Sustentación lab 2

HTTP

Puerto: 80

Protocolos de transporte

· TCP: Confiable, reconoce los datos, reenvía los datos perdidos, entrega los datos en orden secuencial. Primero se asegura la conexión con el servidor.

· UDP: Rápido, baja sobrecarga, no requiere reconocimiento, no reenvía los datos perdidos, entrega los datos a medida que llegan.

TCP seguro

· SSL (secure socket layer): Provee servicios de cifrado a TCP, asegura la integridad de los datos. No es un protocolo, es una mejora a TCP, que se implementa en capa de aplicación.

· 3-Handshake TCP:

o El dispositivo que desea iniciar la conexión (conocido como "cliente") envía un paquete SYN (Synchronize) al dispositivo de destino (conocido como "servidor").

o El servidor responde con un paquete SYN-ACK (Synchronize-Acknowledge) para confirmar que está disponible y listo para establecer la conexión.

o Finalmente, el cliente envía un paquete ACK (Acknowledge) al servidor para confirmar que ha recibido el mensaje SYN-ACK y establecer la conexión.

o Con SSL: Se hace el 3 handshake, junto con un esquema de conexión para establecer una transmisión segura.

· Cuáles son los protocolos de https: Los protocolos de https son SSL (Secure Sockets Layer) y TLS (Transport Layer Security).

1. **SSL (Secure Sockets Layer)**: certificado
    - SSL es un antiguo protocolo de seguridad diseñado para cifrar las comunicaciones entre un cliente y un servidor.

    - SSL proporciona una capa de seguridad adicional a las conexiones web, protegiendo los datos de ser interceptados o modificados.
2. **TLS (Transport Layer Security)**: Protocolo que intercambia el certificado
    - TLS es el sucesor de SSL y se utiliza ampliamente para cifrar y autenticar las comunicaciones en línea.
    - TLS garantiza que las conexiones web sean seguras y que los datos transmitidos estén protegidos de posibles amenazas.
    - HTTPS es una implementación común de TLS que asegura las transacciones en línea y protege la privacidad de los usuarios.

· En qué puerto va https: El puerto de https es el 443.

· Cuáles son los protocolos de la capa de transporte: Los protocolos de la capa de transporte son TCP (Transmission Control Protocol) y UDP (User Datagram Protocol).

Como funciona el three hand shaking: El three hand shaking es un proceso en el que un cliente y un servidor establecen una conexión TCP. El cliente envía un paquete SYN, el servidor responde con un paquete SYN-ACK, y luego el cliente envía un paquete ACK para confirmar la conexión.

Servidor Web

Comandos http: HTTP (Hypertext Transfer Protocol) es un protocolo de la capa de aplicación que define la forma en que los clientes y servidores web se comunican entre sí. Aunque HTTP en sí mismo no es un lenguaje de programación, existen ciertos comandos y métodos que se utilizan para controlar el intercambio de información. Algunos de los comandos o métodos HTTP más comunes son:

- GET: este método se utiliza para solicitar un recurso específico (como una página web o un archivo) desde el servidor web. El servidor responde enviando el recurso solicitado.
- POST: este método se utiliza para enviar datos al servidor, por ejemplo, cuando se envía un formulario web con información de usuario.
- PUT: este método se utiliza para actualizar un recurso existente en el servidor.
- DELETE: este método se utiliza para eliminar un recurso del servidor.

- HEAD: este método se utiliza para obtener sólo la cabecera de un recurso, sin descargar todo el contenido.
- OPTIONS: este método se utiliza para obtener información sobre las opciones de comunicación disponibles en el servidor.
- TRACE: este método se utiliza para recuperar una copia de un mensaje de solicitud tal como se recibió por última vez en el servidor.
- · ¿Qué es un servidor web y cuál es su función en una red?

Un servidor web es un programa que se ejecuta en un servidor para recibir y responder a solicitudes de recursos web, como páginas web y archivos multimedia.

- · ¿Qué es un servidor proxy y cómo se utiliza en una red?

Un servidor proxy actúa como intermediario entre los clientes y los servidores web para ocultar la dirección IP del cliente y mejorar el rendimiento y la seguridad.

- · ¿Cuál es la diferencia entre un servidor web y un servidor de aplicaciones?

Un servidor de aplicaciones es un programa que se utiliza para ejecutar aplicaciones web, mientras que un servidor web se utiliza principalmente para servir contenido estático (como páginas web y archivos).

- · ¿Qué es un servidor DNS y cuál es su función en una red?

Un servidor DNS es un programa que se utiliza para resolver nombres de dominio en direcciones IP.

- · ¿Qué es la arquitectura cliente-servidor y cómo se aplica en el contexto de los servidores web?

La arquitectura cliente-servidor implica que los clientes solicitan recursos y los servidores responden a esas solicitudes.

