenzar y finalizar un trabajo tan arduo como el desarrollo de una tesis, es necesario obtener una gran fuente de apoyo y guía, por eso es para mí un verdadero placer utilizar este espacio para expresar mis agradecimientos principalmente a la ***Dra. María de Lourdes López García***, por haberme permitido desarrollar este trabajo bajo su dirección, brindándome siempre su orientación con profesionalismo ético en la adquisición de los conocimientos.

Por otro lado también quiero expresar mis más sinceros agradecimientos al ***Ing. Jorge Bautista López*** y al ***Dr. Arturo Redondo Galván*** por haberme brindado su apoyo incondicional para el buen desarrollo de este tema de tesis, compartiendo conmigo sus opiniones y observaciones.

En general quiero aprovechar para agradecer a cada uno de los ***docentes*** que me han acompañado durante el largo camino, ya que ellos me proporcionaron los medios suficientes para llevar a cabo todas las actividades propuestas durante el desarrollo de la carrera.

Resumen

En la actualidad, la comunicación a nivel mundial ha crecido a gran escala gracias a la red de redes "Internet", que se basa principalmente en la utilización de servidores que sirven como fuente de intercambio constante de información. Tal funcionalidad resulta por demás compleja debido a los diversos elementos que interactúan para realizar la comunicación en tiempos muy cortos.

Un caso en particular son los Servidores de Nombres de Dominio (DNS), que permiten la interconexión a alguna red existente local o en Internet. Sin embargo, los usuarios utilizan los servidores DNS sin tener muy claro cómo funcionan y los problemas a los que son vulnerables. Esto implica que los usuarios se encuentren expuestos a ataques provocados por usuarios maliciosos.

En general, los servidores DNS son de los más importantes, ya que permiten obtener un resultado sobre la petición de cierta página y su correcto funcionamiento debe ser prioritario, porque no se sabe que tanto nivel de afectación se puede producir ante la ejecución de un ataque, dejando sin servicio a los clientes que quieren obtener alguna información, es por eso que se deben de tomar medidas preventivas que brinden una estabilidad dejando a un lado la afectación que se pudiera generar con ciertos ataques.

Por lo anterior, este documento tiene como objetivo presentar la seguridad existente en los servidores DNS, analizando diversas técnicas de ataque que desestabilizan su funcionalidad e implementando diversas técnicas de ataque que se consideran las más efectivas y eficientes. Además, se proponen e implementan las soluciones a las técnicas de ataques mencionadas, reduciendo con esto la efectividad de los mismos.

Abstract

Nowadays, global communication has grown on a large scale through the Internet, which is based primarily on the use of servers that interchange information, constantly. Thus, the functionality is complicated due to the elements that interact among them to make communication in very short times. A particular case is the Domain Name Servers (DNS), which allows interconnection to any existing local network or the Internet. However, users use DNS servers without having very clear how they work and the problems they are vulnerable. This means that users are exposed to attacks caused by malicious users. In general, the DNS servers are important, in the sense that they allow to obtain a result on the petition in a web page and its correct operation must be a priority, because it is known that both level of affectation may occur before the execution an attack, knocking out service to clients who want to get some information, that is why you should take preventive measures that provide stability aside the affectation that might be generated after the attacks, providing an efficient and fiable results. Therefore, this paper aims to present existing security DNS servers, analyzing various attack techniques that disrupt its functionality and implementing some of the techniques of attack that are considered the most effective and efficient. Furthermore, we propose and implement solutions to the mentioned attack techniques, thereby reducing the percentage of their effectiveness.

Contenido

Lista de Tablas

Lista de Figuras

Capítulo 1

INTRODUCCIÓN

En la actualidad, se utiliza en gran escala la Internet como herramienta de comunicación. La red de redes o Internet, se basa principalmente en la utilización de servidores que sirven como fuente de intercambio constante de información, función que resulta por demás compleja debido a los diversos elementos que interactúan para su adecuado funcionamiento.

El manejo de información por medio de la Internet se ha convertido en una herramienta muy útil, donde el resguardo y la seguridad son una de las principales prioridades.

Un caso en particular es el uso de los DNS (Servidores de Nombres de Dominio), donde los usuarios lo utilizan sin tener muy claro su funcionalidad para acceder a la Internet.

1.1. Planteamiento y definición del problema

La Internet evoluciona constantemente, es funcional para diversos propósitos u objetivos. No obstante estos cambios a veces repercuten de manera negativa en el servicio de acceso a la red, como por ejemplo generando colapsos en los recursos de algún servidor hasta agotarlos y en consecuencia disminuir el ancho de banda. El avance de las nuevas tecnologías, constituye un desafío para los usuarios en la actualidad, ya que demandan

un servicio por excelencia, por otro lado en el ámbito empresarial, la exigencia es primordial, debido a que pueden estar expuestos a los mismos ataques por los que pasa un usuario, generando posiblemente que la conectividad que brinda dicha compañía se vea afectada parcialmente o en su totalidad.

1.2. Hipótesis

En los servidores DNS se puede suplantar un portal de la Internet, provocando que el usuario ingrese a una dirección de Internet en el navegador y visualice una página diferente. Además este tipo de servidores puede saturarse con múltiples solicitudes, generando un retraso en la búsqueda de la información requerida.

Existen diferentes métodos para prevenir este tipo de ataques, en donde los usuarios se beneficien del uso adecuado de la información relacionada con los DNS.

1.3. Justificación

Actualmente en la Internet existen diversos tipos de vulnerabilidades en los Sistemas de Nombres de Dominio, manifestados tanto en las redes locales y no locales, el efecto del daño puede ser provocando a gran o menor escala, debilitando por un lado el buen servicio que se debe de proporcionar a los usuarios finales, esto se podría convertir en un problema muy relevante, ya que en estos tiempos el uso e intercambio de información a través de la red se da con gran frecuencia, logrando principalmente que los recursos de la red no se utilicen en su máxima totalidad. Es por eso que en este documento se presentará un análisis a detalle de los diferentes métodos que existen para evitar la diversidad de vulnerabilidades en los DNS, obteniendo como beneficio una estabilidad en la navegación, como el acceso sin retrasos de las páginas que se visiten y utilizar el ancho de banda que se contrato que por derecho le corresponde al usuario.

1.4. Objetivos

1.4.1. Objetivo General

Determinar la seguridad existente en los servidores DNS, mediante la revisión de información de distintas fuentes para señalar sus ventajas, desventajas y los tipos de ataques a los que es vulnerable, además de revisar métodos eficaces para evitarlos.

1.4.2. Objetivos Específicos

- Investigar las funciones del servidor DNS.
- Describir las ventajas y desventajas de los servidores DNS.
- Identificar los posibles ataques a los servidores DNS.
- Implementar los ataques identificados.
- Proponer e implementar métodos eficaces para evitar los ataques y brindar mayor seguridad a los servidores DNS.

1.5. Alcance del proyecto

Se analizarán diferentes métodos de vulnerabilidad para el servidor DNS, con la finalidad de encontrar algún mecanismo de seguridad, que ayude a reducir los tipos de vulnerabilidades que existen.

1.6. Organización del documento

La presente Tesis está estructurada de la siguiente manera, en el capítulo 2 se presenta un preámbulo de lo que son las redes, mencionando sus origenes, los tipos, su aplicación, además de un análisis de los modelos OSI y TCP/IP.

En el capítulo 3 se menciona las funciones de un DNS a través de los nombres de dominio y la clasificación de estos, así como las ventajas y desventajas que se producen al utilizar este tipo de servidores.

El capítulo 4 entra en materia acerca de los tipos de ataques a los que se exponen los DNS, mencionando en que consisten cada uno y mostrando la ejecución de estos, con el resultado obtenido.

En el capítulo 5 se presentan las técnicas de prevención ante ataques a los DNS, exponiendo a detalle los requisitos que se deben de tener para seguir una metodología de solución.

Finalmente el capítulo 6 contiene las conclusiones, dando a conocer los resultados obtenidos y mencionando como puede ayudar todo este conocimiento adquirido para la prevención de ciertos ataques.

Capítulo 2

Aspectos generales de una red

A lo largo de este capítulo se describen los conceptos básicos de los tipos de redes, los elementos que las conforman, las aplicaciones, entre otras definiciones esenciales para comprender un poco más acerca de las redes.

Las redes son un mecanismo eficaz que permiten ahorrar tiempo ya sea para la búsqueda de información que se requiere, o para enviar algún mensaje a zonas muy distantes, es por eso que su estudio y comprensión es fundamental en este documento.

2.1. Redes

El concepto de redes es utilizado desde hace mucho tiempo. Antes de la existencia de la tecnología en las computadoras y la telefonía, la forma de comunicarse era algo complicada, ya que los mensajes eran enviados por medio de palomas mensajeras, sonido, señales de fuego, etcétera. Con el tiempo apareció el uso del telégrafo, donde su funcionamiento radicaba en la utilización de señales eléctricas.

Actualmente ya no se usan como medio de comunicación aquellos sistemas primitivos, debido a la aparición de las computadoras ya que pueden comunicarse entre sí, interconectadas por medio de cables, fibra óptica o microondas.

La finalidad de todo sistema de comunicación es intercambiar información, en la figura 2.1 se muestra el modelo para las comunicaciones que presenta dos entidades: una fuente y un destino, dos componentes: un transmisor y un receptor, y un sistema

de trasmisión, definidos a continuación [24]:

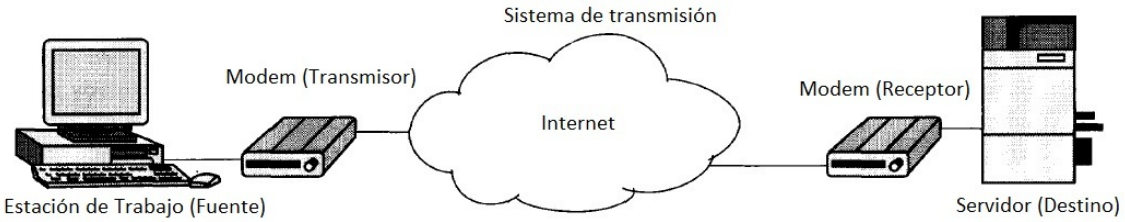

Figura 2.1: Modelo para las comunicaciones

1. Fuente:

   Dispositivo, ya sea teléfono o computadora, que genera los datos a transmitir y envíar al transmisor.

2. Transmisor:

   Dispositivo que modula y codifica la información, generando señales electromagnéticas, para poder ser enviadas a un sistema de transmisión. Este convierte las cadenas de bits generadas por algún dispositivo de cómputo o móvil y las transforma en señales que se adaptan al medio, es muy importante la participación de este dispositivo ya que los datos no se pueden transmitir de forma directa.

3. Sistema de transmisión:

   Medio de comunicación que permite realizar la conexión con el destino de forma cableada o inalámbrica.

4. Receptor:

   Dispositivo que recibe la señal proveniente del sistema de transmisión y la transforma de analógica a digital para poder ser utilizada en el dispositivo destino.

5. Destino:

   Dispositivo que toma los datos del receptor.

La comunicación no sería posible sin algun tipo de estandarización, en el área de las redes de computadoras a este tipo de estándares se les conoce como *protocolos*, que son códigos de comportamiento con reglas muy estrictas [18].

Por tanto, una red se puede definir como la interconexión de dos o más computadoras para compartir recursos, donde el objetivo es hacer que todos los programas, y en particular los datos, estén disponibles para todos los que se conecten a una red, independientemente de la ubicación física del recurso y del usuario [4].

Otra forma común de definir una red es como una colección de nodos, con la capacidad de realizar una comunicación entre sí, confiando en los servicios de una cantidad determinada de máquinas encargadas de transmitir datos para quien lo solicite. Los nodos por lo regular son equipos de cómputo, pero no necesariamente debe ser así, en algunas ocasiones son otro tipo de terminales, impresoras inteligentes o dispositivos móviles [18].

2.2. Tipos de redes

Las redes pueden clasificarse por dos cosas: la aplicación para lo cual son utilizadas y por su tamaño [4].

En el primer caso, se encuentran las redes modelo cliente-servidor y las redes nombradas de acuerdo al tipo de usuario como son P2P (Peer to Peer), C2C (Consumer to Consumer), B2B (Business to Business), B2C (Business to Consumer) y G2C (Government to Consumer).

Para el caso de la clasificación por el tamaño se encuentran las redes BAN (Body Area Network), PAN (Personal Area Network), LAN (Local Area Network), MAN (Metropolitan Area Network), WAN (Wide Area Network) y la red pública Internet. Ambos tipos de redes serán descritos de forma breve.

2.2.1. Redes de acuerdo al tipo de aplicación

Modelo Cliente-Servidor

Actualmente el Software desarrollado se basa en el modelo cliente-servidor. Básicamente este modelo es dividido en dos partes, una parte de este servicio se proporciona del lado del servidor y la otra parte está siendo utilizada por parte del cliente, algunas aplicaciones de esto pueden ser el acceso a una impresora, a archivos almacenados, o

a una base de datos. Este modelo también es distinguido como un sistema distribuido porque reduce las congestiones de la red, mediante una división de las tareas de procesamiento con el cliente y el servidor, el servidor normalmente es más potente que una computadora que está siendo utilizada como cliente, además en este se almacena y presenta la información. En la figura 2.2 se observa a grandes rasgos este modelo.

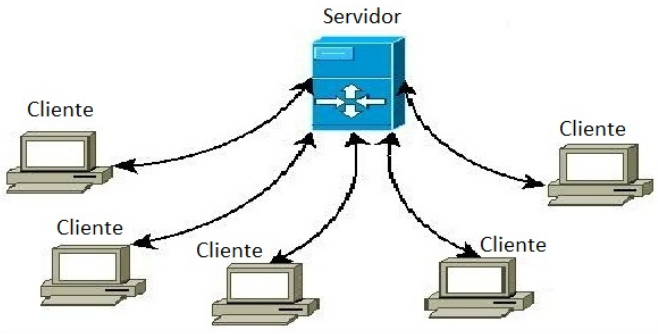

Figura 2.2: Modelo Cliente-Servidor

También se debe mencionar que no necesariamente un servidor debería poseer muchos recursos, ya que puede ser incluso una computadora con bajas capacidades, todo esto depende de la complejidad del uso que se le dará.

Otra forma de ver como utilizamos en repetidas ocasiones el modelo cliente-servidor es, cuando navegamos en la Internet, una de las aplicaciones cliente que nos permiten utilizar este modelo es sencillamente el navegador. Estos navegadores nos permiten acceder a una página Web, en donde se contacta con el servidor que aloja dicho sitio y nos despliega en pantalla la página que nosotros hemos solicitado desde el servidor, el usuario entonces podría estar accediendo a distintos enlaces y también a distintos servidores, pero no descartamos que la máquina de este usuario pueda estar como servidor para que alguien más pueda tener acceso a las páginas que el aloja.

Para realizar una sesión típica Cliente-Servidor se comprende el siguiente proceso mostrado en la figura 2.3:

1. El usuario efectúa una petición de datos.

2. La computadora cliente traduce la petición en un formato que el servidor puede

entender.

3. El cliente envía la petición al servidor.

4. El servidor procesa la petición, la cual puede involucrar comunicación con servidores remotos.

5. El servidor envía la respuesta al cliente.

6. El cliente envía la respuesta a su pantalla.

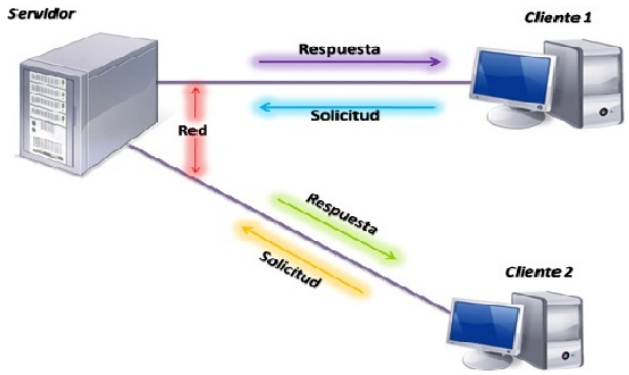

Figura 2.3: Proceso de Solicitud y respuesta del Modelo Cliente-Servidor

La ventaja de este modelo es que nos permite compartir la responsabilidad del proceso de las tareas y nos permite añadir más componentes al sistema, dejando de lado el aspecto de la limitación a sólo una única solución.

Otra ventaja que se puede mencionar es que este sistema puede trabajar eficientemente tanto en servidores de Unix y Windows, permitiendo desempeñar un trabajo en conjunto de acuerdo a las necesidades y demandas del cliente [3].

Tipos de Usuarios

Esta clasificación depende en gran manera del tipo de negocio que se este manejando, ya que el usuario es clasificado de acuerdo al grado de consumo que genere, y por lo tanto el nombre entre cada red es diferente. Acontinuación se mencionara cada una de estas [4].

1. Persona a persona P2P:

   Son redes entre pares o redes punto a punto. En estas redes no existen ni terminales clientes ni terminales que hagan la función de servidor, las redes P2P permiten el intercambio directo de información en cualquier formato entre las computadoras interconectadas.

   Estas redes aprovechan, administran y optimizan el uso del ancho de banda de los demás usuarios de la red por medio de la conectividad entre los mismos.

2. Consumidor a consumidor C2C:

   Generalmente es utilizado este término para definir un modelo de negocio en la red, que pretende acercar comercialmente al usuario final con otro usuario final. Se basa en transacciones donde actúa usualmente una empresa mediadora, que acerca la oferta y demanda de artículos o servicios, un claro ejemplo de esto puede ser www.ebay.com o también www.mercadolibre.com.

3. Negocio a consumidor B2C:

   Este tipo de red es utilizada normalmente cuando se quiere hablar de una estrategia realizada por una empresa para llegar a sus clientes o consumidores. Los beneficios que se obtienen son que evita largas colas o visitas, a distintas empresas buscando un precio, además en un clic se tiene la posibilidad de comparar productos, un ejemplo de este tipo de red es cuando se hacen pedidos de libros en línea.

