UT4. Generación de Servicios en Red

4.1 Protocolos Estándar de Comunicación en Red

DAM2 - Programación de Servicios y Procesos

Febrero de 2025

Índice

1. Introducción a los Protocolos de Red

En la era digital actual, los protocolos de comunicación en red forman la columna vertebral de todas las interacciones en Internet. Desde la navegación web hasta las comunicaciones de correo electrónico y la transferencia segura de archivos, estos protocolos facilitan, protegen y optimizan el intercambio de información. Este documento explora los protocolos estándar de comunicación en red a nivel de aplicación, proporcionando una visión detallada de cada uno para comprender mejor su funcionamiento, aplicaciones y seguridad. A través de esta sección, se pretende dotar a los estudiantes y profesionales del conocimiento esencial para implementar y manejar tecnologías de comunicación eficaces en el desarrollo de software moderno.

Los protocolos de red operan en diferentes niveles, los cuales son organizados en modelos como OSI (Open Systems Interconnection) y TCP/IP (Transmission Control Protocol/Internet Protocol), proporcionando un marco para entender la interacción compleja entre diversas tecnologías de red.

1.1. Modelo OSI

El modelo OSI es un marco conceptual de siete capas que detalla las diversas funciones de los sistemas de comunicaciones. Este modelo facilita el diseño modular de redes, asegurando que las tecnologías de diferentes capas puedan desarrollarse e implementarse de manera independiente pero interoperable.

1.2. Modelo TCP/IP

El modelo TCP/IP simplifica las capas del modelo OSI en cuatro niveles y es la base para la Internet. Este modelo proporciona un entendimiento detallado de cómo los datos se mueven de un punto a otro y cómo se asegura la entrega confiable y segura de estos a través de redes heterogéneas.

Esta introducción brinda a los alumnos las herramientas conceptuales necesarias para ’pensar en protocolo’ y comprender mejor la infraestructura subyacente que soporta las comunicaciones en la red moderna.

2. Telnet

2.1. Descripción

Telnet, abreviatura de Teletype Network, es un protocolo de red utilizado para proporcionar una comunicación de texto bidireccional interactivo. Se utiliza principalmente para acceder a servidores remotos y dispositivos de red.

2.2. Funcionamiento

Telnet opera en el puerto 23 y establece una conexión de cliente a servidor, permitiendo a los usuarios ejecutar comandos en un sistema remoto como si estuvieran físicamente presentes.

2.3. Consideraciones de Seguridad

Telnet transmite todos los datos, incluidos los credenciales de usuario, en texto plano, sin cifrar, lo que lo hace susceptible a interceptaciones. Por esta razón, su uso ha sido ampliamente reemplazado por SSH (Secure Shell) en entornos que requieren seguridad.

2.4. Usos Comunes

A pesar de sus riesgos de seguridad, Telnet todavía se utiliza en situaciones donde la seguridad no es una preocupación crítica, o en redes cerradas y controladas. Es útil para la administración de dispositivos de red y servidores en entornos controlados.

2.5. Comandos Básicos

- **open**: Inicia una nueva sesión Telnet.
- **close**: Termina la sesión Telnet.
- **display**: Muestra la configuración actual o los parámetros de Telnet.

2.6. Referencias

Para más información sobre Telnet y recomendaciones de seguridad, se puede consultar:

- Wikipedia: Telnet

3. FTP - File Transfer Protocol

3.1. Descripción

FTP, o Protocolo de Transferencia de Archivos, es uno de los protocolos más antiguos utilizados en Internet para transferir archivos entre un cliente y un servidor en una red TCP/IP.

3.2. Modos de Operación

FTP opera en dos modos distintos que determinan cómo se establecen las conexiones de datos:

- **Modo Activo**: En este modo, el cliente inicia la conexión de comando hacia el servidor y luego escucha en un puerto aleatorio para la conexión de datos. El servidor, conociendo este puerto, inicia la conexión de datos hacia el cliente.
- **Modo Pasivo**: Más amigable con firewalls modernos, en el modo pasivo el cliente inicia tanto la conexión de comando como la conexión de datos, solicitando al servidor que .[es]cuche.[en] un puerto, al cual el cliente luego se conecta.

3.3. Tipos de Transferencia de Ficheros

- **ASCII**: Utilizado para transferir texto. Asegura que los saltos de línea se convierten correctamente entre sistemas con diferentes convenciones de fin de línea.
- **Binario**: Utilizado para cualquier tipo de archivo que no sea texto. Los datos se transfieren en un flujo binario sin conversión, asegurando que el contenido no se altera.