- · ¿Qué protocolos utilizan los servidores web y cómo se utilizan?

Los protocolos utilizados por los servidores web incluyen HTTP, HTTPS, FTP, entre otros.

- · ¿Qué es la escalabilidad en el contexto de los servidores web y cómo se puede lograr?

La escalabilidad se refiere a la capacidad de un servidor para manejar un número creciente de solicitudes y se puede lograr mediante la adición de más recursos de hardware o mediante técnicas de equilibrio de carga.

- · ¿Cuáles son las medidas de seguridad que se deben tomar en un servidor web para evitar ataques y vulnerabilidades?

Las medidas de seguridad que se deben tomar en un servidor web incluyen el uso de cifrado HTTPS, la aplicación de parches de seguridad regulares y el uso de herramientas de seguridad como cortafuegos y programas antivirus.

Preguntas del lab

- · ¿Qué información le muestra en pantalla el comando ipconfig /displaydns?

El comando "ipconfig /displaydns" muestra en pantalla la caché de resolución de DNS, es decir, una lista de todos los nombres de dominio recientemente resueltos por el sistema y sus correspondientes direcciones IP.

- · ¿Qué ocurre si desde un PC cliente se intenta hacer ping a la URL de uno de los servidores, pero dicho cliente tiene configurada la IP de forma estática y no le fue definida la dirección IP de servidor DNS?

Si un PC cliente intenta hacer ping a la URL de un servidor sin tener configurada la dirección IP del servidor DNS, no podrá resolver el nombre de dominio en una dirección IP y, por lo tanto, no podrá establecer una conexión con el servidor.

- · ¿Es posible ver durante la autenticación en la captura de tráfico de wireshark, el nombre de usuario y contraseña de un usuario en el servidor FTP? ¿A qué se debe? Encuentre evidencia que sustenta el enunciado.

Es posible ver el nombre de usuario y la contraseña de un usuario en el servidor FTP durante la autenticación en la captura de tráfico de Wireshark, ya que estos datos se transmiten en texto claro. Esto se debe a que FTP utiliza autenticación en texto claro y no cifra los datos de autenticación.

- · ¿Identifica tráfico HTTP y HTTPS generado al ingresar a los sitios web? ¿El tráfico HTTP es significativo frente al tráfico HTTPS? ¿Identifica que componentes o información de su navegación en el portal web generó este tráfico HTTP?

Es posible identificar el tráfico HTTP y HTTPS generado al ingresar a sitios web en una captura de tráfico de Wireshark. El tráfico HTTP puede ser significativo o no dependiendo del sitio web y de si se utiliza HTTPS para proteger la comunicación. Es posible identificar algunos componentes o información de navegación generados en el tráfico HTTP, como imágenes, scripts, y otros recursos de página.

- · ¿Quién envía el certificado, el cliente, el servidor, o ambos?

El certificado es enviado por el servidor al cliente para autenticar su identidad. Es posible identificar quién envía el certificado en una captura de tráfico de Wireshark al examinar los mensajes de la comunicación SSL/TLS.

- · ¿Es posible encontrar información adicional sobre los servicios prestados en la topología usando la captura de paquetes? Por ejemplo, ¿sería posible encontrar la ubicación de los recursos de contenido? Justifique su respuesta.

Es posible encontrar información adicional sobre los servicios prestados en una topología utilizando la captura de paquetes, como la ubicación de los recursos de contenido. Sin embargo, esto dependerá de cómo esté configurada la red y de si la información se transmite en texto claro o está cifrada. En algunos casos, puede ser difícil o imposible obtener información adicional debido a la protección de la privacidad y seguridad de los datos transmitidos.

- Para el protocolo HTTPS: Explique con sus palabras el proceso de handshake. Encontrar e inspeccionar los detalles del intercambio de certificados, incluyendo la expansión del bloque de protocolo de enlace dentro de la TLS Record. Al igual que los me

El proceso de handshake en el protocolo HTTPS (HTTP seguro) es un intercambio crucial de información entre un cliente (como un navegador web) y un servidor para establecer una conexión segura y cifrada. A continuación, explicaré este proceso con mis palabras:

1. **Cliente Saluda (ClientHello)**: El proceso comienza cuando un cliente (por ejemplo, tu navegador web) envía un mensaje denominado "ClientHello" al servidor al que está tratando de conectarse. Este mensaje contiene información sobre las capacidades del cliente, como los algoritmos de cifrado y las versiones de protocolo que puede utilizar. Además, el cliente genera una serie de números aleatorios.