4. Negocio a negocio B2B:

   En este tipo de red usualmente se puede realizar cualquier operación comercial que se genere entre una empresa y su cadena de abastecimiento o con su cadena de distribución, ya sea directamente o a través de un intermediario que esté en línea, un ejemplo de este tipo de redes es www.syscom.mx.

5. Gobierno a consumidor G2C:

   Las operaciones se realizan entre los ciudadanos y las administraciones públicas, un claro ejemplo de esto es la declaración de impuestos a través de Internet.

2.2.2. Redes de acuerdo a su tamaño

Red de área corporal (BAN: Body Area Network)

Dentro de las redes de área corporal podemos encontrar algunas tecnologías como lo son Bluetooth, ZigBee, millimeter-wave y Ultra Wide Band. Este tipo de redes son sistemas de comunicaciones de pequeña escala, las transmisiones se realizan dentro, alrededor o sobre el cuerpo humano, por lo tanto la cobertura no debe exceder distancias superiores a los 2 ó 3 metros, además debe observarse que las emisiones de energía de este tipo de tecnología son muy bajas, propiciando así larga duración a las baterías incorporadas en los dispositivos [15].

Red de área personal (PAN: Personal Area Network)

Este tipo de red está reservada para la utilización de una sóla persona, sin ir más allá de su área de alcance. En ella se interconectan dispositivos de cómputo en un área de cobertura pequeña, aproximadamente a 1 metro. Las primeras redes PAN manejaban enlaces infrarrojos para su interconexión, brindando una velocidad de 2.4kb/s hasta 16Mb/s, el detalle es que a la hora de transmitir ambos dispositivos deberían estar librados de obstáculos para evitar la interferencia, logrando así una comunicación correcta y estable, a esto se le denomina como línea de vista.

Algunos ejemplos de la PAN son una red inalámbrica entre una computadora y un ratón, teclado e impresora, incluso un PDA que controla el audífono o el marcapaso de un usuario son consideradas como redes de área personal. En la figura 2.4 se muestra como se da la interconexión de este tipo de red [4, 17].

Red de área local (LAN: Local Area Network)

Las redes de área local son de propiedad privada y son utilizadas para conectar computadoras personales a estaciones de trabajo en oficinas de algunas empresas, ya sea para intercambiar información o compartir recursos, pueden alcanzar una velocidad de transmisión desde 10 Mbps hasta 10 Gbps. En la figura 2.5 se observa como se conforma una red de este tipo.

Las LANŽs se delimitan por el tamaño, pueden ser utilizadas en hogares pequeños,

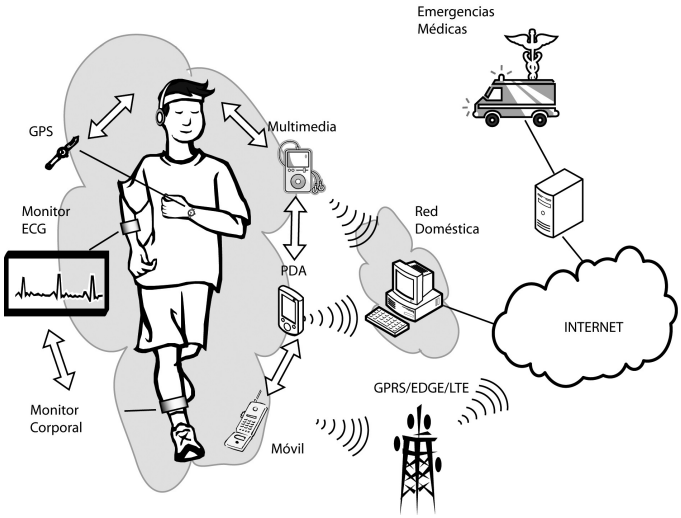

Figura 2.4: Interconexión PAN

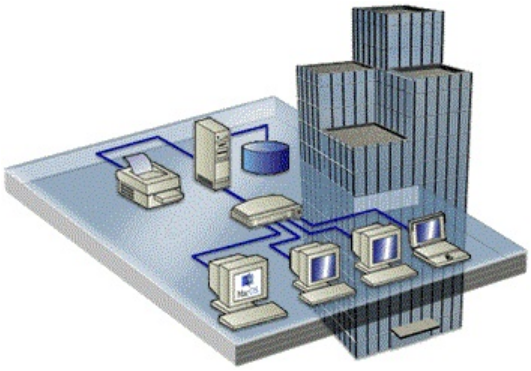

Figura 2.5: LAN: Local Area Network

edificios o en un campus de pocos kilómetros de longitud. También se clasifican de acuerdo al tipo de tecnología, utilizando diferentes estándares de comunicación como lo son: IEEE 802.2, IEEE 802.3-Ethernet, IEEE 802.3u fast Ethernet, IEEE 802.3z/802.3 ab-Gigabit Ethernet, IEEE 802.5 Token Ring, entre otros que pueden revisarse en [4, 3].

Redes de área metropolitana (MAN: Metropolitan Area Network)

Una MAN posee características similares a una LAN, pero respecto a su dimensión esta es una versión más grande, puede comprender una extensión de oficinas cercanas, o una ciudad con extensión de decenas de kilómetros, en una MAN se pueden manejar

voz, datos y video.

Un ejemplo es la red de televisión por cable, que existe actualmente en muchas ciudades, el sistema tuvo sus origenes a partir de los sistemas de antena comunitaria, que consistía en una antena grande situada en la cima de una colina para poder captar la señal de televisión, un amplificador y un cable coaxial para poderla enviar a las casas de las personas, posteriormente empresas hicieron negocio brindando servicio de cable a las personas con canales exclusivos, y con el paso del tiempo se dieron cuenta de que con algunos cambios al sistema, se podría proporcionar servicio de Internet de dos vías, ya a estas alturas el sistema de TV por cable comenzaba a transformarse de manera distribuida, a una red de área metropolitana. Para poder entender cómo funciona una MAN observe la figura 2.6 donde se aprecia que las señales de TV e Internet se alimentan hacia un amplificador, para enseguida transmitirse a las casas de las personas.

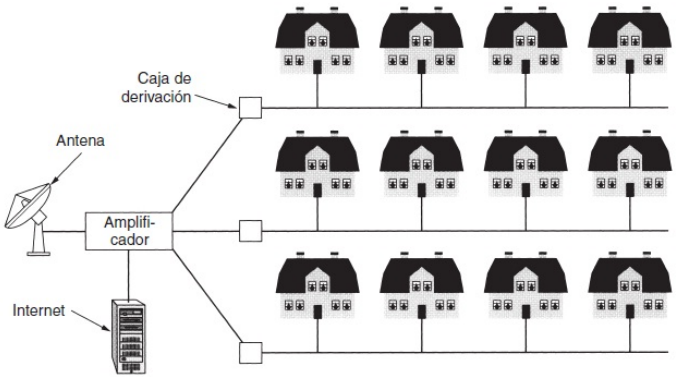

Figura 2.6: MAN, basada en TV por cable

Redes de área amplia (WAN: Wide Area Network)

Una red WAN abarca una gran extensión, ya sea un país o continente, esta red consta de dos componentes que son: las líneas de transmisión y elementos de conmutación. Las líneas de transmisión pueden estar hechas por cable de cobre, fibra óptica o incluso radioenlaces, por otro lado los elementos de conmutación son computadoras especializadas a conectar tres o más líneas de transmisión.

La mayoría de las WANŽs, poseen varias líneas de transmisión, que a su vez se conectan a un par de enrutadores, en caso de no existir líneas de transmisión deberán

hacerlo de forma indirecta, a través de otros enrutadores.

Se dice que cuando un paquete es enviado desde un enrutador a otro a través de uno o más enrutadores intermedios, el paquete se recibe en cada enrutador intermedio en su totalidad, y se almacena ahí hasta que la línea de salida requerida esté libre y, por último, se reenvía. Una subred organizada a partir de este principio se conoce como subred de almacenamiento y reenvío o de conmutación de paquetes. Casi todas las redes de área amplia (excepto las que utilizan satélites) tienen subredes de almacenamiento y reenvío.

No todas las WANs son de conmutación de paquetes. Una segunda posibilidad para una WAN es un sistema satelital, donde cada enrutador tiene una antena a través de la cual puede enviar y recibir. La figura 2.7 se muestra la dimensión de este tipo de red [4].

Figura 2.7: Dimensión de una WAN

Internet

Es clasificada como una red mundial de computadoras vinculadas por puertas de enlace (gateways), conformada por una multitud de pequeñas redes y computadores individuales, que independientemente poseen sistemas operativos diferentes, garantizando así el acceso a cualquier tipo de información, como fotografías enviadas por algún satélite, hasta la información conseguida de alguna universidad, o conseguir la descarga de un programa de utilidad [4]. La figura 2.8 muestra la expansión de esta red mundial.

En resumen, para poder clasificar el tipo de redes que existen, se hace tomando

Figura 2.8: La red pública Internet

en cuenta la distancia entre los equipos y por la tecnología que se esta utilizando. En la tabla 2.1 observamos de forma más clara el nombre de cada red de acuerdo a la distancia de expansión.

Descripción	**Distancia entre terminales**	**Ubicación**
PAN	1 m	Metro cuadrado
LAN	10 m	Cuarto
LAN	100 m	Edificio
LAN	1 km	Campus
MAN	10 km	Ciudad
WAN	100 km	País
WAN	1,000 km	Continente
Internet	10,000 km	Planeta

Tabla 2.1: Comparativa entre tipos de redes [4].

2.3. Modelo OSI

El modelo de referencia OSI (Interconexión de Sistemas Abiertos) se basa en una propuesta desarrollada por la ISO (Organización Internacional de Estándares), este posee siete capas en donde cada capa debe realizar una tarea bien definida, de acuerdo a los protocolos estandarizados internacionalmente. En seguida analizara cada capa del

modelo comenzando desde la capa inferior [4]:

1. Física:

   Esta capa realiza la transmisión de bits en su totalidad, mediante un canal de comunicación, es decir que cuando se está transmitiendo de un lado un bit 1, éste deberá recibirse en el otro lado como tal. Las interrogantes que genera este primer nivel tienen que ver con la duración de la transmisión, el establecimiento de la conexión inicial y el efecto de la finalización producida.

2. Enlace de datos:

   En el segundo nivel se tiene que transformar un medio de transmisión puro en una línea de comunicación, para que posteriormente al llegar a la capa de red, se presente libre de errores de transmisión, para realizar esto el emisor fragmenta los datos de entrada en tramas de datos, y los trasmite de forma secuencial, si se confirma el servicio como confiable, el receptor confirma la recepción de forma correcta de cada trama devolviendo una trama de confirmación de recepción.

3. Red:

   En el tercer nivel se controla las operaciones de la subred, y algo fundamental es determinar cómo se enrutan los paquetes desde su origen a su destino, las rutas pueden estar basadas en tablas estáticas manipuladas por algún administrador, o de manera dinámica, dejando así que los enrutadores decidan la ruta que seguirán los paquetes hacia el destino.

   Cuando hay muchos paquetes en la subred al mismo tiempo, se interpondrán en el camino entre ellos, provocando la existencia de cuellos de botella, y la responsabilidad de la capa de red es controlar esta congestión, cuando un paquete tiene que viajar de una red a otra para llegar a su destino, pueden surgir inconvenientes, uno de estos puede ser que el direccionamiento utilizado por la segunda red podría ser diferente de la primera, y en este caso la segunda probablemente no aceptaría todo el paquete, o quizás los protocolos son diferentes. En este nivel se tienen que resolver todos estos problemas para permitir una buena conexión.

4. Transporte:

   En el cuarto nivel se aceptan los datos originarios de las capas superiores, y si es necesario dividirlos, y enviarlos a la capa de red verificando que todos los fragmentos lleguen correctamente al otro extremo. Este nivel también determina qué tipo de servicio proporcionar cuando se realiza la conexión a la capa de sesión y finalmente a los usuarios de la red.

5. Sesión:

   El quinto nivel permite que usuarios de máquinas diferentes establezcan sesiones entre ellos, en donde se involucra el control de diálogo (es el seguimiento de a quién le toca transmitir), administración de token (este impide que las dos partes traten de realizar la misma operación al mismo tiempo) y la sincronización (se crean puntos de referencia en las transmisiones largas que permiten continuar desde donde se encontraban después de alguna caída).

6. Presentación:

   Al sexto nivel le compete analizar la sintaxis y semántica de la información transmitida, para permitir que las computadoras con diferentes representaciones de datos se puedan comunicar. Esta capa maneja estructuras de datos abstractas y permite definir e intercambiar estructuras de datos incluso de un nivel más bajo.

7. Aplicación:

   El último nivel comprende de varios protocolos que el usuario final requiere para su utilización, por ejemplo un protocolo de aplicación es HTTP (Protocolo de Trasferencia de Hipertexto), que es la base de Word Wide Web, de manera que cuando un navegador solicita una página Web, utiliza este protocolo para enviar al servidor el nombre de dicha página y como resultado el servidor devuelve la página si existe en el servidor Web.

En resumen cada una de las capas del Modelo OSI se puede enteder como lo expresa la tabla 2.2.

Capas	Función
Aplicación	Aplicaciones de Red: transferencia de archivos
Presentación	Formatos y representación de los datos
Sesión	Establece, mantiene y cierra sesiones
Transporte	Entrega confiable/no confiable de mensajes
Red	Entrega los paquetes y hace enrutamiennto
Enlace de datos	Transfiere, y verifica errores
Físico	Transmite datos binarios sobre un medio

Tabla 2.2: Modelo OSI

2.4. Modelo TCP/IP

Con la aparición de nuevas redes fue necesario implementar una nueva arquitectura de referencia, para dar conectividad a múltiples redes, esta llego a ser conocida como el modelo de referencia TCP/IP.

Este modelo cuenta con cuatro capas a diferencia con el modelo OSI que tiene siete. A continuación se analizara cada capa del modelo comenzando de abajo hacia arriba [4]:

1. Acceso a la red:

   En esta capa la computadora se tiene que conectar a la red mediante el protocolo IP, para que así pueda enviar y recibir paquetes.

2. Internet:

   El trabajo de esta capa es permitir que los equipos inyecten paquetes dentro de cualquier red para poder así viajar a su destino de manera independiente, posiblemente no lleguen en el orden de cómo fueron enviados.

   Aquí se define un paquete de formato y protocolo oficial llamado IP, además la capa de Red tiene la función de entregar paquetes IP al destinatario, por medio del enrutamiento de paquetes, pero sin dejar de lado el hecho de mantener un orden para evitar la congestión.

3. Transporte:

   Aquí se permite una conversación entre equipos, además se definen dos protocolos de transporte de extremo a extremo, el primero es TCP (Protocolo de Control de Transmisión), es un protocolo confiable y orientado a conexión, tiene la finalidad de permitir el flujo de bytes que genere una máquina y entregarlo sin errores en cualquier otra máquina en la red. Divide el flujo de bytes entrantes en mensajes discretos y pasa cada uno de ellos a la capa de Red, en el destino el proceso TCP receptor reensambla en el flujo de salida los mensajes recibidos. Cabe mencionar que TCP lleva un control de flujo para asegurarse que un emisor rápido no sature a un receptor lento con más mensajes de los que puede manejar.

   El segundo protocolo que maneja esta capa es UDP (Protocolo de Datagrama de Usuario), es un protocolo no confiable y no orientado a la conexión para aplicaciones que no desean la secuenciación o el control de flujo de TCP y que desean proporcionar el suyo. Además tiene un amplio uso en consultas únicas de solicitud-respuesta de tipo cliente-servidor en un solo envío, así como aplicaciones en donde la entrega es más importante que la precisa, como por ejemplo en la transmisión de voz y vídeo.

4. Aplicación:

   En esta capa encontraremos todos aquellos protocolos de nivel más alto, como TELNET que es un protocolo que incluye una terminal virtual, FTP protocolo dedicado para la transferencia de archivos, SMTP para el correo electrónico, posteriormente llego DNS dedicado para la resolución de nombres en sus direcciones de red.

En resumen cada una de las capas del Modelo TCP/IP se puede enteder como lo expresa la tabla 2.3.

Capas	Función
Aplicación	Aplicaciones de Red: transferencia de archivos
Transporte	Entrega confiable/no confiable de mensajes
Internet	Inyectar paquetes dentro de cualquier red
Acceso a la red	Conectar mediante el protocolo IP

Tabla 2.3: Modelo TCP/IP

2.5. Protocolos de comunicación

La Internet es el resultado de la interconexión de miles de redes con tecnologías diferentes, por lo tanto debido a esa diversidad que existe para transmitir la información, se necesita regular el tráfico de esta, es por eso que se recurre al uso de protocolos.

El uso de los protocolos lo vemos reflejado desde que accedemos a un sitio Web, o incluso cuando realizamos una llamada telefónica solicitando información por determinado producto a un proveedor, quien desempeñaría el papel de servidor. Por lo anterior a la forma de ponerse de acuerdo para el envío y recepción de algún producto adquirido telefónicamente, se le llama protocolo.

Informalmente se puede definir a un protocolo como aquellas normas y procedimientos útiles para trasmitir datos, conocidos por el emisor y el receptor [1].

2.5.1. Características y funciones del Protocolo IP

Las siglas IP significan Internet Protocol, este tiene las características de proporcionar un servicio de entrega de paquetes, y es un protocolo no orientado a conexión, lo que significa que no detecta ni recupera paquetes pérdidos o erróneos, no garantiza que los paquetes lleguen en orden ni duplicados.

Las funciones básicas de IP son [14]:

1. Direccionamiento:

   Muestra un sequema global de direccionamiento.

2. Fragmentación y reensamblaje de paquetes:

Divide los paquetes en fragmentos de un tamaño aceptable por la red.

3. Encaminamiento de datagramas:

   Encamina los paquetes por medio de tablas de ruteo.