3.4. Comandos Básicos

A continuación, se enumeran algunos de los comandos FTP más importantes:

- **USER**: Identifica al usuario al servidor.
- **PASS**: Autentica al usuario con una contraseña.
- **LIST**: Enumera los archivos en un directorio.
- **RETR**: Recupera un archivo del servidor al cliente.
- **STOR**: Almacena un archivo del cliente al servidor.
- **QUIT**: Termina la sesión FTP.
- **DELE**: Borra un archivo en el servidor.
- **RMD**: Elimina un directorio en el servidor.
- **MKD**: Crea un directorio en el servidor.
- **PWD**: Muestra el directorio actual en el servidor.
- **RNFR** y **RNTO**: Renombra archivos en el servidor.

3.5. Referencias

Para una lista completa de comandos y detalles adicionales, se recomienda consultar:

- RFC 959 en la IETF
- Wikipedia: Protocolo de Transferencia de Archivos

4. HTTP y HTTPS

4.1. Descripción

HTTP (Protocolo de Transferencia de Hipertexto) es el protocolo de red fundamental utilizado en la World Wide Web para la transferencia de datos. HTTPS (HTTP Seguro) es la versión segura de HTTP, que cifra la comunicación entre el navegador y el servidor para aumentar la seguridad.

4.2. Evolución de HTTP a HTTPS

Inicialmente, HTTP transmitía datos en texto plano, lo que presentaba riesgos significativos de seguridad. Con la introducción de HTTPS, que utiliza SSL/TLS para cifrar los datos, se mejora la confidencialidad y la integridad de la información transmitida.

4.3. Cómo Funciona HTTPS

HTTPS cifra la sesión utilizando un sistema de cifrado basado en certificados, que asegura que todos los datos enviados entre el cliente y el servidor sean inaccesibles para cualquier observador externo.

4.4. Importancia de HTTPS

El uso de HTTPS es crucial para proteger las transacciones en línea, asegurando que los detalles sensibles, como la información de tarjetas de crédito y las credenciales de inicio de sesión, no puedan ser interceptados fácilmente.

4.5. Implementación

Para implementar HTTPS, los administradores de sitios web deben obtener un certificado SSL/TLS de una Autoridad de Certificación (CA) y configurar su servidor para manejar HTTPS adecuadamente.

4.6. Comandos Básicos

Los comandos HTTP como GET, POST, PUT y DELETE se utilizan para solicitar acciones en recursos identificados por URLs. En HTTPS, estos comandos son simplemente cifrados.

4.7. Referencias

- MDN Web Docs: HTTP
- RFC 2818: HTTP Over TLS

5. Protocolos de Correo Electrónico: SMTP, IMAP y POP3

5.1. Introducción y Puntos Comunes

Los protocolos de correo electrónico SMTP, IMAP y POP3 son fundamentales en la gestión de mensajes electrónicos. Mientras que **SMTP** (Simple Mail Transfer Protocol) se utiliza para el envío de correos electrónicos, **IMAP** (Internet Message Access Protocol) y **POP3** (Post Office Protocol v3) se encargan de la recepción de correos en los clientes.

Estos protocolos operan sobre la pila TCP/IP y, en la mayoría de los casos, admiten cifrado mediante TLS/SSL para garantizar la seguridad en la transmisión de datos.

5.2. SMTP - Simple Mail Transfer Protocol

5.2.1. Descripción

SMTP es el protocolo estándar para el envío de correos electrónicos a través de Internet. Se encarga de la comunicación entre servidores de correo, permitiendo que los mensajes sean transferidos desde el cliente remitente hasta el servidor de destino.

5.2.2. Comandos Básicos

Los comandos principales de SMTP incluyen:

- **HELO**: Inicia la sesión SMTP con el servidor.
- **MAIL FROM**: Especifica la dirección del remitente.
- **RCPT TO**: Especifica la dirección del destinatario.
- **DATA**: Indica el inicio del contenido del mensaje.
- **QUIT**: Termina la sesión SMTP.
- **VRFY**: Verifica la existencia de un usuario en el servidor.
- **EXPN**: Expande una lista de distribución.

5.2.3. Seguridad

SMTP por sí solo no proporciona cifrado, pero se puede mejorar su seguridad con **STARTTLS** para establecer una conexión segura mediante TLS.

5.3. IMAP - Internet Message Access Protocol

5.3.1. Descripción

IMAP permite acceder a los correos electrónicos directamente en el servidor sin necesidad de descargarlos al dispositivo local. Esto permite que múltiples dispositivos sincronicen el estado de los mensajes en tiempo real.