2. **Servidor Responde (ServerHello)**: El servidor recibe el mensaje "ClientHello", selecciona la versión del protocolo y los algoritmos de cifrado más fuertes que ambos pueden usar, y luego responde con un mensaje "ServerHello". El servidor también genera números aleatorios y proporciona su certificado digital.
3. **Autenticación del Certificado (Certificate)**: El servidor envía su certificado digital al cliente. Este certificado contiene la clave pública del servidor y está firmado por una entidad de certificación confiable (como una autoridad de certificación). El cliente verifica la autenticidad del certificado, asegurándose de que pertenezca al servidor al que está intentando conectarse y de que la firma sea válida.
4. **Generación de Clave de Sesión (Key Exchange)**: En este paso, el cliente y el servidor generan una clave de sesión compartida utilizando los números aleatorios generados previamente y otros datos. Esto se hace de manera segura para garantizar que solo el cliente y el servidor tengan acceso a esta clave.
5. **Finalización del Handshake (Finished)**: Para confirmar que ambos lados han completado con éxito el proceso de handshake y están listos para comunicarse de manera segura, se envían mensajes "Finished" tanto desde el cliente como desde el servidor. Estos mensajes contienen un resumen (hash) de toda la comunicación hasta este punto, lo que permite verificar la integridad de los datos.
6. **Comunicación Segura**: Con el proceso de handshake completado y la clave de sesión compartida en su lugar, el cliente y el servidor pueden cifrar y descifrar los datos que se envían entre ellos. Esto garantiza que cualquier información transmitida, como contraseñas o datos personales, esté protegida y no sea accesible para terceros no autorizados.

En cuanto a la "expansión del bloque de protocolo de enlace dentro de la TLS Record", este es un aspecto técnico del funcionamiento interno de TLS (Transport Layer Security). TLS divide los datos en registros y agrega encabezados a estos registros para facilitar la transmisión segura. La expansión del bloque de protocolo de enlace se refiere a cómo TLS estructura estos registros para incluir el tipo de mensaje (por ejemplo, Handshake, Application Data) y la longitud del mensaje para garantizar que los datos se transmitan correctamente.

En resumen, el proceso de handshake en HTTPS es esencial para establecer una conexión segura entre el cliente y el servidor, autenticar la identidad del servidor y

generar una clave de sesión compartida para la comunicación cifrada. Esto asegura la privacidad y la integridad de los datos transmitidos a través de la web.

- Los servidores web como utilizan los protocolos http y https

Los servidores web utilizan los protocolos HTTP (Hypertext Transfer Protocol) y HTTPS (HTTP Secure) para gestionar las solicitudes y respuestas de los clientes web. Aquí te explico cómo funcionan ambos protocolos:

1. HTTP (Hypertext Transfer Protocol):
   - HTTP es un protocolo de comunicación utilizado para transferir datos, generalmente en forma de páginas web, entre un servidor web y un navegador web (cliente).
   - Cuando un cliente (navegador) realiza una solicitud a un servidor web, envía una solicitud HTTP al servidor para obtener una página web o algún otro recurso.
   - El servidor web procesa la solicitud y devuelve una respuesta HTTP, que incluye la página web o recurso solicitado.
   - Las comunicaciones HTTP son generalmente en texto sin cifrar, lo que significa que la información transmitida entre el cliente y el servidor no está protegida y es susceptible de ser interceptada o modificada por terceros. Por lo tanto, HTTP no es seguro para la transferencia de datos sensibles, como contraseñas o información financiera.

2. HTTPS (HTTP Secure):
   - HTTPS es una extensión segura del protocolo HTTP que utiliza cifrado para proteger las comunicaciones entre el cliente y el servidor.
   - Para implementar HTTPS, el servidor web debe tener un certificado SSL/TLS (Secure Sockets Layer/Transport Layer Security) válido instalado. Este certificado incluye una clave pública y una firma digital emitida por una Autoridad de Certificación (CA) de confianza.
   - Cuando un cliente realiza una solicitud a un servidor HTTPS, se establece un proceso de handshake TLS (Transport Layer Security) entre el cliente y el servidor. Durante este proceso:
     - El servidor presenta su certificado digital al cliente para su verificación.
     - El cliente verifica la autenticidad del certificado y genera una clave de sesión compartida de forma segura.
     - Todas las comunicaciones posteriores entre el cliente y el servidor se cifran utilizando esta clave de sesión compartida.

- El cifrado en HTTPS protege la confidencialidad e integridad de los datos transmitidos, lo que hace que sea seguro para la transferencia de datos sensibles.

En resumen, los servidores web utilizan el protocolo HTTP para la comunicación estándar y la transferencia de datos no segura. Para proporcionar una comunicación segura, utilizan el protocolo HTTPS, que agrega una capa de cifrado a través de SSL/TLS para proteger los datos transmitidos. En la actualidad, es recomendable que los sitios web utilicen HTTPS para garantizar la privacidad y la seguridad de los usuarios en línea.