Las direcciones IP, son números que constituyen la dirección de todo aquel dispositivo conectado a una red para ser localizado, mediante la cual un usuario pueda recibir aquellos archivos de información que solicitó. Debido al alto crecimiento y uso de la Internet por industrias, universidades y el gobierno, se esta usando en gran escala las direcciones IPv4 y por lo tanto van quedando pocas de estas, es por eso que pensando en esa situación en el año de 1990 se empezó a trabajar en una versión nueva de IP, considerando así el hecho de que no se agotaran las direcciones, las ventajas principales que se obtendrían, serían manejar miles y millones de terminales, reducir el tamaño de tablas de enrutamiento y proporcionar mayor seguridad en la verificación de autenticidad y confidencialidad, esta nueva versión recibe el nombre de IPv6. A continuación se mencionará de forma breve en que consiste cada una de estas versiones [4].

1. Formato de IPv4

   La dirección IPv4, tiene una longitud de 32 bits, estas direcciones están representadas por cuatro cifras separadas por puntos, o de cuatro octetos, los 32 bits se dividen en cuatro grupos de 8 bits cada uno, además cada uno de estos bytes se traduce a su equivalente en decimal, el resultado es que de cada conversión resulta un número comprendido entre 0 y 255.

   Las direcciones IP proporcionan dos datos que son, el número de red y el número de host. El número de bits empleado para definir la red y el número de bits que identifican al *host* (término que es utilizado en informática para referirse a las computadoras conectadas a una red) pueden cambiar. Cada dirección tiene un prefijo cuya longitud indica qué bits corresponden al identificador de red y cuáles al host, la longitud de este prefijo la establecen los bits de la dirección de la máscara de subred [1, 14]. La figura 2.9 muestra el formato de una dirección IPv4, con su respectiva máscara de subred.

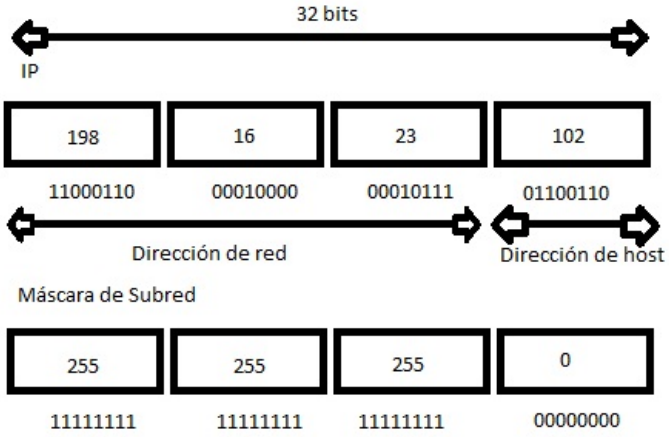

Figura 2.9: Formato de dirección IPv4

Desde hace mucho tiempo las direcciones IP se dividieron en cinco categorías, la clasificación se hizo para definir a las redes por tamaño, grande como clase A, mediano como clase B, pequeño como clase C, de uso multicast clase D y de uso experimental E.

La tabla 2.4 muestra en detalle como están conformadas estas clases, el número de redes que soportan cada una y la cantidad de host que se incorporan en su totalidad [14].

2. Formato de IPv6

   La dirección IPv6 está conformada por una longitud de 128 bits, separadas por grupos de cuatro dígitos hexadecimales, dando como resultado ocho grupos, cada grupo con una longitud de dos bytes, a diferencia de IPv4 ya no se separan por medio de un punto, sino de dos puntos. En la figura 2.10 se muestra el formato de una dirección IPv6.

   Otra consideración a tomar en cuenta es la representación de la IPv6 de diferentes maneras, por ejemplo, si tenemos la dirección: 2001:0db8:85a3:0000:0000:8a2e:0370:7334 la podremos representar omitiendo los **ceros iniciales** de cada grupo, aunque cada grupo debe contener al menos un dígito hexadecimal y quedaría de la siguiente manera:

Clase	**Identificador de clase en binario**	**Bits de red en decimal**	**Número de redes por clase**	**Número de hosts por cada red**	**Ejemplo**
A	0	0-127 (7bits)	$2^7 = 128$	$2^24 = 16777216$	**26.**56.120.9
B	10	128-191 (14bits)	$2^{14} = 16384$	$2^{16} = 65536$	**147.96.**50.110
C	110	192-223 (21 bits)	$2^{21} = 2097152$	$2^8 = 256$	**217.6.95.**44
D	1110	224-239 (28 bits)			**224.0.0.1**
E	1111	Experimental (28 bits)			

Tabla 2.4: IP's basadas en Clases

2001:db8:85a3:0:0:8a2e:370:7334.

Otro ejemplo de representar este tipo de direcciones es si tenemos la misma dirección: 2001:0db8:85a3:0000:0000:8a2e:0370:7334 en donde uno o más grupos de ceros pueden ser sustituidos por dos puntos, esta sustitución puede realizarse únicamente una vez en la dirección. En caso contrario, obtendríamos una representación ambigua, si pueden hacerse varias sustituciones, debemos observar la de mayor número de grupos; si el número de grupos es igual, debemos hacer la situada más a la izquierda, con esta regla, reduciríamos aún más la dirección quedando así: 2001:db8:85a3::8a2e:370:7334 [12, 2].

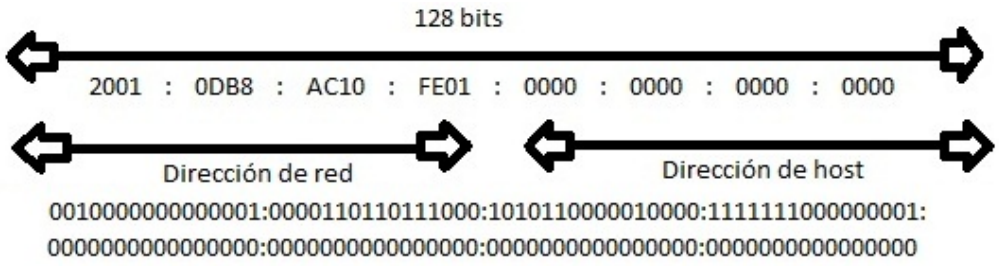

Figura 2.10: Formato de dirección IPv6

Capítulo 3

DNS

A lo largo de este capítulo se abordarán los orígenes y funciones de un servidor DNS, así como la clasificación que hay en los dominios, y las herramientas que necesita para el correcto funcionamiento como lo son los registros de recursos, y posteriormente se mostrarán las ventajas y desventajas de este modelo.

3.1. Orígenes y funciones de DNS

Para mencionar las funciones de un DNS, debemos saber que su funcionalidad se rige mediante servidores. Un servidor es un nodo que forma parte de una red, que provee servicios a otros nodos denominados clientes, estos no necesariamente deben ser una máquina de última generación con grandes capacidades y con características especiales, un servidor puede ser desde una computadora sencilla sin muchas complejidades, hasta una máquina con características muy detalladas. Todo esto depende del uso que se le dé al servidor, si se desea se puede convertir incluso el equipo que estamos ocupando en un servidor, instalando un programa que trabaje por la red y a la que los usuarios puedan ingresar.

Antes de que se constituyerá en gran medida lo que ahora conocemos como la red de redes Internet, esta se administraba mediante una lista de nombres de computadoras con sus respectivas direcciones de Internet relacionadas a un archivo denominado host.txt que se hacía público en la red, las modificaciones eran realizadas por voluntarios de

un grupo de Stanford Research Institute (SRI), en donde cada noche todos los hosts obtenían este archivo para ver las posibles actualizaciones que se le habían realizado. Sin embargo, con el paso del tiempo el crecimiento acelerado de la Internet con miles de estaciones de trabajo, demostró que seguir utilizando el mismo sistema con un sólo archivo de configuración para la gestión de una gran cantidad de equipos, era un método que ya no sería funcional, además de que ocurrirían conflictos constantes entre los nombres de los hosts, todo esto llevo a una nueva forma que permitiera mejorar el registro de computadoras conectadas a Internet.

Posteriormente se dio origen a los Servidores de Nombres de Dominio, una solución elaborada en 1983 por Paul Mockapetris. Lo que el exponía era básicamente un sistema jerárquico, definiendo así una estructura jerárquica por niveles. DNS es una base de datos distribuida que almacena información relacionada a nombres de dominio en una red, permitiendo así acceder a distintas direcciones de Internet mediante la traducción de direcciones IP a nombres, esto es una gran ventaja ya que a las personas se les dificulta recordar una dirección, además también tienen diversas utilidades como lo es el utilizar buzones de correo electrónico así como el uso de otros recursos mediante sus direcciones de red.

La asignación de nombres a direcciones IP puede ser por ejemplo con esta dirección www.google.com que posee la IP 74.125.133.104, a los usuarios se les hace más fácil recordar un nombre que una dirección numérica. Cabe destacar que no se sabe en qué momento se pueda cambiar de dirección, entonces si fuera ese el caso no se podría acceder a el sitio visitado con anterioridad con la misma referencia, en cambio con la traducción a un nombre no importa si este tiene cambios una y otra vez, ya que dicho nombre siempre será el mismo [4].

3.2. Nombres de dominio

Un nombre de dominio es una tira de menos de 255 caracteres, conformada por etiquetas separadas en niveles por medio de puntos, entre los cuales debe tener menos de 63 caracteres, estos pueden ser desde letras, números y el guión medio, pero hay que

tomar en cuenta que no puede iniciar ni terminar con guión y puede tener hasta 127 niveles siempre y cuando no se rebase la condición de los 255 caracteres. Los nombres no se distinguen entre mayúsculas y minúsculas, además están organizados de forma jerárquica en forma de árbol, desde el más bajo nivel jerárquico, hasta el más alto de modo que toda aquella secuencia formada por etiquetas existentes llega hasta la raíz.

Por lo tanto a cada nivel de la estructura se le asigna un nombre o etiqueta comenzando así desde la parte derecha, el nivel cero o también conocido como raíz no posee un nombre, posteriormente, el nivel 1 puede ser alguno como .net, .com, o .mx, después se tiene por debajo mas niveles hasta llegar al final y conformar en su totalidad la estructura del nombre. A continuación se listan algunos ejemplos de nombres de dominios válidos:

1. www.dominiovalido.com
2. www.dominiovali-do2.com
3. www.dominiovalidooooooooooooooooooo-o-o.com

La siguiente lista muestra algunos casos de dominios no válidos:

1. www.-dominionovalido.com
2. www.dominionovalidooooooooooooooooooooooooooooooooooooooooooooooooooooo.com
3. www.-dominionovalido-.com

Esta asignación de nombres es admnistrada por la IANA (Internet Assigned Numbers Authority), y por otra parte esta la ICAN (Internet Corporation for Assigned Names and Numbers) que también llevan un control de estos registros. En la figura 3.1 se muestra la estructura de los nombres de dominio de forma más entendible [18, 10, 4].

3.2.1. Clasificación de los dominios

Para llevar a cabo la administración de un grupo tan grande y continuamente cambiante de nombres se requiere de una organización muy adecuada, es por eso que la

Figura 3.1: Formato de un nombre de dominio

división de este trabajo se realiza mediante la división de 200 dominios de nivel superior, de los cuales abarca muchos hosts, posteriormente cada uno de estos se divide en otros subdominios, y así sucesivamente.

Por lo tanto, en el nivel absoluto superior o raíz, existe una clasificación de los dominios como geográficos y genéricos. Los ccTLD (Country Code Top Level Domain) son identificados como geográficos o territoriales, y son nombres de dominio de primer nivel, este grupo pretende identificar los nombres por medio de una división de países. En la tabla 3.1 se muestra algunos ejemplos de ccTLD más conocidos con sus respectivos usos.

Extensión	**País**	**Extensión**	**País**
.zm	Zambia	.mx	México
.ye	Yemen	.it	Italia
.uy	Uruguay	.hr	Croacia
.pr	Puerto Rico	.de	Alemania

Tabla 3.1: Ejemplos de ccTLD

En cuanto a la clasificación de los dominios genéricos o gTLD (generic Top Level Domain), no referencian a algún lugar geográfico. Los gTLD son denominados genéricos porque realizan la división en base al tipo de función que desempeña la organización. La tabla 3.2 muestra algunos de los gTLD con su respectivos usos.

Para poder obtener un dominio de segundo nivel, ya sea para uso personal o para alguna compañía u organización, se debe seguir un proceso. En primer lugar se necesita ir con el registrador del dominio de nivel superior correspondiente, y verificar si en

realidad está disponible ese nombre que escojamos, ya que alguien más podría haberlo elegido con anterioridad y lo esté ocupando, en caso contrario lo que procede es que el solicitante estará pagando una pequeña cuota anual y obtiene los derechos del nombre. Por lo tanto, los nombres adquiridos reflejan los límites organizacionales, no las redes físicas, puede suceder el caso de que haya dos departamentos uno de recursos humanos y otro de informática, ubicados dentro del mismo edificio pero ambos comparten la misma LAN, eso no significa que deban tener el mismo dominio, y de manera similar si ahora estos dos departamentos se encuentran ubicados en edificios diferentes, todos los hosts podrían pertenecer al mismo dominio [5, 4].

Extensión	**Descripción**
.gov ó .gob	Gobierno
.mil	Administración militar de EE.UU
.int	Internacional
.edu	Para instituciones educativas
.name	Personas
.biz	Para empresas (aligera la carga a los TLD .com)
.info	Personas, empresas, organizaciones, (abiertos a todos), que publican información sobre sus actividades
.org	Reservado inicialmente para las organizaciones y asociaciones
.net	Tradicionalmente destinado para los sitios conectados con actividades de infraestructura Internet
.com	Previsto inicialmente para las compañías comerciales, actualmente es para todo tipo de sitios

Tabla 3.2: Ejemplos de gTLD

3.3. Modelado del DNS

Como se había mencionado anteriormente el sistema DNS, está organizado de forma jerárquica, de tal forma que se le puede asimilar a un sistema de archivos que posee una computadora como lo muestra la figura 3.2. En dicha estructura jerárquica se definieron niveles de la misma manera como lo hay en un sistema de archivos organizados en forma de directorios, además de incluir los nombres o etiquetas, pero no solamente es eso sino que también se requiere para el correcto funcionamiento de los DNS, los siguientes agentes:

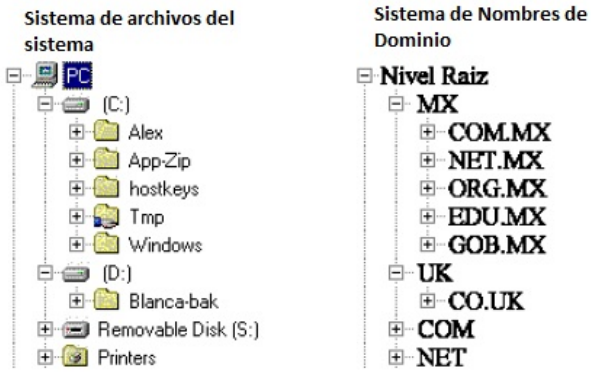

Figura 3.2: Relación jerárquica entre un sistema de archivos y un Sistema de Nombres de Dominio.

1. **Servidores:**

   Estos reciben las consultas y envían las repuestas necesarias, además tienen la autorización de acceder a bases de datos de forma local, en donde se encuentra una parte de la totalidad de las bases de datos del sistema de nombres, incluso se puede tener acceso de forma indirecta a el resto de las bases de datos con la ayuda de otros servidores.

2. **Resolvedores:**

   Estos son programas o librerías que reciben peticiones de algunas aplicaciones de usuario, las lee y las traduce a consultas DNS extrayendo la información solicitada,

por ejemplo cuando tenemos una aplicación dedicada a copiar documentos de algunas otras computadoras, el usuario podría indicar a que equipo desea acceder por medio de su nombre, es ahí cuando la aplicación deberá realizar una llamada al resolvedor para obtener la dirección a partir del nombre.

3. **Zonas:**

   Las zonas permiten realizar la división de un dominio con el propósito de simplificar la gestión y administración del sistema, en cada zona se encuentra un nodo llamado superior y consecuentemente todos los que estén en los niveles jerárquicamente inferiores hasta llegar a los nodos terminales. El propósito de esto es poder delegar la administración asignando la responsabilidad para dicha gestión a otras autoridades, y así el administrador designado será el único con esa autoridad de añadir o borrar nodos dentro de su zona, modificar información de esos nodos e incluso crea nuevas subzonas y delegar su gestión en otras autoridades.

   Para hacer una consulta a una zona esta debe estar almacenada en la base de datos local del servidor del que se dice que es un servidor con autoridad sobre esta zona, después este puede contestar directamente aquellas consultas que reciba sobre todos los nodos de su zona, sin que tenga que consultarlo con otros servidores. Por lo tanto se puede decir que el servidor estará enviando respuestas con autoridad como lo muestra la figura 3.3.

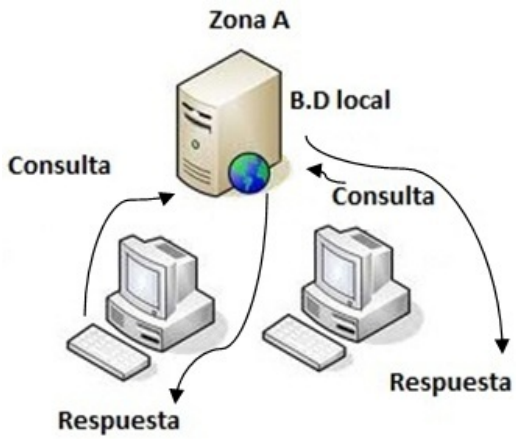

Figura 3.3: Respuesta con autoridad ante una consulta.

Existen dos maneras de generar una respuesta ante una consulta que se refiere a otra zona, estas son las siguientes:

1. **Modo no recursivo:**

   Ocurre cuando se genera una consulta por parte de algún nodo a el servidor, y este no tiene la respuesta en su base de datos local, y como consecuencia únicamente la respuesta incluye una referencia a algún servidor que puede proporcionar más información, y por lo tanto el cliente debe preocuparse en continuar realizando la búsqueda con otras consultas hasta encontrar la respuesta final. En la figura 3.4 se muestra la forma en cómo funciona este tipo de respuesta.