5.3.2. Características

- Permite el acceso simultáneo desde múltiples dispositivos.
- Los correos pueden organizarse en carpetas dentro del servidor.
- Permite la búsqueda y filtrado de mensajes directamente en el servidor.

5.3.3. Comandos Básicos

Algunos de los comandos fundamentales de IMAP incluyen:

- **LOGIN usuario contraseña**: Inicia sesión en el servidor IMAP.
- **SELECT carpeta**: Selecciona una carpeta de correo.
- **FETCH**: Recupera el contenido de un mensaje.
- **STORE**: Modifica atributos de un mensaje (por ejemplo, marcarlo como leído).
- **LOGOUT**: Finaliza la sesión.

5.4. POP3 - Post Office Protocol v3

5.4.1. Descripción

POP3 es un protocolo diseñado para la descarga de correos electrónicos desde un servidor a un cliente local. A diferencia de IMAP, en POP3 los mensajes generalmente se eliminan del servidor después de ser descargados, lo que impide su acceso desde múltiples dispositivos.

5.4.2. Funcionamiento

- Los correos se descargan en el dispositivo local y, en la mayoría de los casos, se eliminan del servidor.
- Ofrece una gestión más sencilla del correo electrónico, aunque con menos opciones avanzadas que IMAP.

5.4.3. Comandos Básicos

- **USER usuario**: Indica el nombre de usuario.
- **PASS contraseña**: Introduce la contraseña de acceso.
- **LIST**: Lista los mensajes disponibles en el servidor.
- **RETR número**: Descarga el mensaje especificado.
- **DELE número**: Elimina un mensaje del servidor.
- **QUIT**: Finaliza la sesión de POP3.

5.5. Conclusión

Mientras que **SMTP** se utiliza para el envío de correos electrónicos, los protocolos **IMAP** y **POP3** cumplen la función de recuperar mensajes desde un servidor. IMAP permite mantener sincronizados los correos en varios dispositivos, mientras que POP3 es una opción más sencilla que almacena los mensajes localmente.

5.6. Referencias

- RFC 5321 - SMTP
- RFC 3501 - IMAP
- RFC 1939 - POP3

6. SSH - Secure Shell

6.1. Descripción

SSH, o Secure Shell, es un protocolo que proporciona una comunicación segura entre un cliente y un servidor, diseñado para reemplazar conexiones no seguras como las proporcionadas por Telnet. Al contrario de Telnet, que transmite la información en texto claro, SSH cifra la sesión, ofreciendo seguridad contra interceptaciones y ataques.

6.2. Comparación con Telnet

A diferencia de Telnet, que se ha utilizado históricamente para acceder a servidores y dispositivos de red, SSH proporciona un método mucho más seguro debido a su cifrado integral. Telnet puede exponer credenciales de usuario y datos a cualquiera que esté escuchando en la red, mientras que SSH utiliza técnicas de cifrado fuertes para proteger la información transmitida.

6.3. Características Principales de SSH

- **Cifrado de Sesión**: Utiliza algoritmos de cifrado como AES y RSA para garantizar que todos los datos, incluyendo las credenciales y los comandos enviados, sean ilegibles para terceros.
- **Autenticación**: Soporta varios métodos de autenticación, incluyendo contraseña, autenticación basada en clave pública y autenticación de dos factores.
- **Túneles Seguros**: Capacidad para crear túneles seguros para otros protocolos (port forwarding), permitiendo un acceso seguro a servicios de red que de otro modo serían inseguros.
- **Transferencia de Archivos**: Incluye utilidades como SCP y SFTP para transferir archivos de forma segura entre el cliente y el servidor.

6.4. Uso de SSH

SSH es ampliamente utilizado para administrar servidores y dispositivos de red de forma remota, configuración de infraestructuras, y automatización de tareas a través de scripts y comandos seguros.

6.5. Comandos Básicos

- **ssh usuario@servidor**: Conectar al servidor como usuario especificado.
- **scp archivo usuario@servidor:/ruta**: Copiar un archivo al servidor.
- **ssh-keygen**: Generar un par de claves pública y privada para autenticación.

6.6. Referencias

- OpenSSH
- RFC 4251 - The Secure Shell (SSH) Protocol Architecture

7. DNS - Domain Name System

7.1. Descripción

DNS, o Sistema de Nombres de Dominio, es uno de los protocolos más importantes de Internet, diseñado para traducir nombres de dominio fáciles de recordar (como www.example.com) en direcciones IP numéricas (como 192.0.2.1) que son utilizadas por las redes para identificar y localizar dispositivos de manera única.