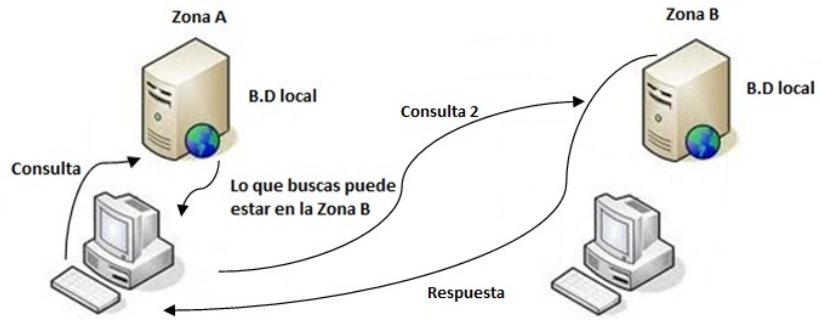

Figura 3.4: Respuesta en modo no recursivo.

2. **Modo recursivo:**

   Se da cuando se genera una consulta por parte de algún nodo en el servidor, y en caso de no obtener la respuesta en su base de datos local, busca la información donde sea necesario encontrarla, y sólo puede retomar la respuesta final o un error, pero no referencia a otros servidores como en el caso anterior. En la figura 3.5 se muestra la forma cómo funciona este tipo de respuesta.

Generalmente la especificación establece que todos los servidores deben soportar el modo no recursivo, y el modo recursivo es algo opcional, aunque en el uso la mayoría de las consultas por parte de los clientes se da en el modo recursivo. Cuando se responde ante una consulta en el modo recursivo se solicita la información en otros servidores y se devuelve dicha respuesta a quien hizo la petición, entonces es costumbre añadir

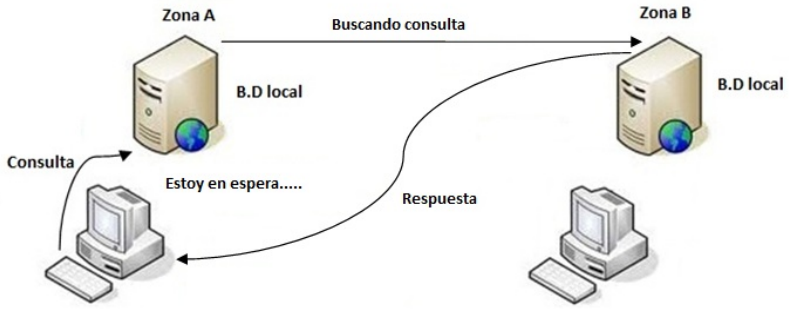

Figura 3.5: Respuesta en modo recursivo.

información obtenida a su base de datos, lo que se denomina caché (es un tipo de memoria que funciona de manera similar a la memoria RAM pero es de menor tamaño y de acceso rápido que reduce el tiempo de acceso a los datos que se utilizan con más frecuencia), de esta forma cuando se realiza otra consulta con la misma información no será necesario volverla a solicitar de nuevo a otros servidores, más bien se aprovecha la que existe en caché. Pudiera darse el caso de que los datos se hayan modificado en el servidor de origen desde que solicitaron por primera vez, entonces el servidor debe poner al tanto al cliente de que se trata de una respuesta sin autoridad, también los resolvedores suelen tener este tipo de prácticas guardando en caché las respuestas recibidas de otros servidores.

Por lo mencionado anteriormente ante una consulta siempre debemos obtener una alta fiabilidad en el servicio, otra especificación sobre los DNS es que se requiere que para cada zona exista como mínimo dos servidores con autoridad respondiendo a las siguientes funciones:

1. Uno de ellos debe ser el servidor primario que servirá para guardar los ficheros originales de la base de datos correspondiente a la zona, dejando así que el administrador actualice dichos archivos de configuración cada vez que se realice una modificación en sus nodos.

2. El segundo debe actuar como servidor secundario y este debe actualizar de forma automática sus bases de datos a partir de las del primario. Esto es una ventaja ya

que si sólo existiese un servidor, podría estar temporalmente inaccesible debido a una caída de la red, o incluso del mismo servidor, y en ese caso los clientes estarían enviando consultas sin recibir respuesta, en cambio con esta modalidad los clientes pueden enviar las consultas a uno de los servidores secundarios y obtener una respuesta satisfactoria [10, 19].

3.4. Registros de recursos

Todo nombre de dominio, llámese host individual o un dominio de nivel superior, consta de un conjunto de registros asociados a él, habiendo mencionado al host individual, el registro de recurso más común es su dirección IP. Cuando se da un nombre de dominio, lo que recibe en realidad son sólo los registros de recursos asociados a dicho nombre, como por ejemplo CNAME, MX, PTR entre otros, entonces se puede decir que la función principal del DNS es relacionar los dominios con los registros de recursos.

La estructura de un nombre de dominio posee cinco secciones, y por motivos de eficiencia se codifican en binario, además el orden de las secciones no es significativo, entonces el formato que se sigue es el siguiente: **Nombre_dominio, Tiempo_de_vida, Clase, Tipo, Valor** [18, 10].

1. **Nombre_dominio:**

   Este campo contiene el nombre de dominio del nodo al que está relacionado el registro, es muy importante porque es la clave de la búsqueda para atender las consultas.

2. **Tiempo_de_vida (TTL):**

   Da a conocer la estabilidad que tiene dicho registro, y se considera que la información es sumamente estable cuando encontramos un valor grande, y la información sumamente volátil cuando recibe un valor pequeño, demostrando así el tiempo máximo que un servidor puede guardar el registro en su caché.

3. **Clase:**

   Muestra la familia de protocolos utilizada, en el caso de tener información de Inter-

net siempre se utiliza IN (es un tipo de clase), para efectos de información donde no se requiere Internet se utilizan otros códigos, que raramente son utilizados.

4. **Tipo:**

   Aquí nos enseña sobre todo aquel tipo de información que contiene dicho registro, podemos encontrar muchos valores, y los tipos de información con que cuenta son los siguientes:

   - A:

     Tiene la responsabilidad de asociar la dirección IP con los nombres, para cada equipo debe sólo existir un registro A.

   - CNAME:

     Relaciona un alias con su nombre canónico, este nombre canónico esta determinado con un registro A.

   - MX: Declarara un servidor de correo en algún dominio, cuando un agente de transferencia de mensajes quiera entregar información al correo del dominio, lo que hará es tratar de conectarse a estos servidores hasta lograr entregar dicho mensaje. El formato del registro MX es el siguiente:

     [domain] [ttl] [class] MX preference host

   - PTR:

     Es utilizado para asociar nombres a partir de un dominio especial denominado in-addr.arpa, es decir obtiene los nombres a partir de las direcciones IP.

   - NS:

     Apunta a un servidor de nombres con autoridad para una cierta zona.

   - HINFO:

     Esta parte muestra la información sobre el tipo de computadora que se está utilizando, mostrando así las características sobre el hardware y el software, el formato de este valor es el siguiente:

     [domain] [ttl] [class] HINFO hardware software

Un ejemplo sobre este tipo de valor es Dominio 36500 IN HINFO IMB-PC LINUX3.0, que nos describe un sistema Intel ejecutado en linux.

- SOA:

  Significa Start of Authority o Comienzo de Autoridad y da a conocer información sobre una zona de autoridad, cada zona primaria debe tener un registro del tipo SOA, los datos relacionados con SOA comprende los siguientes campos:

  a) Origin:

  Se refiere al nombre canónico del servidor de nombres primario, y es utilizado como nombre absoluto.

  b) Contact:

  Muestra la información de la persona que es responsable de mantener el dominio, aquí en lugar de utilizar el carácter @ se utiliza un punto, como por ejemplo si el responsable es Alberto, este campo quedaría: Alberto.dominio.com

  c) Serial:

  Es un número de versión del archivo de zona expresado de forma decimal, se utiliza para mantener actualizados los servidores secundarios, estos hacen una petición cada determinado tiempo a el registro SOA del servidor primario, y hace una comparativa entre el número de versión con el que tiene en caché, en caso de haber cambiado entonces el servidor secundario solicitará de nuevo la información de zona del primario para actualizarse. El formato para este campo se suele denotar como la versión de un día actual, como se muestra a continuación AAAAMMDDnn siendo AAAA el año, MM el mes, DD el día y nn el número de revisión de ese día (01 quedaría si no existe más de una). Un ejemplo de esto podría ser el siguiente 2001072201 para el 22 de julio de 2001.

  d) Refresh:

  Da a conocer un valor en segundos, que esperan los servidores secun-

darios por cada petición de registros SOA a los servidores primarios, está conformado por un número decimal de hasta ocho dígitos.

e) Retry:

Número que determina las veces de reintentos de comunicación que han tenido los servidores secundarios con los primarios cuando una petición de zona falla.

f) Expire:

Muestra el tiempo en segundos que demorará el servidor en descartar los datos de zona cuando no se ha podido contactar con el servidor primario.

g) Minimum:

Es un valor predeterminado para el valor ttl en los registros de recursos que no lo hayan especificado, indicando así a otros servidores que descarten los registros de recursos tras cierto tiempo.

- MB: Especifica el nombre de un buzón de correo electrónico.
- MG: Especifica el nombre de un miembro en un grupo de correo electrónico.
- MR: Da un nombre nuevo de un buzón de correo.
- MINFO: Da información sobre un buzón o lista de distribución de correo.
- WKS: (Well Know Services o Servicios bien conocidos) Genera una lista de aquellos servicios que proporciona un equipo de cómputo.
- TXT: Genera un texto descriptivo.
- NULL: Arroja un registro vacío.

5. **Valor:** Esta última sección depende mucho del tipo de registro seleccionado, como lo es A, CNAME, HINFO, MX, PTR, NS.

Después de haber analizado la estructura de un nombre de dominio, se muestra un ejemplo en las siguientes líneas donde se describe como son utilizados realmente estos campos de registros de recursos [18, 10]:

```
\# archivo de configuración que muestra como son
utilizados los registros de recursos

acme.com. IN SOA servidor.acme.com. admin.acme.com. (
38; SERIAL
7200; REFRESH
600; RETRY
3600000; EXPIRE
60); MINIMUM
NS servidor.acme.com.
NS servidor2.acme.com.
NS dns.competencia.com.
MX 10 correo.acme.com.
MX 20 servidor.acme.com.
servidor.acme.com. A 128.52.46.32
A 128.32.11.99
servidor2.acme.com. A 128.52.46.33
correo.acme.com. A 128.52.46.34
```

En resumen para poder asimilar todo el contenido antes mencionado, la tabla 3.3 muestra los cinco campos de la estructura de un nombre de dominio.

Campos	**Descripción**
Nombre_dominio	Nombre para atender las consultas
Tiempo_de_vida (TTL)	Tiempo máximo para guardar un registro
Clase	Familia de protocolos utilizada
Tipo	Tipo de información del registro
Valor	Valores que dependen del tipo de registro seleccionado

Tabla 3.3: Estructura de un nombre de dominio y sus funciones

3.5. Resolución directa e inversa

Básicamente la operación que realiza un DNS es obtener la dirección IP correspondiente a un nombre, a esto se le denomina resolución directa, pero también es necesario agregar la operación opuesta, que es encontrar algún nombre a partir de la dirección IP, a esto le llamamos resolución inversa y es usado para realizar una comprobación de identidad del cliente, la herramienta que se utilizara para realizar tal proceso es el dominio in-addr.arpa, que contiene las direcciones IP de todos los sistemas en una notación de puntos invertida, por ejemplo a la dirección 192.168.1.40 le corresponde el nombre 1.168.192.in-addr.arpa y se relaciona con el registro de recurso PTR. Para entender mejor el funcionamiento de esto a continuación se muestra el proceso que se debe seguir para una correcta resolución inversa.

1. Supongamos que a un departamento le delegamos autoridad sobre la dirección 149.76.8.3, y por nombre queremos que tenga departamento.com, entonces el archivo de configuración en relación con el registro de recurso PTR para la asociación y correcta resolución inversa podría quedar de la siguiente manera:

```
; dominio 8.76.149.in-addr.arpa
@ IN SOA departamento.com root.departamento.com (
1999090200  ; Serial
360000      ; Refresh
3600        ; Retry
3600000     ; Expire
3600        ; Cache TTL
)
 IN PTR cs.departamento.com
```

2. Se crea la zona para el nombre departamento.com, el archivo de configuración sería el siguiente [18, 6]:

```
zone "departamento.com" {
```

```
type master;
    file "/etc/bind/db.departamento.com";
};
zone "8.76.149.in-addr.arpa" {
    type master;
file "/etc/bind/db.8.76.149";
};
```

Por lo tanto la representación de forma general en cuanto a la resolución directa e inversa se puede ver reflejada en la figura 3.6.

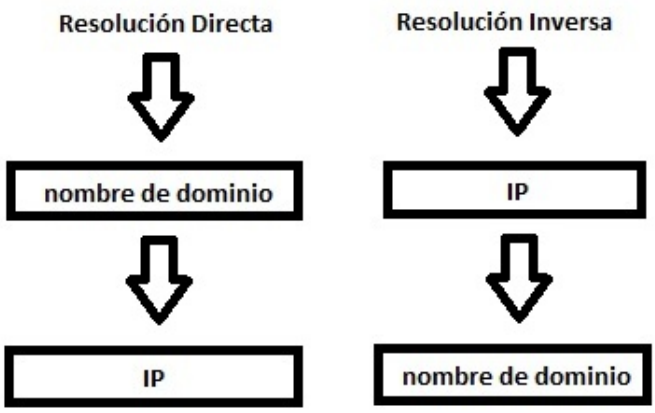

Figura 3.6: Representación de la resolución directa e inversa

3.6. World Wide Web

Se ha mencionado que es un nombre de dominio, la declaración de zonas y algunos otros aspectos más relacionados a algunas configuraciones típicas para el buen desempeño de un servidor DNS, ahora toca señalar la relación que presenta la World Wide Web, con el protocolo HTPP.

La World Wide Web es una Red de Extensión Mundial que permite acceder a documentos vinculados distribuidos en miles de máquinas de toda Internet por medio de páginas Web, esto comenzó como una propuesta inicial para una red de documentos

vinculados en el CERN (Centro Europeo de Investigación Nuclear), por medio del físico Tim Berners-Lee en marzo de 1989, más tarde en 1991 conforme se lograban tener más avances se presentó el primer prototipo basado en texto. Con el paso del tiempo se implementaron los navegadores para el acceso a la Web, como lo son Internet Explorer y Netscape que ya no existe actualmente, la función de estos era obtener una página solicitada interpretando el texto y desplegando en pantalla la información requerida. La forma como funciona la petición y respuesta de acceso a la Web se da por medio del modelo Cliente-Servidor, a continuación se examinará la forma de actuar de cada una de estas partes.

Cliente:

Cuando el usuario solicita una dirección, el navegador lleva a cabo su tarea en una serie de pasos para obtener la página a la que se está apuntando, supongamos que dicho usuario solicita la dirección www.google.com.mx. entonces el navegador tiene que realizar lo siguiente:

1. El navegador determina la dirección solicitada.
2. El navegador solicita al DNS la dirección www.google.com.mx.
3. El DNS responde con 74.125.227.216.
4. El navegador realiza una conexión TCP por medio del puerto 80 en 74.125.227.216.
5. Se envía un mensaje solicitando el archivo index.html.
6. El servidor www.google.com.mx. envía el archivo index.html.
7. Se libera la conexión TCP
8. El navegador despliega todo el contenido de index.html.

Servidor:

Cuando el cliente realiza la petición de algún sitio, entonces el navegador obtiene la dirección IP del servidor estableciendo una conexión TCP por medio del puerto 80, posteriormente el servidor responde devolviendo el archivo index.html para que el navegador lo despliegue en pantalla, para realizar esto se requiere de los siguientes pasos:

1. El servidor acepta una conexión TCP de un cliente.

2. Obtiene el nombre del archivo solicitado.

3. Obtiene el archivo del disco.

4. Regresa el archivo al cliente.

5. Se libera la conexión TCP.

Por lo tanto, después de haber analizado el funcionamiento de la Web, se puede resumir dicho proceso mediante la figura 3.7, donde el navegador despliega una página Web en una máquina cliente después de que el usuario hizo la petición que está vinculada al servidor con el nombre abcd.com, de igual forma esa página puede estar referenciada hacia otro sitio, que podría ser un servidor xyz.com y se vuelve a realizar el mismo proceso de forma indefinida [4].

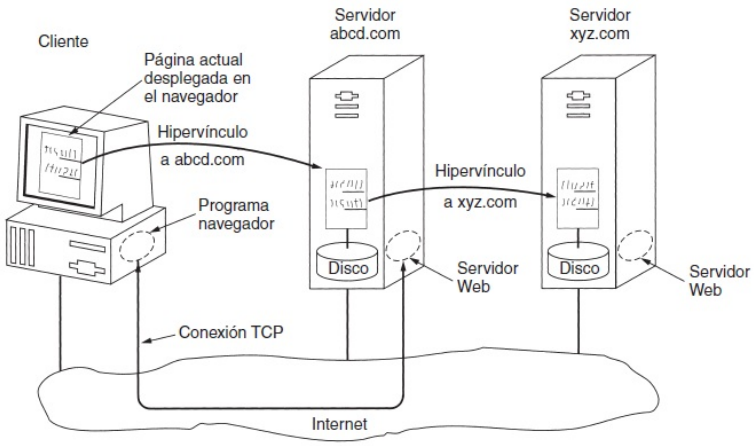

Figura 3.7: Funcionamiento de la World Wide Web

3.7. HTTP

HTTP es un protocolo dedicado a ser utilizado en la transferencia de la Word Wide Web, significa Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto), y decide que mensajes pueden enviar los clientes a los servidores y las respuestas

que obtienen, todos los clientes deben someterse a este protocolo, este se rige por dos características:

Conexiones y métodos:

Se habla de conexiones cuando se refiere a la forma cotidiana en que un navegador realiza una conexión con el servidor por medio de TCP en el puerto 80 de la máquina del servidor, por lo tanto no se sabe cuanta información o iconos pueda contener el sitio al se está contactando, es por eso que HTTP soporta conexiones persistentes, haciendo posible la solicitud y respuesta en múltiples solicitudes logrando que la sobrecarga resultante de TCP sea mucho menor por cada una de estas solicitudes.

HTTP fue diseñado con miras hacia el futuro, teniendo las herramientas necesarias para aplicaciones orientadas a objetos, debido a esto soportan otro tipo de operaciones llamadas métodos, diseñadas para realizar distintos tipos de solicitudes en la comunicación, estas peticiones son acciones a realizar por medio de solicitud y respuesta. La tabla 3.4 muestra los métodos de solicitud que maneja HTTP.

Método	Descripción
GET	Solicita la lectura de una página Web
OPTIONS	Consulta ciertas opciones
HEAD	Solicita la lectura del encabezado de una página Web
CONNECT	Reservado para uso futuro
PUT	Solicita el almacenamiento de una página Web
POST	Inserta algo a un recurso con nombre (por ejemplo, una página Web)
TRACE	Repite la solicitud entrante
DELETE	Elimina la página Web

Tabla 3.4: Métodos de solicitud HTTP

Cuando los usuarios que navegan por la Web intentan obtener acceso al contenido de un servidor mediante HTTP, cada solicitud obtiene una respuesta que consiste en devolver un código numérico, que indica si el intento de conexión se ha realizado o no correctamente, a esto se le llama código de estado, y consta de tres dígitos, el primer dígito se utiliza para dividir las respuestas en cinco categorías, los otros dos muestran

el tipo de error que se presenta, se puede observar detenidamente en qué consiste cada uno en la lista que se muestra a continuación:

1. 1xx: Casi no son utilizados pero indica que el servidor está listo para iniciar la primera parte de una solicitud y que está esperando el resto.

2. 2xx: Significan que la solicitud se manejó de manera exitosa y que se regresa el contenido si es que existe alguno.

3. 3xx: Indica al cliente que debe hacer una búsqueda en otro lado.

4. 4xx: Indican que la solicitud falló provocado por un error del cliente, como por ejemplo una solicitud inválida o alguna página no existente.

5. 5xx: Indica que el servidor tiene un problema, debido a un error en su código o a una sobrecarga temporal.

La presencia de todos estos códigos de estado, puede ser temporal, por lo tanto, quizás se deba consultar la disponibilidad del sitio Web en otro momento. Si el problema persiste, es posible que se tenga que poner en contacto con los administradores del sitio y pedirles una solucion. La tabla 3.5 muestra los grupos de respuesta de los códigos de estado con su significado y algunos ejemplos [4].

Código	**Significado**	**Ejemplos**
1xx	Información	100 = el servidor está de acuerdo en manejar la solicitud del cliente
2xx	Éxito	200 = la solicitud es exitosa; 204 = no hay contenido
3xx	Redirección	301 = página movida; 304 = la página en caché aún es válida
4xx	Error del cliente	403 = página prohibida; 404 = página no encontrada
5xx	Error del servidor	500 = error interno del servidor; 503 = trata más tarde

Tabla 3.5: Códigos de estado

3.8. Ventajas y desventajas de DNS

La utilización de los nombres de dominio nos ayuda a resolver una dirección sin que se tenga que poner directamente la IP, pero más allá de simplemente ver en nuestra pantalla la información que se solicita directamente al servidor, se debe tomar en cuenta lo importante que es analizar a fondo como es que la información no siempre viaja de la manera más segura, es por ello que a continuación se listan ciertas ventajas y desventajas, sobre el comportamiento de los DNS:

Ventajas:

1. Desaparece la carga excesiva en la red y en los hosts, debido a que la información está distribuida por toda la red al tratarse de una base de datos distribuida.

2. No hay duplicidad de nombres, y el problema se elimina debido a la existencia de dominios controlados por un único administrador, puede haber nombres iguales pero en dominios diferentes.

3. Existe consistencia de la información, ahora la información que está distribuida es actualizada automáticamente por las consultas que se generen sin intervención de ningún administrador.

Desventajas:

1. Si no se cubre el gasto del servicio en el tiempo requerido, nuestro nombre de dominio podría quedar fuera de la red.

2. Si se produjera algún error en la configuración sobre los DNS, podría ser muy inoportuno y costoso el procedimiento para desinstalar y volver a empezar con la configuración.

3. Se pueden presentar casos de suplantación de nombres de dominio.

4. Se pueden presentar casos de múltiples peticiones de consultas que pudieran ocasionar que el servidor no responda de la manera adecuada y se sature en la respuesta.

5. Se expone al retraso de la red mundial Internet cuando se da de alta un nombre de dominio, ya que se tiene que esperar a que el nombre resuelva correctamente.

Capítulo 4

Ataques sobre DNS

En este capítulo se abordan aspectos relacionados con la seguridad informática, demostrando cómo es que los servidores de nombres de dominio son susceptibles a fallas provocadas por determinados ataques, además pueden ser ejecutados en alguna computadora con cualquier sistema operativo.

Cada uno de los ataques desempeñan varias funciones, pero el objetivo principal radica en que el servicio no sea utilizado en el momento en que se solicita, o que el usuario pueda ser víctima de engaños para entregar datos confidenciales. Algunos ejemplos de estos ataques son: DoS (Denial of Service - Denegación de servicio), escaneo de puertos, DNS Spoofing (suplantación de identidad por nombre de dominio) y Phishing (técnica que permite obtener usuarios y contraseñas).

4.1. Tipos de ataques

Un ataque siempre aprovecha cualquier vulnerabilidad en un determinado sistema informático, comprometiendo así el correcto funcionamiento de este. Comúnmente las vulnerabilidades suceden por errores de programación o de alguna deficiente configuración del servicio. En algunas ocasiones es difícil darse cuenta de esto como usuario final, porque se está acostumbrado a ver sólo los resultados de un sistema, en este caso los datos que arroja como el único punto de medición de la calidad, es por eso que se derivan dos tipos de ataques que pueden llegar a afectar algún sistema [9]:

1. Ataques pasivos:

   Son todos aquellos donde el intruso monitorea el tráfico en la red, escuchando así lo que sucede para únicamente obtener información que esta siendo transmitida.

2. Ataques activos:

   En este tipo de ataques, el intruso interfiere con el tráfico legítimo que fluye a través de la red, actuando así de manera engañosa con el protocolo de comunicación.

En el resto del capítulo se presenta una serie de ataques realizados de forma exitosa al servidor DNS, tanto pasivos como activos.

4.2. DoS (Denial of Service)

Este ataque es de tipo activo, su objetivo es provocar que algún servicio o programa deje de funcionar, ya sea total o parcialmente, y que los usuarios legítimos carezcan de esa disponibilidad retrasando la respuesta para el uso individual. Generalmente, este ataque se puede dividir en dos clases:

1. Denegación de servicio por inundación:

   Saturan un equipo con varias solicitudes, más de las que puede manejar para que no responda a las solicitudes reales.

2. Denegación de servicio por explotación de vulnerabilidades: Es provocada por alguna falla en el software y es utilizada de la manera más adecuada por intrusos que quieren agotar los recursos del sistema, provocando así, que se vuelva inestable, un ejemplo de esto es él famoso ping de la muerte que consiste en una malformación mandando paquetes ICMP (Internet Protocol Message Protocol o Protocolo de Mensajes de Control de Internet) con un tamaño mayor a 65535 bytes, no obstante la mayoría de los sistemas operativos a partir de 1998 han solucionado este problema.

Funcionalidad:

DoS es realizado a nivel de red, enviando datagramas cuidadosamente preparados y malintencionados de forma tal que las conexiones de red fallen, otra forma para su realización es a nivel de aplicación, ejecutando órdenes debidamente construidas generando así que el sistema esté saturado por múltiples peticiones. Este ataque puede ser utilizado por cualquier usuario que actúe como cliente, y tenga las herramientas necesarias para hacerlo, el destino del ataque provoca que el servidor DNS no resuelva en el tiempo indicado.

La denegación de servicio es una complicación que puede afectar a cualquier servidor de alguna compañía o individuo conectado a una red con o sin salida a Internet, estos ataques no son muy complicados de realizar pero eso no significa que dejen de ser eficaces contra cualquier equipo.

Metodología:

En este ataque se utilizaran dos scripts, uno desarrollado en bash y el otro desarrollado en perl, ejecutados bajo el sistema operativo de BackTrack 5 por medio de una terminal, el primer script básicamente tiene la función de realizar una conexión exitosa, el segundo script satura con múltiples peticiones a la víctima que es un servidor DNS local previamente instalado y configurado en Ubuntu Server 12.

Entonces, al realizar la denegación de servicio por medio de estos dos programas el efecto provocado se puede ver en la figura 4.1.

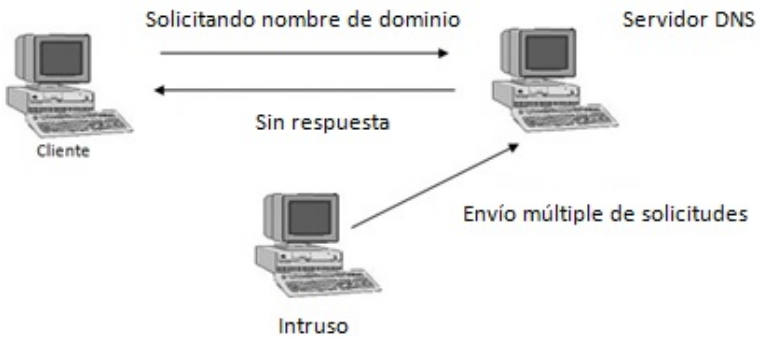

Figura 4.1: Esquema del ataque DoS

El ataque fue implementado en una red local sin acceso a Internet al servidor DNS que lleva como nombre de dominio www.elihu.com, la figura 4.2 muestra cómo es que se logra resolver satisfactoriamente el nombre antes de ejecutar dicho ataque.

Figura 4.2: Resolviendo nombre de dominio

En la figura 4.3 se observa el diagrama de flujo correspondiente a la autenticación mediante la dirección IP de la víctima, mientras que la figura 4.4 muestra el diagrama de flujo cuya función es inundar con múltiples solicitudes al servidor DNS.

A continuación se menciona el proceso a seguir para lograr el éxito deseado en la denegación de servicio [18, 9]:

1. Realizar la conexión con el objetivo, el cuál adquiere la siguiente dirección IP destino 192.168.140.4, para esto se ejecuta el programa en la terminal atendiendo los siguientes parámetros:
   - Nombre del programa: lbd.sh
   - IP destino: 192.168.140.4
   - Modo de ejecución ./lbd.sh 192.168.140.4
2. Saturar el servidor DNS elaborando y mandando múltiples solicitudes, para esto se ejecuta el programa en la terminal atendiendo los siguientes parámetros:
   - Nombre del programa: slowloris.pl
   - Argumento: dns
   - IP destino: 192.168.140.4
   - Modo de ejecución ./slowloris.pl -dns 192.168.140.4

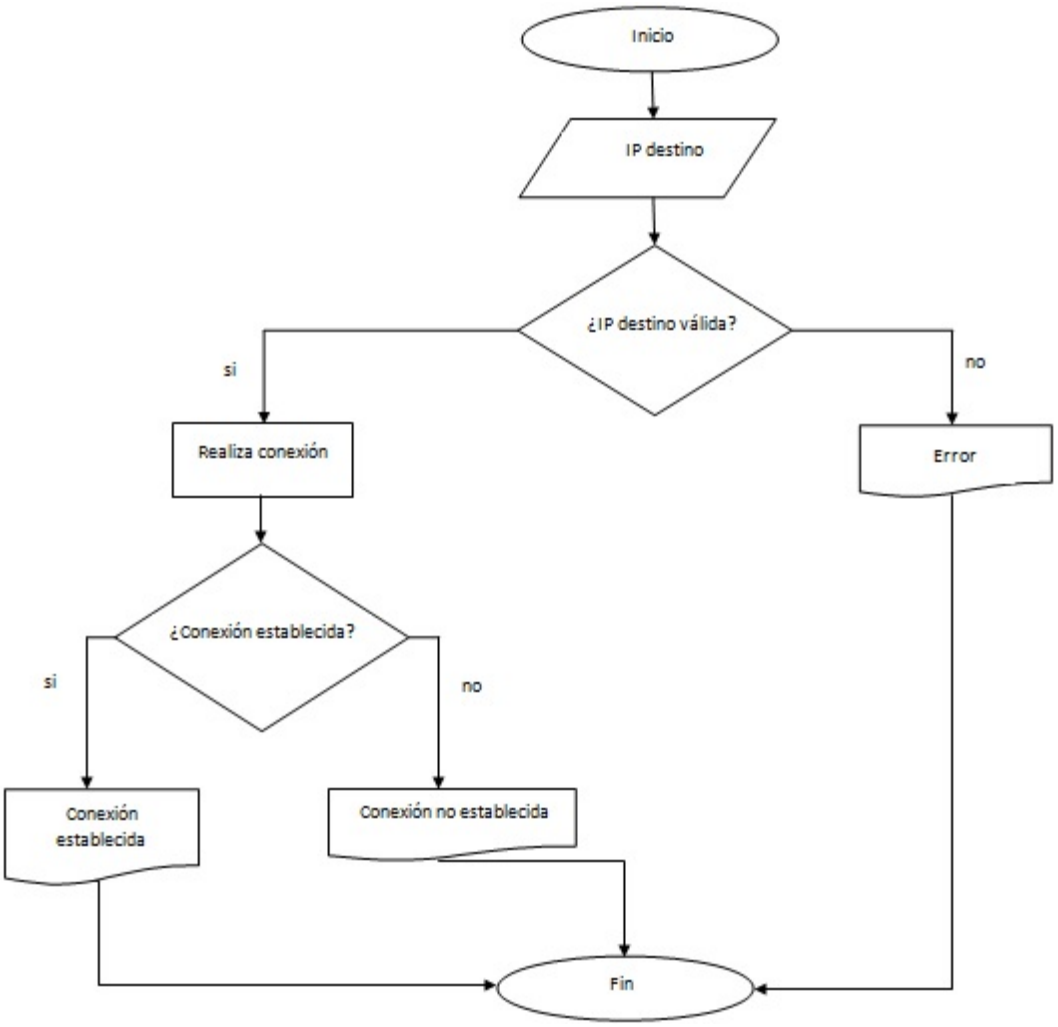

Figura 4.3: Diagrama de flujo para desarrollar la autenticación mediante la dirección IP de la víctima

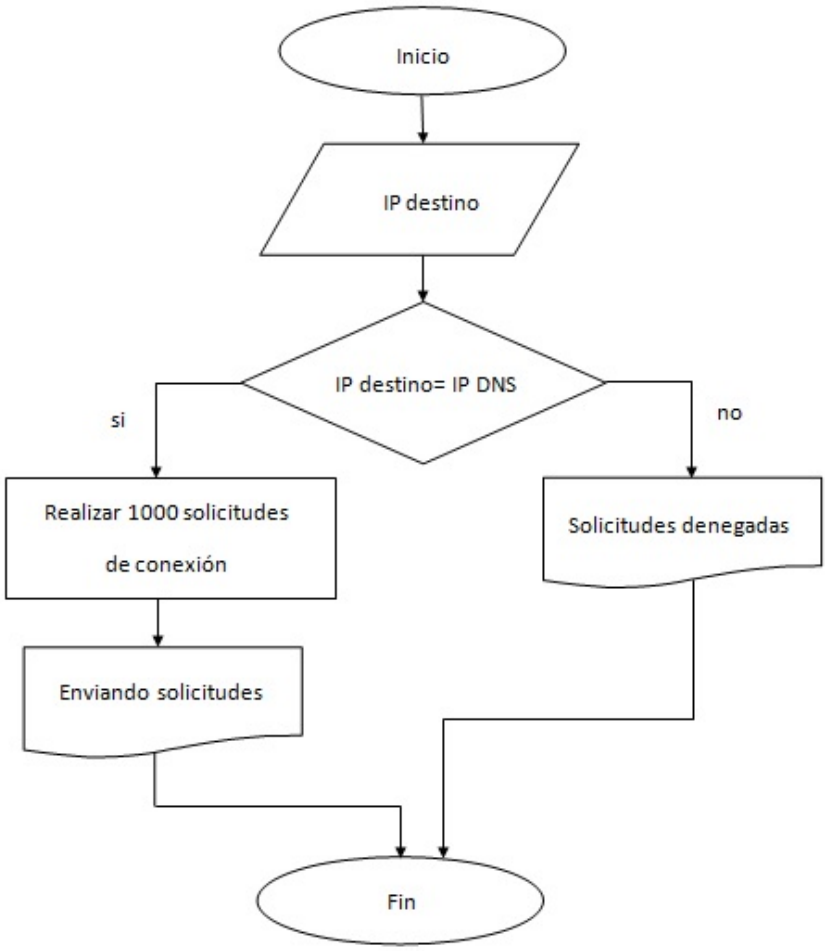

Figura 4.4: Diagrama de flujo para la ejecución de múltiples solicitudes al servidor DNS

Resultados:

El resultado final sobre la saturación de las solicitudes se ve reflejada en la figura 4.5, en donde el servicio no tiene el funcionamiento apropiado, provocando un retraso de cinco minutos para su uso. En total se realizaron 20 pruebas, obteniendo una efectividad de un 80 %, la tabla 4.1 muestra la efectividad de cada evento.