7.2. Funcionamiento de DNS

Cuando un usuario ingresa un nombre de dominio en su navegador, DNS actúa como un traductor entre lo que el usuario escribe y la dirección IP necesaria para localizar el recurso de Internet correspondiente. Este proceso se conoce como resolución de nombres y es fundamental para la funcionalidad de la red.

7.3. Componentes Principales de DNS

- **Servidores de Nombres (Name Servers)**: Almacenan bases de datos de nombres de dominio y sus direcciones IP asociadas.
- **Registradores de Dominios (Domain Registrars)**: Entidades que permiten registrar nombres de dominio.
- **Resolutores DNS (DNS Resolvers)**: Funcionan como intermediarios que reciben preguntas de los clientes y las traducen a respuestas efectivas, consultando a otros servidores si es necesario.

7.4. Tipos de Registros DNS

Los registros DNS más comunes incluyen:

- **A (Address)**: Mapea un nombre de dominio a una dirección IPv4.
- **AAAA (IPv6 Address)**: Mapea un nombre de dominio a una dirección IPv6.
- **MX (Mail Exchange)**: Especifica los servidores de correo electrónico para un dominio.
- **CNAME (Canonical Name)**: Permite que múltiples nombres de dominio apunten al mismo lugar al ser alias de otros nombres de dominio.

7.5. Seguridad en DNS

La seguridad es una preocupación significativa con DNS, ya que puede ser explotado mediante ataques como DNS spoofing o cache poisoning. Para mitigar estos riesgos, se ha desarrollado DNSSEC (DNS Security Extensions), que agrega capas de autenticación a las respuestas de DNS, protegiendo la integridad de los datos.

7.6. Referencias

- IANA Root Zone Database
- RFC 1035 - Domain Names - Implementation and Specification

8. LDAP - Lightweight Directory Access Protocol

8.1. Descripción

LDAP, o Protocolo Ligero de Acceso a Directorios, es un protocolo utilizado para acceder y gestionar información de servicios de directorio, facilitando la organización y el acceso a información sobre usuarios y recursos dentro de una red.

8.2. Funcionamiento de LDAP

LDAP permite a los usuarios consultar y modificar información almacenada en un directorio activo o cualquier otro servicio de directorio compatible con LDAP. Funciona sobre TCP/IP o sobre otros protocolos de transporte que proporcionan un servicio orientado a la conexión.

8.3. Componentes Principales de LDAP

- **Directorio LDAP**: Una base de datos organizada jerárquicamente que contiene entradas organizadas en un árbol de directorio.
- **Servidor LDAP**: Un servidor que maneja las solicitudes de información, búsqueda y modificación del directorio.
- **Cliente LDAP**: Un cliente que interactúa con el servidor LDAP para solicitar o actualizar información.

8.4. Estructura y Formato de las Consultas LDAP

LDAP utiliza un formato de consulta basado en criterios de búsqueda que especifican atributos y condiciones que deben cumplir las entradas del directorio para ser retornadas como resultados. Estas consultas se basan en un estilo descriptivo que permite una gran flexibilidad y precisión.

8.4.1. Formato de Consulta

Las consultas en LDAP se estructuran usando lo que se conoce como ”Filtro de Búsqueda”. Un filtro de búsqueda define los atributos y condiciones que una entrada debe cumplir para ser incluida en los resultados de la búsqueda. Ejemplo:

```
(atributo=valor)
```

8.4.2. Ejemplo de Consulta

Para buscar a todos los usuarios cuyo apellido sea ”Gonzalez”, se utilizaría el siguiente filtro de búsqueda:

```
(sn=Gonzalez)
```

Donde ‘sn‘ representa el atributo ”surname.[en] el directorio LDAP. Se pueden combinar múltiples criterios usando operadores lógicos como AND (&), OR (—) y NOT (!).

8.4.3. Uso de Operadores Lógicos

Un ejemplo de una búsqueda más compleja que utiliza operadores lógicos podría ser:

```
(&(sn=Gonzalez)(givenName=Juan))
```

Este filtro retornaría entradas para usuarios cuyo apellido sea ”Gonzalez”Y cuyo nombre de pila sea ”Juan”.

8.5. Operaciones Comunes en LDAP

Las operaciones típicas que se pueden realizar en un servidor LDAP incluyen:

- **Bind**: Autenticación de un cliente en el servidor LDAP.
- **Search**: Búsqueda de entradas en el directorio.
- **Add**: Añadir nuevas entradas al directorio.
- **Delete**: Eliminar entradas existentes del directorio.
- **Modify**: Modificar entradas existentes.

8.6. Referencias

- RFC 4511 - Lightweight Directory Access Protocol (LDAP): The Protocol