Figura 4.5: Resultado del ataque DoS

Medición	Éxito	Fracaso	Medición	Éxito	Fracaso
1	x		**11**		x
2	x		**12**	x	
3		x	**13**	x	
4	x		**14**	x	
5		x	**15**	x	
6		x	**16**	x	
7	x		**17**	x	
8	x		**18**	x	
9	x		**19**	x	
10	x		**20**	x	
Total	7	3	**Total**	9	1

Resultado: Éxitos 16 Fracasos 4

Tabla 4.1: Mediciones del ataque DoS

4.3. Escaneo de puertos

Este es un ataque pasivo ya que hace un monitoreo del estado de los puertos, el objetivo radica en saber que puertos están abiertos o cerrados para aprovechar las vulnerabilidades que pudieran presentar los servicios que ofrecen, además se puede obtener información más completa sobre algún host en particular.

A continuación se presenta una clasificación de los tipos de escaneo más comunes y utilizados:

1. Reverse Ident (TCP): Consiste en un escaneo normal TCP, con la finalidad de observar si el administrador de los servicios tiene bien configurado su sistema.

2. Ping Scan: Su función principal es averiguar que máquinas están disponibles y responden con el paquete ICMP.

3. Bounce Track (vía FTP): Aprovecha la conexión del FTP, utilizando una IP y un puerto, interceptando una transmisión de datos por terceras personas.

4. UDP Scan: Su función es encontrar aquellos puertos abiertos, que utilizan el protocolo UDP.

5. ACK Scan: Muestra que tipo de configuración existe en algún sistema, además permite determinar si el tráfico ha sido filtrado o no.

6. Windows Scan: Muestra los puertos que están abiertos.

7. Null Scan: Trata de recabar información de forma sigilosa, dejando la menor cantidad de rastro posible.

8. Xmas Scan: Se realiza este tipo de escaneo con paquetes TCP, previamente configurados para nuestro beneficio y con banderas.

9. Idle Scan: Es un escaneo sigilosono que tiene la principal característica de no enviar ningún paquete con nuestra dirección IP, además se le puede responsabilizar del escaneo a otro host sin que haya intervenido de forma alguna.

10. RCP Scan: Su función es enviar un comando a los puertos TCP o UDP abiertos, para conocer que programas se están utilizando.

11. TCP Connect: Intenta establecer una conexión con algún host remoto con un puerto que tiene que estar abierto.

12. TCP SYN: Escaneo que tiene la condición de que no se establecerá una conexión completa, esto es con el objetivo de observar si el host en realidad está conectado o desconectado.

13. Stealth Scan: La finalidad de este escaneo es enviar paquetes a la víctima, y si los ignora quiere decir que el puerto esta abierto [22].

Funcionalidad:

Se realizara un escaneo de puertos, para contemplar que servicios arroja la red a la que está conectado y cuales están más vulnerables, y para esto se lanza una serie de instrucciones de mapeo a las víctimas arrojando información de los puertos que tiene abiertos y cerrados, como la dirección IP, dirección MAC, el tipo de modem al que está conectado, el sistema operativo de cada host y algunos otros datos dependiendo como realicemos la instrucción de escaneo.

Metodología:

Para realizar la exploración de puertos se utilizá el programa de Nmap, cuando se ejecuta se indica el rango de direcciones IP en las que se quiere escanear, luego sólo se deja trabajando por un momento, puede tardar de 1 a 5 minutos o más dependiendo de la instrucción que se haya generado, ya que puede solamente escanear las direcciones IP de los equipos, o incluso arrojar detalles adicionales a estos, al final se obtiene el resultado del mapeo con las propiedades que le hayamos solicitado.

Por lo tanto al realizar el mapeo de los puertos a través de la aplicación Nmap, el efecto esperado se puede observar en la figura 4.6.

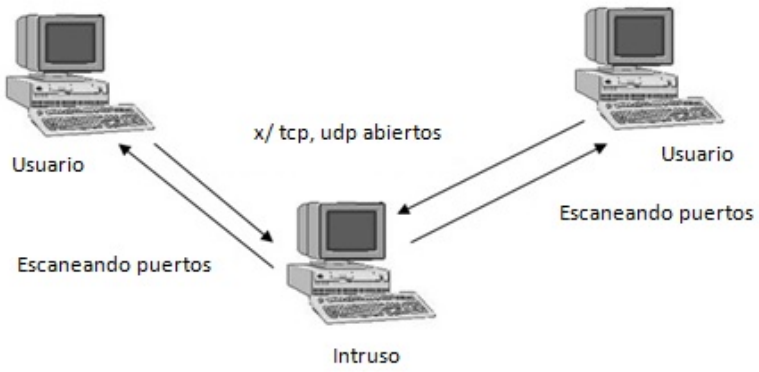

Figura 4.6: Esquema del mapeo de puertos

La exploración del estado de los puertos, fue implementado en una red local con acceso y sin acceso a Internet logrando los mismos resultados en ambos, tomando como

punto de referencia el diagrama de flujo mostrado en la figura 4.7, que nos presenta la secuencia a seguir para obtener la funcionalidad deseada sobre Nmap.

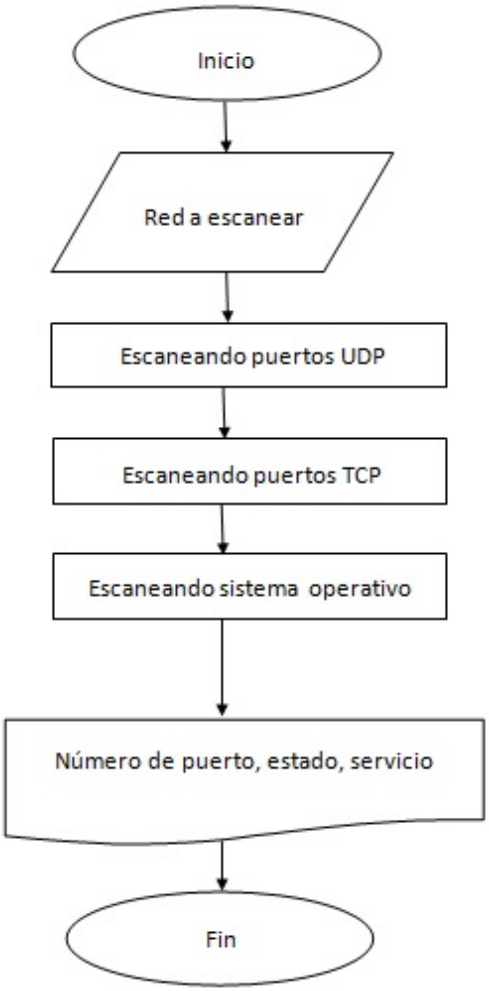

Figura 4.7: Diagrama de flujo para desarrollar el escaneo de puertos

El mapeo se realiza utilizando dos computadoras, en una de ellas se ejecuta el programa Nmap y la otra hace la función de cliente que tiene la dirección IP correspondiente a 192.168.1.64, a continuación se menciona el proceso para lograr el éxito deseado en el escaneo de puertos:

1. Nombre del programa: nmap
2. Argumento 1: sU, escanea puertos UDP.
3. Argumento 2: sT, escanea puertos TCP.

4. Argumento 3: O, identifica el sistema operativo.

5. Argumento 4: sP, identifica las direcciones IP en la red.

6. Red: 192.168.1.0/24

7. Modo de ejecución: nmap -sU -sT -O -sP 192.168.1.0/24

Resultados:

El resultado final sobre el escaneo de puertos se ve reflejado en la figura 4.8, en donde se observa el número de puerto, el estado que presenta, el servicio que arroja dicho puerto, los equipos que están conectados a la red, entre otras características como lo es el sistema operativo y la dirección MAC. En total se realizaron 20 pruebas, obteniendo una efectividad de un 100 %.

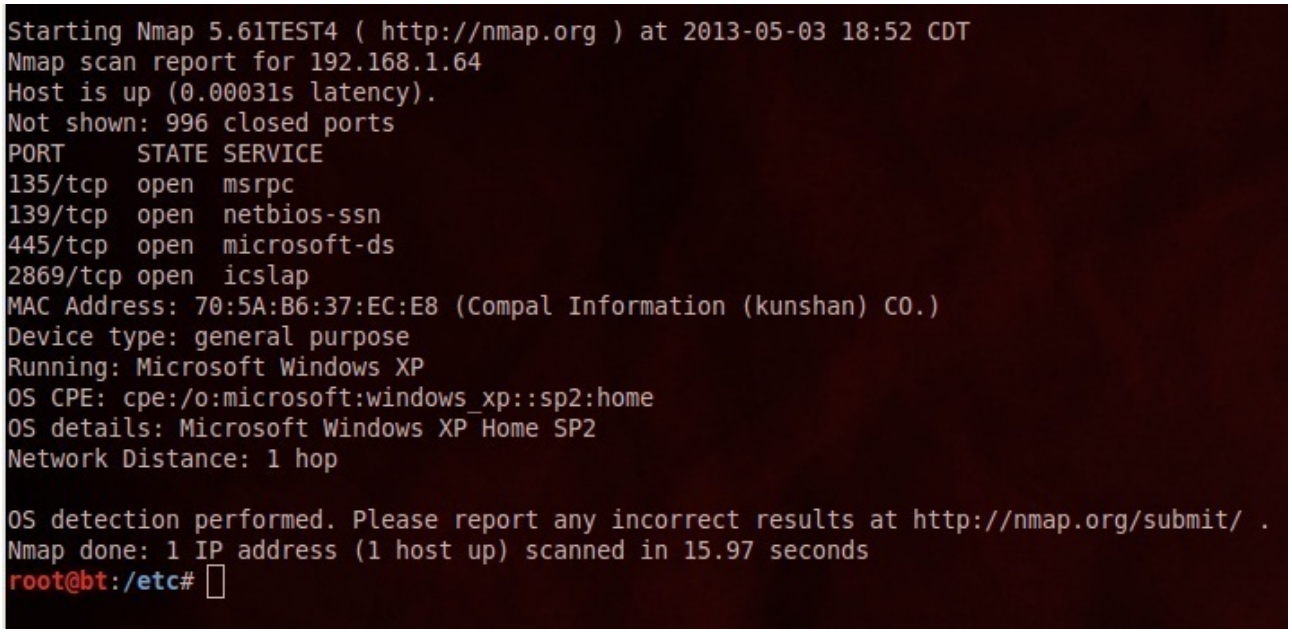

Figura 4.8: Escaneo de puertos en red local

4.4. DNS Spoofing

Es un ataque activo y su función principal es poder simular acciones provenientes de un intruso, consiste en el uso de técnicas de suplantación de identidad generalmente para

fines maliciosos o de investigación, esta acción puede ser realizada por cualquier usuario, a cualquier servidor de nombres de dominio. Cabe resaltar que el objetivo primordial es darle énfasis al DNS Spoofing, pero existen otras clasificaciones de Spoofing que se mencionarán a continuación:

1. DNS Spoofing: Se refiere a la suplantación de identidad por algún nombre de dominio, y consiste en resolver una dirección IP falsa, o nombre de dominio ante una consulta generada. El atacante se hace pasar por un equipo inocente siguiendo el rastro de las direcciones IP y una vez que se localiza un objetivo víctima, se substituye la dirección IP origen de algún paquete TCP/IP por otra dirección IP a la cual se desea que el usuario utilice, por lo tanto el resultado que se obtendrá en pantalla puede ser otro sitio totalmente diferente del que buscaba o la copia exacta del sitio que esperaba visitar [18].

2. GPS Spoofing: Tiene la intención de engañar a un receptor GPS, mediante la transmisión de una señal ligeramente más potente que la recibida desde los satélites del sistema GPS, estas señales son modificadas con la intención de que el receptor determine una posición diferente a la real, que es determinada por el atacante [13].

3. Suplantación de identidad por falsificación de tabla ARP: Consiste en construir tramas que generen una solicitud y una respuesta ARP (Address Resolution Protocol o Protocolo de resolución de direcciones) para traducir direcciones IP a direcciones MAC modificadas, de tal forma que el objetivo envíe los paquetes al host atacante en lugar de hacerlo con su destino legítimo.

4. Mail Spoofing: Consiste en la suplantación de identidad mediante el uso de correo electrónico ajeno, es decir cuando alguien envía e-mails desde la dirección de correo de otra persona [20].

Funcionalidad:

Para el caso de DNS Spoofing, cuando se quiere acceder a un nombre de dominio como por ejemplo www.dominio.com, el atacante puede lograr engañar al usuario, de

modo tal que resuelva ese mismo nombre de dominio en su equipo, pero no con la dirección IP correcta, y por lo tanto el usuario pensará que está en el sitio correcto.

Metodología:

Para elaborar este ataque se obtiene la dirección IP de la víctima, posteriormente se ejecuta el programa llamado SET (Social-Engineer Toolkit) y ettercap colocando el dominio que se quiere suplantar, para que cuando el cliente quiera acceder a la página de su preferencia, observe el mismo sitio al que esta acostumbrado pero con la dirección IP del atacante [11]. La Figura 4.9 muestra cual es el efecto tras utilizar el ataque de DNS Spoofing.

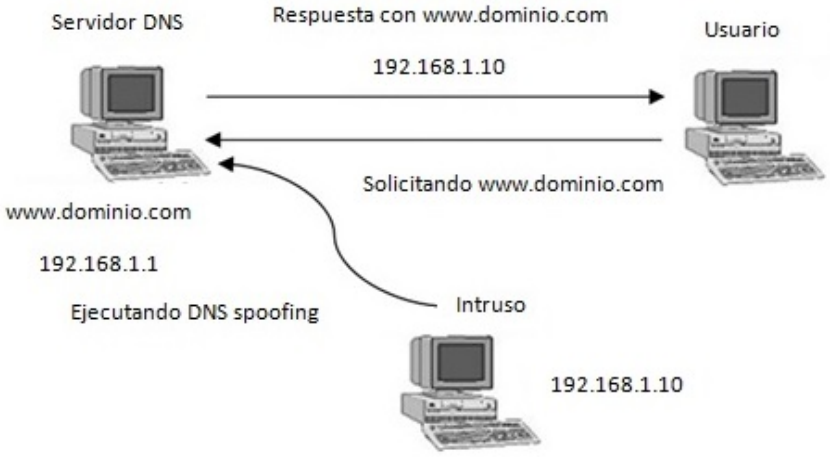

Figura 4.9: Esquema del ataque DNS Spoofing

El ataque fue implementado en una red local con acceso a Internet, sobre el mismo equipo que tiene instalada la aplicación de SET y ettercap con la siguiente dirección IP 192.168.222.128, en la Figura 4.10 se muestra el diagrama de flujo correspondiente a los pasos que se deben seguir para la realización de DNS Spoofing.

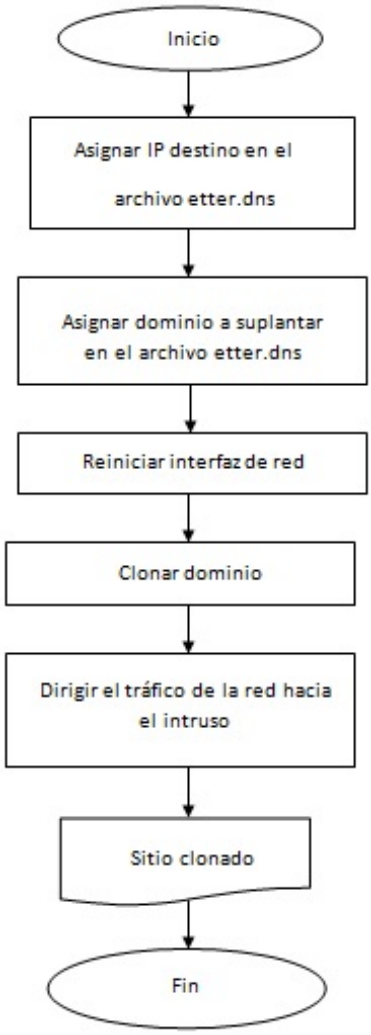

Figura 4.10: Diagrama de flujo para desarrollar el ataque DNS Spoofing

A continuación se menciona el proceso a seguir para lograr el éxito deseado en la suplantación del dominio:

1. Nombre de los programas: SET y ettercap

2. Configurar el archivo etter.dns ubicado en la ruta /usr/local/share/ettercap y

agregar la IP destino: 192.168.222.128

3. Configurar el archivo etter.dns y agregar el dominio a suplantar: http://gmail.com
4. Reiniciar la interfaz de red: /etc/init.d/networking restart
5. Ejecutar SET ubicado en la ruta /pentest/exploits/set: ./set
6. Elegir la opción: Social -Engineering Attacks
7. Elegir la opción: Site Cloner
8. En otra terminal ejecutar ettercap con los siguientes argumentos:
   - Argumento 1: -T, ejecución en modo texto.
   - Argumento 2: -q, oculta algunos paquetes en hexadecimal.
   - Argumento 3: -M arp:remote, desvía el tráfico hacia el equipo del intruso.
   - Argumento 4: -P dns_spoof, indica que sólo el tráfico provocado por el nombre de dominio suplantado será desviado hacia el equipo del intruso.
   - Argumento 5: //, agrega al ataque sólo la dirección IP destino o de la víctima.
   - Modo de ejecución: ettercap -T -q -M arp:remote -P dns_spoof //

Resultados:

Para observar el resultado final sobre la suplantación de dominio, se realizó una comparación del antes y después del mencionado ataque, contemplando el efecto en el navegador, antes de generar el ataque no puede resolver dicha dirección la Figura 4.11 lo demuestra.

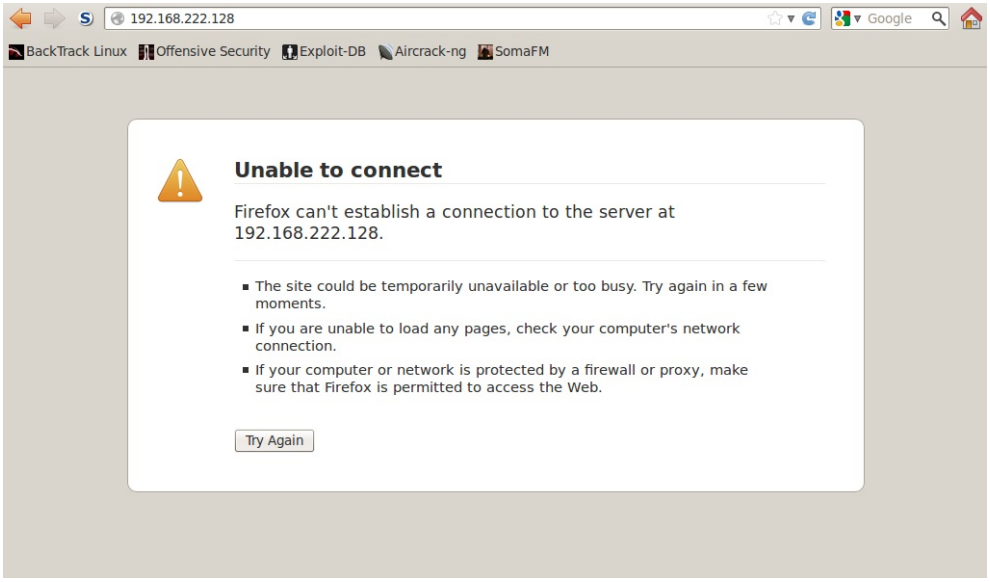

Figura 4.11: Resultado en el navegador antes de generar el ataque DNS Spoofing

El efecto que toma después de la implementación del ataque en el navegador es satisfactorio ya que trata de resolver con la misma dirección IP utilizada, el resultado fue una copia exacta de gmail, tal como lo muestra la figura 4.12. En total se realizaron 20 pruebas, obteniendo una efectividad de un 75 %, la tabla 4.2 muestra la efectividad de cada evento.

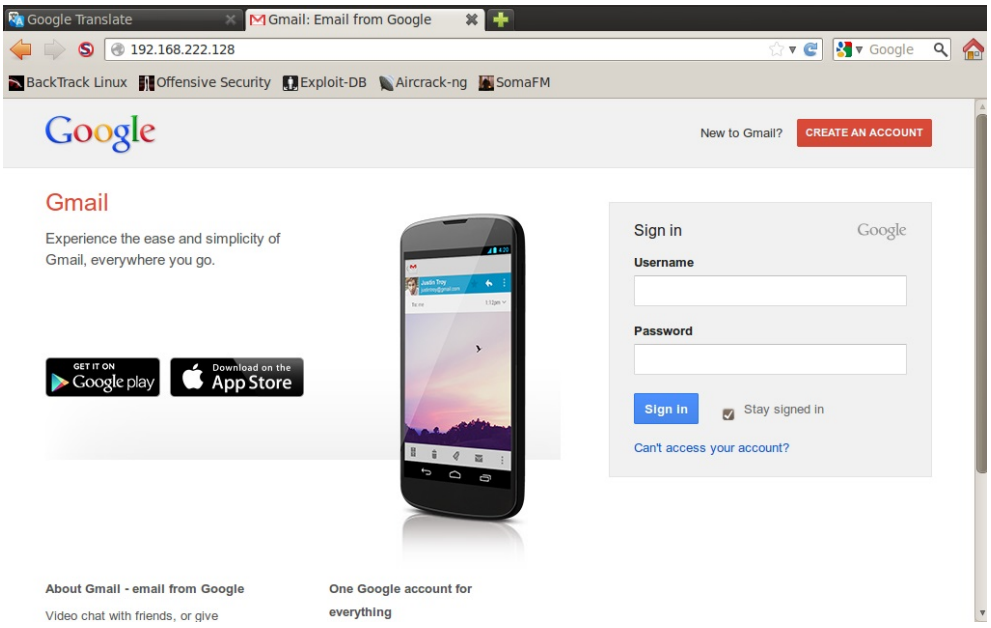

Figura 4.12: Resultado en el navegador después de generar el ataque DNS Spoofing

Medición	Éxito	Fracaso	Medición	Éxito	Fracaso
1	x		**11**		x
2	x		**12**		x
3	x		**13**	x	
4	x		**14**	x	
5		x	**15**		x
6	x		**16**	x	
7	x		**17**	x	
8	x		**18**	x	
9	x		**19**	x	
10	x		**20**		x
Total	9	1	**Total**	6	4
Resultado: Éxitos 15 Fracasos 5					

Tabla 4.2: Mediciones del ataque DNS Spoofing

4.5. Phishing

El término phishing proviene de la palabra inglesa fishing (pesca), que hace alusión al intento de provocar que los usuarios piquen en el anzuelo, este es un ataque activo, además el término phishing es la contracción de password harvesting fishing (cosecha y pesca de contraseñas).

Este tipo de ataque se remonta a los años 90Žs, donde America Online (AOL) solía proporcionar un servicio de cuentas para usuarios como el número de tarjetas de crédito. El phishing consistía en que el atacante se hacía pasar por algún empleado de AOL enviando un mensaje instantáneo a una víctima, para poder engañarla de modo que accediera a revelar su información, el mensaje podía contener textos como verificando su cuenta o confirmando información de factura, y una vez que el usuario enviaba su contraseña, el atacante podía tener acceso a la cuenta de la víctima y utilizarla para diversos propósitos.

Con el paso del tiempo AOL reforzó su política respecto al phishing y los atacantes

fueron expulsados de los servidores de AOL, pero a pesar de esto, la otra medida a la cual recurrieron fue añadir en su sistema de mensajería instantánea, un breve texto diciendo: nadie que trabaje en AOL le pedirá su contraseña o información de facturación. Posteriormente AOL desarrolló un sistema que desactivaba de forma automática una cuenta involucrada en phishing antes de que la víctima pudiera responder [7].

Funcionalidad:

Para realizar un ataque de este tipo, no es necesario tener acceso al equipo físicamente para atacar a la víctima, ya que puede ser implementado de forma local con y sin acceso a Internet, básicamente el phishing consiste en duplicar una página web para hacer creer al usuario que se encuentra accediendo a la página original, además de que está alojada en un servidor diferente al real, controlando así los datos que se reciban por el atacante.

Metodología:

Para realizar el ataque de phishing se tiene contratado un plan de hosting (alojamiento) más dominio, posteriormente se diseña una página web falsa para la víctima, la cual está configurada para guardar usuarios y contraseñas por medio de los campos de texto, para que el intruso reciba en su correo electrónico estos datos. Además de estar alojada en el plan de hosting, se genera un subdominio con el nombre de facebook.dklase.com, para acercarse más a la realidad y la víctima crea más en la veracidad de la página. La figura 4.13 muestra el efecto provocado del mencionado ataque, mientras que en la figura 4.14 se observa un diagrama de flujo correspondiente a la serie de pasos a seguir para la realización del ataque Phishing.

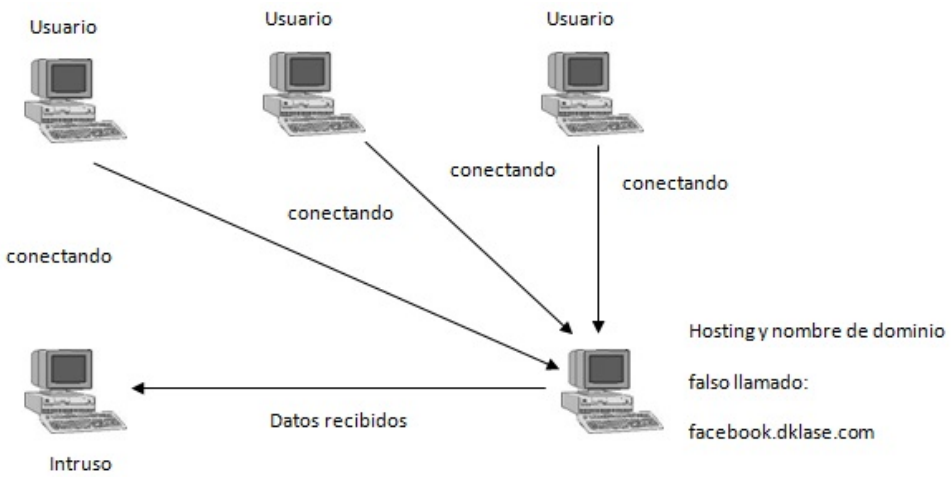

Figura 4.13: Esquema del ataque Phishing

Resultados:

Después de diseñar y alojar la página web falsa, adquiriendo el siguiente nombre de dominio www.facebook.dklase.com, el resultado será que cualquier usuario que cargue la página la observará como lo muestra la figura 4.15.

Posteriormente cuando la víctima acceda y coloque su nombre de usuario y contraseña, la página arrojará el siguiente mensaje: El servicio ha sido desabilitado temporalmente debido a reparaciones en el sitio favor de esperar. La figura 4.16 nos da detalle de esto.

Por último los datos colocados en los campos de texto, se reciben en la bandeja de entrada del intruso, como lo muestra la figura 4.17.

En total se realizaron 20 pruebas, obteniendo una efectividad del 100 %, siempre y cuando el usuario en realidad crea certeramente que está en la página correcta.

En resumen cada una de ataques se puede enteder como lo expresa la tabla 4.3.

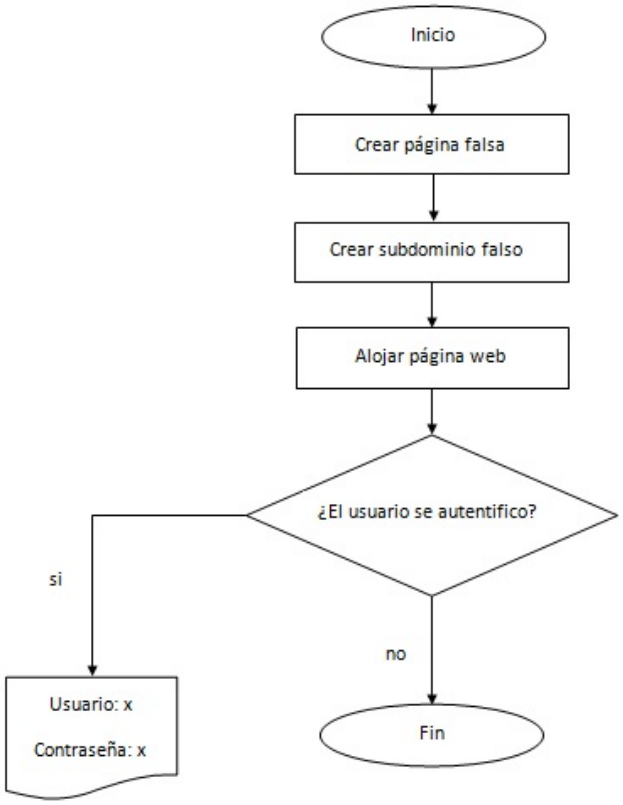

Figura 4.14: Diagrama de flujo para desarrollar el ataque Phishing

Ataque	Función
DoS	Provoca el retraso de algún programa o servicio
Escaneo de puertos	Obtiene la cantidad de puertos abiertos o cerrados
DNS Spoofing	Genera una suplantación de identidad por un nombre de dominio
Phishing	Engaña al usuario para revelar información través de un sitio falso

Tabla 4.3: Tipos de ataques

Figura 4.15: Cargando página falsa

Figura 4.16: Mensaje de página falsa

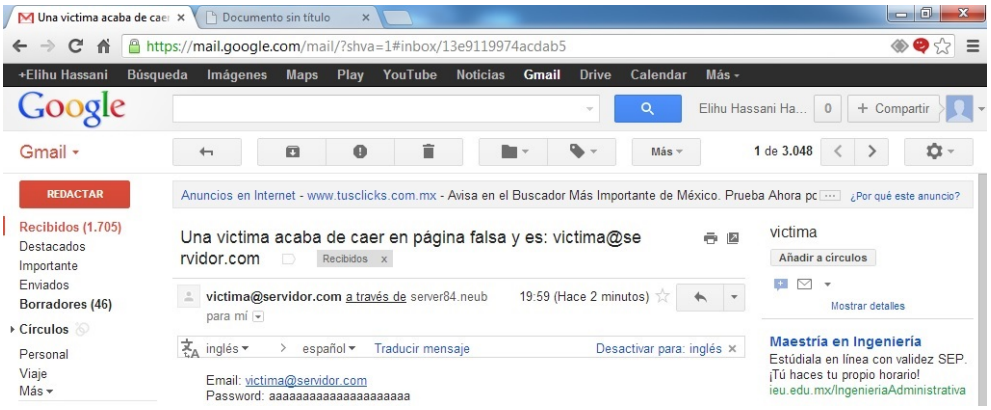

Figura 4.17: Obteniendo usuario y contraseña del usuario que se a utenticado

Capítulo 5

Técnicas de prevención ante ataques a DNS

Debido a los ataques expuestos en el capÃtulo cuatro, y las consecuencias que se pueden originar ante una deficiente organización y configuración de los servicios, en este capítulo se abordan algunas tÃ©cnicas para reducir este tipo de ataques, logrando que el servidor de nombres de dominio, consiga un funcionamiento adecuado realizando todas aquellas funciones para lo cual está diseñado.

5.1. DoS

Anteriormente se ha mencionado como es que afecta la denegación de servicio, en especial para el servidor DNS provocando que no resuelva ante una petición de algún cliente. Para contrarrestar este ataque, es posible evitar todas las solicitudes que el atacante esté enviando para saturar el tráfico en la red y dejar deshabilitado el servidor de nombres de dominio, utilizando los programas denominados *libapache2-mod-evasive* y *libapache-mod-security*, instalados en Ubuntu Server 12. La manera de hacerlo se describe a continuación.

Requisitos:

1. Instalar el programa libapache2-mod-evasive en el equipo donde se encuentra

configurado el servidor DNS.

2. Instalar el programa libapache-mod-security en el equipo donde se encuentra configurado el servidor DNS.

3. Utilizar las reglas descargadas del programa modsecurity-apache_2 en el equipo donde se encuentra configurado el servidor DNS.

Metodología [16, 21]:

1. Crear un directorio en la ruta `/var/log`: mkdir mod_evasive

2. Abrir el archivo *modevasive* ubicado en la ruta `/etc/apache2/conf.d`.

3. Agregar los siguientes parámetros:

   - DOSHashTableSize 3097: Establece el número de nodos a almacenar para cada proceso de peticiones de la tabla hash (contenedor asociativo de recuperación de peticiones por medio de claves que agiliza las respuestas del servidor).
   - DOSPageCount 2: Indica el número de peticiones de una misma página, cuando este valor se excede, la IP del intruso se añade a la lista de bloqueos.
   - DOSSiteCount 50: Cuenta las peticiones de cualquier tipo que puede hacer un cliente, dentro del intervalo definido en DOSSiteInterval.
   - DOSPageInterval 1: Genera un intervalo en segundos, para el mínimo número de peticiones de las páginas.
   - DOSSiteInterval 1: Genera un intervalo en segundos, para el número mínimo de peticiones de objetos de cualquier tipo.
   - DOSBlockingPeriod 10: Establece el tiempo en segundos, sobre un intruso que ha sido bloqueado, una vez que ha sido añadido a la lista de bloqueos.
   - DOSEmailNotify correo@dominio.com: Un correo electrónico será enviado a la dirección especificada cuando una dirección IP quede bloqueada.

- DOSWhitelist 127.0.0.1: Es la dirección IP que queda excluida dentro de los demás parámetros para no ser tomada en cuenta.

4. Crear un directorio en la ruta `/etc/apache2/`: mkdir mod_security_rules
5. Mover las reglas de *modsecurity-apache_2* al fichero *mod_security_rules*, que está ubicado en la ruta `/etc/apache2/`
6. Abrir el archivo *modsecurity* ubicado en la ruta `/etc/apache2/conf.d`
7. Agregar las siguiente línea:
   Include mod_security_rules/*.conf: Indica que se anexarán todas las reglas en el archivo *modsecurity.*
8. Reiniciar el programa de libapache2-mod-evasive: a2enmod mod-evasive
9. Reiniciar el programa de libapache-mod-security: a2enmod mod-security
10. Reiniciar apache: /etc/init.d/apache2 restart

Resultados:

1. Evita el envío múltiple de solicitudes de conexión al servidor DNS.
2. Permite dar una respuesta satisfactoria ante cualquier petición de un cliente, con el nombre de dominio.

En la figura 5.1 se puede observar el efecto que se adquiere tras haber realizado esta configuración, en donde la computadora del atacante trata de realizar la denegación de servicio, sin embargo ha fracasado porque no se puede enviar ningún paquete para saturar el tráfico.

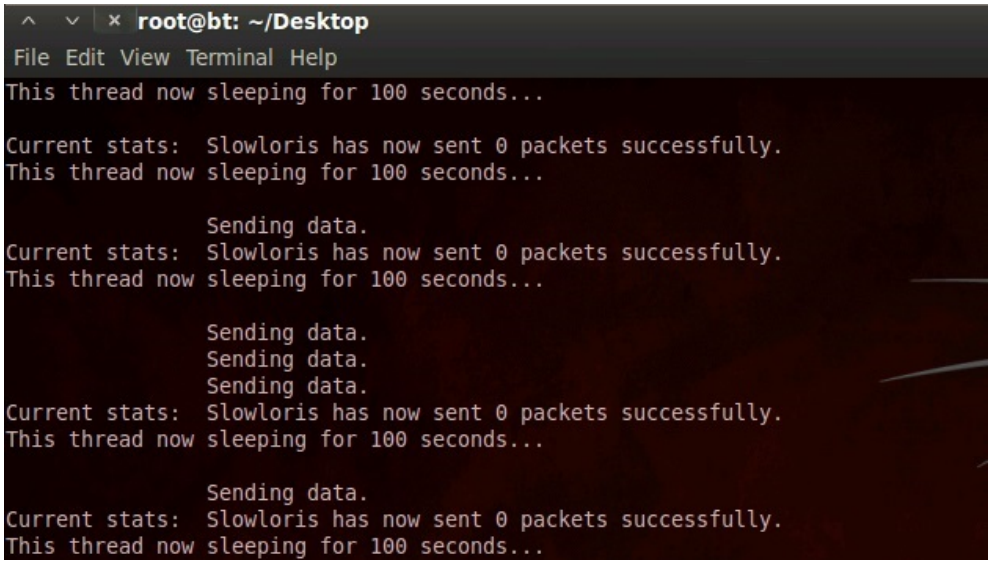

Figura 5.1: Evitando denegación de servicio

Mientras tanto el servidor DNS, sigue estando en funcionamiento, resolviendo las peticiones que se le solicitan, evitando así el retraso que se tenía anteriormente tras realizar el ataque, la figura 5.2 nos muestra el resultado.

Figura 5.2: Funcionamiento estable del servidor DNS

5.2. Escaneo de puertos

El escaneo de puertos ayuda al intruso a revelar información sobre el estado de la red y los servicios a los que puede afectar. Para contrarrestar este ataque, es posible detectar las direcciones IP de los intrusos utilizando el programa denominado *portsentry* y configurando el archivo hosts.deny que se encuentra en el sistema operativo Ubuntu Server 12, para que con esta configuración se bloquee la conexión de los intrusos en la

red. La manera de hacerlo se describe a continuación.

Requisitos:

1. Instalar el programa portsentry en el equipo donde se encuentra configurado el servidor DNS

2. Localizar el archivo hosts.deny en el sistema operativo Ubuntu Server.

Metodología:

1. Abrir el archivo portsentry.conf ubicado en la ruta `/etc/portsentry/`

2. Modificar los siguientes parámetros: BLOCK_UDP='0' y BLOCK_TCP='0' por BLOCK_UDP='1' Y BLOCK_TCP='1', lo que significa que se está bloqueando el escaneo de puertos UDP y TCP.

3. Identificar las direcciones IP que se consideran como intrusos.

4. Abrir el archivo de configuración hosts.deny, ubicado en la ruta /etc:,

5. Agregar al final de ese archivo la regla:
   `ALL Red_permitida EXCEPT IPs_para_denegar`
   Indica la denegación sobre las conexiones de aquellos intrusos que estén escaneando la red.

6. Detener el servicio de portsentry: /etc/init.d/portsentry stop

7. Iniciar el servicio de portsentry: /etc/init.d/portsentry start

Resultados:

1. Evita la visualización de los puertos con sus respectivos servicios.

2. Bloquea la conexión de los intrusos

El efecto resultante tras haber realizado esta configuración, se puede observar en la utilización de dos equipos, el equipo A, que pertenece al intruso y posee la dirección IP de 192.168.1.2 y el equipo B, que pertenece al servidor de nombres de dominio y posee la dirección IP 192.168.1.1. La figura 5.3 muestra cómo el intruso trata de escanear el estado de los puertos del equipo B, pero el resultado que obtiene no es satisfactorio y no puede observar ningún puerto en pantalla el equipo A.

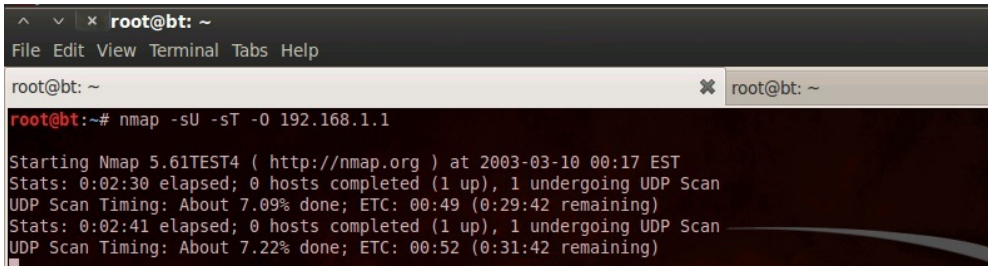

Figura 5.3: Escaneo de puertos denegado

Por otro lado, lo que sucede con el equipo B, en donde se realizó la configuración de seguridad, es que detecta quién trato de escanear los servicios que están corriendo en sus puertos y se agregan todas aquellas IPs, a la lista de configuración del archivo hosts.deny como lo muestra la figura 5.4, para bloquear así todas aquellas conexiones de intrusos de forma automática.

```
ALL: 192.168.1. EXCEPT 192.168.1.2

ALL: 192.168.1.2 : DENY
ALL: 192.168.1.3 : DENY
```

Figura 5.4: Denegando conexiones de intrusos

Mientras tanto también se puede comprobar, que ya no existe ningún tipo de conexión con el equipo B, ya que se envía un ping desde el equipo A y este no puede recibir respuesta de manera satisfactoria tal como lo muestra la figura 5.5

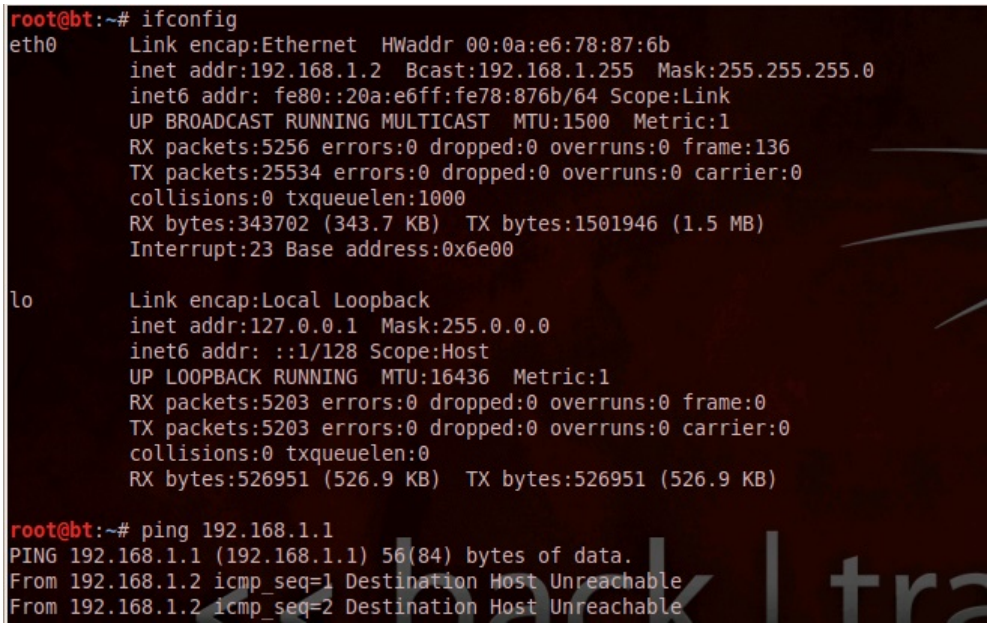

Figura 5.5: Sin respuesta de conexión

5.3. DNS Spoofing

Como observo en el capítulo anterior, el DNS Spoofing realiza una suplantación de algún nombre de dominio, por la dirección IP del atacante, provocando que todo el tráfico se desvíe hacia dicha dirección. Para contrarrestar este ataque, es necesario configurar el archivo *host.conf* que se encuentra en el sistema operativo de BackTrack 5 y la manera de hacerlo se describe a continuación.

Requisitos:

1. Localizar el archivo host.conf en el sistema operativo BackTrack 5.

Metodología [23]:

1. Encontrar el archivo host.conf con el comando *locate*
2. Abrir el archivo host.conf ubicado en la ruta */etc*
3. Agregar la siguiente instrucción que indica una resolución inversa para impedir falsificaciones de direcciones IP: *nospoof on*

4. Reiniciar el servicio DNS: *service named restart*

Resultados:

1. Evita la suplantación del nombre de dominio

El resultado final muestra que el usuario puede evitar este tipo de ataque, donde inicialmente, se logró realizar la suplantación de IP con el nombre de dominio de gmail.com, pero ahora la figura 5.6 demuestra que después de realizar la configuración correspondiente, ya no se puede realizar satisfactoriamente el ataque.

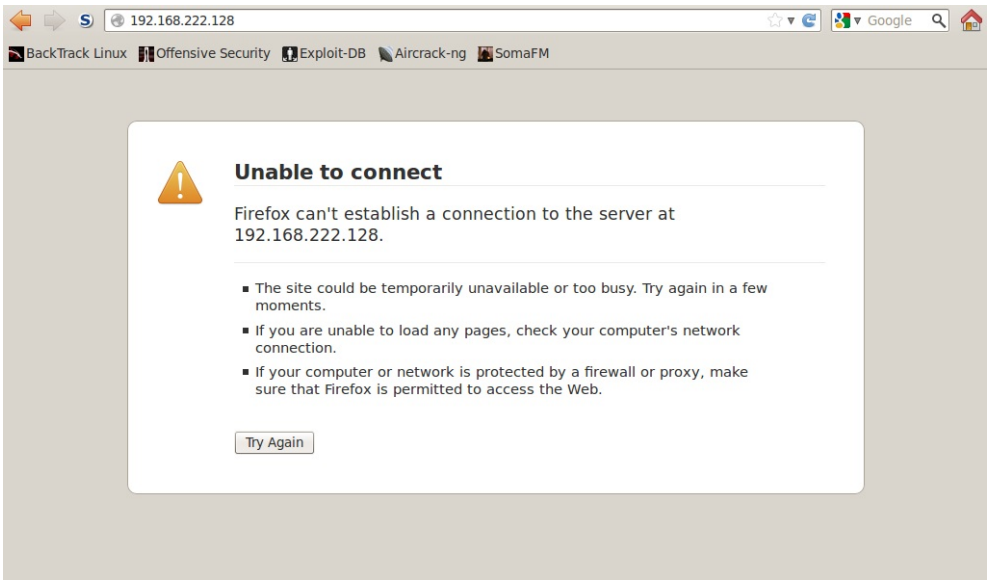

Figura 5.6: Evitando DNS Spoofing

5.4. Phishing

Anteriormente hemos visto esta modalidad de ataque, que permite obtener los nombres de usuarios y contraseñas mediante un engaño, pero para tomar las medidas correspondientes ante este ataque, es necesario saber cómo identificar la presencia del phishing, y la forma de hacerlo es como se lista a continuación [8]:

1. El diseño web implementado para el ataque en algunos casos tendrá el logo empresarial de cualquier empresa de renombre.

2. Se pueden identificar errores gramaticales y de ortografía.

3. Se incluyen textos dando a entender al usuario que su seguridad está expuesta ante un mal uso, y por lo tanto debe ingresar su usuario y contraseña para verificar que están bien los datos.

4. Los botones del diseño de la página web falsa pueden ser imágenes en lugar de los originales, esto para acercarse más a la apariencia del sitio original.

Metodología:

1. Cerciorarse si la página utiliza https (Hypertext Transfer Protocol Secure o Protocolo Seguro de Transferencia de Hypertexto) además de que aparezca el símbolo del candado con el certificado digital.

2. No acceder a páginas web comerciales, financieras o bancarias provenientes de algún enlace originado por correo electrónico ya que la dirección podría no ser la correcta.

3. No introducir datos confidenciales en caso de desconocer el origen legítimo de los mensajes arrojados por el sistema falso.

4. Verificar que realmente sea el origen correcto del nombre de dominio que nos brinda sus servicios.

Resultados:

El resultado final consiste en sólo visitar aquellos sitios, que realmente sean legítimos y la prueba de la veracidad de estos se puede corroborar como lo muestra la Figura 5.7, que es un claro ejemplo donde se observa que se utiliza https y el símbolo del candado con el certificado digital.

Figura 5.7: Pagina web legítima

Capítulo 6

Conclusiones y trabajo a futuro

En este trabajo se presentó un listado de las vulnerabilidades de los servidores de nombres de dominio, para lo cual fue necesario conocer los conceptos básicos de una red, su clasificación de acuerdo a su magnitud y servicio, los modelos OSI y TCP/IP, la forma de la conectividad y la correcta comunicación a través del protocolo de comunicación IP.

Además, se realizó un análisis de la funcionalidad de los DNS, mostrando el formato válido para estos y su clasificación de acuerdo al territorio donde se encuentren o respecto a la funcionalidad de la organización. En cuanto a la configuración se mostró la necesidad del uso de registros, además de una resolución directa e inversa, con lo cual, el usuario final observará en su computadora una página web tras haber realizado alguna petición al servidor de dominio de su preferencia.

Como fue planteado en los objetivos, se abordaron aspectos relacionados a la seguridad de los DNS, estudiando, analizando y poniendo en práctica ataques tales como, DoS, escaneo de puertos, DNS Spoofing y Phishing, siendo todos ellos efectivos. Para evitar lo anterior, se analizaron, describieron e implementaron algunas técnicas de defensa contra los ataques, con la ayuda de la configuración correcta de ciertos archivos.

Finalmente se puede concluir que en lo particular el servidor de nombres de dominio, es uno de los más importantes, ya que permite obtener un resultado sobre la petición de cierta página y su correcto funcionamiento debe ser prioritario, ya que no se sabe que tanto nivel de afectación se puede producir ante la ejecución de un ataque, dejando sin servicio a los clientes que quieren obtener alguna información, es por eso que se deben de tomar medidas preventivas que brinden una estabilidad, dejando a un lado la afectación que se pudiera generar tras los ataques, brindando un servicio eficaz y con buenos resultados.

6.1. Trabajo a futuro

Como trabajo a futuro se considerara lo siguiente:

1. Investigar los ataques distribuidos sobre los servidores de nombre de dominio.
2. Proponer técnicas de prevención sobre los ataques distribuidos.
3. Investigar sobre los ataques que afectan a otros servidores, como por ejemplo el servidor de correo y web.
4. Proponer técnicas de prevención a los servidores de correo y web.

Referencias

[1] Estrada, A. (2004). Protocolos TCP/IP de Internet. *Revista Digital Universitaria. 5(8).*

[2] Fernández, A. (2010). *Tutorial de IPv6.* Sitio web: http://www.ipv6.unam.mx/documentos/Tutorial-IPv6-UNAM.pdf.

[3] Perpinan, A. (2006). *Administración de Redes GNU/LINUX.* Santo Domingo-República Dominicana: IMPRESOS GAMMA.

[4] Tanenbaum, A. (2003). *Redes de computadoras.* México: PEARSON Prentice Hall.

[5] Internet Assigned Numbers Authority. (2013). *Root Zone Database.* Sitio web: http://www.iana.org/domains/root/db.

[6] Pecos, D. (2003). *Domain Name Server.* Sitio web: http://danielpecos.com/docs/dns/dns.pdf.

[7] Instituto Nacional de Tecnologías de la Comunicación. (2007). *Estudio sobre usuarios y entidades públicas y privadas afectadas por la práctica fraudulenta conocida como phishing.* Sitio web: http://www.inteco.es/file/rmqBMNKZwCoOKzEyqx15mg.

[8] Seguridad Empresarial. (2013). *Guías de prevención como evitar el pishing.* Sitio web: http://www.seguridadempresarial.com.mx/tips/como_evitar_phishing.pdf.

[9] Chablé, H.M. (2007). *Herramientas de monitoreo y detección de intrusos en servidores Linux.* Centro de Investigación y de Estudios Avanzados del Instituto Politecnico Nacional, D.F, México.

[10] Barceló, J; Íñigo, J; Martí, R; Peig, E; Perramon, X. (2004). *Redes de computadores.*

[11] Coupe, J. (2012). *ARP y DNS Spoofing.* Sitio web: http://josephcoupe.wordpress.com/2012/01/17/10-arp-y-dns-spoofing/.

[12] Palet, J. (2006). *IPv6 Tutorial.* Sitio web: http://archive.icann.org/en/meetings/saopaulo/presentation-ipv6-tutorial-basics-03dec06.pdf.

[13] Wen, H; Yih, P; Dyer, John; Archinal, A; Fagan, J. (2013). *Countermeasures for GPS signal spoofing.* Sitio web: http://www.blockyourid.com/g̃bp-prorg/mil/gps4/Wen_Spoof.pdf.

[14] Moreno, R; Santiago, R; Fabero, J.C. (2013). *Redes.* Sitio web: http://www.fdi.ucm.es/profesor/rmoreno/redes-grado/transparencias/Tema%206-%20Capa%20de%20red%20y%20protocolo-%20IP.pdf.

[15] Betancur, L. (2011). Redes de área corporal. Una perspectiva al futuro desde la investigación. *Revista Sistemas y Telemática. 9(16).*

[16] Linux Log. (2013). *Ubuntu 10.04 - Apache mod_evasive & mod_security.* Sitio web: Sitio web: http://www.linuxlog.org/?p=135.

[17] Incera, J; Cartas, R; Cairó, O. (2007). *Redes Digitales: Presente y Futuro.* Sitio web: http://allman.rhon.itam.mx/j̃incera/IntroRedesDigitales.pdf.

[18] Kirch, O. (1999). *Guía de Administración de Redes con Linux.* Darmstadt, Alemania.

[19] Robles, O. (2003). *¿Qué es el DNS?.* Boletín de Política Informática, 1.

[20] Sun Yat sen University. (2013). *Web Security: Theory & Applications.* Sitio web: http://my.ss.sysu.edu.cn/WebSec/download/chap5.pdf.

[21] Tail-f. (2013). *Instalar mod_evasive en Apache para dificultar ataques DoS*. Sitio web: http://www.tail-f.com.ar/servicios/httpd/apache-httpd-servicios/instalar-mod_evasive-en-apache-para-dificultar-ataques-dos.html.

[22] Nebrija Universidad. (2013). *Técnicas de port scanning y uso del nmap*. Sitio web: http://www.nebrija.es/c̃malagon/seguridad_informatica/Lecturas/2-port_scann1ng_nmap_hxc.pdf.

[23] Linker, V. (2013). *Evitar ataques de IP-Spoofing en el servicio DNS (Centos)*. Sitio web: http://www.vicolinker.net/evitar-ataques-de-ip-spoofing-en-el-servicio-dns-centos/.

[24] Stallings, W. (2000). *Comunicaciones y Redes de computadores*. PEARSON Prentice Hall.