



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA
INGENIERÍA Y TECNOLOGÍAS AVANZADAS



SOCKETS

PROFESORA:

Susana Sanchez Najera

INTEGRANTES DEL EQUIPO:

Vargas Sánchez Andrea Liliana
Cisneros Martínez Daphne Liliana
Escala Acosta Andres Rafael

GRUPO: 2TV3

Fecha de entrega: 06 de diciembre del 2021

¿QUÉ ES?

Son un mecanismo de comunicación entre procesos que permiten la comunicación bidireccional tanto entre procesos que se ejecutan en una misma máquina como entre procesos lanzados en diferentes máquinas. Trabaja como un tipo de software que actúa como un punto final, funciona estableciendo un enlace de comunicación de red bidireccional entre el extremo de un servidor y el programa receptor del cliente.

Los sockets se realizan y movilizan junto con un conjunto de peticiones de programación que se identifican como llamadas de función, que es denominado interfaz de programación de aplicaciones.

Un socket es capaz de simplificar el funcionamiento de un programa, así los programadores únicamente tienen que manipular las funciones del socket y así confiar en el sistema operativo para transportar los mensajes a través de la red correctamente.

Antes de definir los tipos de sockets, se deben definir las propiedades que tienen los mismos:

1. Fiabilidad de la transmisión, no se pierden los datos transmitidos.
2. Conservación del orden de los datos, llegan en el orden que fueron emitidos.
3. No duplicación de los datos, solo va a existir un ejemplar de cada dato que se emite.
4. Comunicación en modo conectado, se establece una conexión entre dos puntos antes del principio de la comunicación. Una emisión desde un extremo está implícitamente destinada al otro extremo conectado.
5. Conservación de los límites de los mensajes, los límites se pueden encontrar en el destino.
6. Envío de mensajes urgentes, puede existir un envío de datos fuera del flujo normal y accesibles inmediatamente.

A comparación con una tubería que es básicamente un espacio de memoria compartida entre procesos a los cuales se accede de manera ordenada dentro de una misma computadora. Los sockets transmiten información sin importar si los procesos en los cuales se alojan se encuentran o no dentro de la misma computadora.

TIPOS

- **SOCK_STREAM**: orientado a conexión, Socket de flujo

Permiten comunicaciones fiables en modo conectado (propiedades 1,2,3,4), eventualmente autorizan, según el protocolo aplicado, los mensajes fuera de flujo (propiedad 6). El protocolo subyacente en el dominio de internet es TCP. Un socket stream debe estar en estado conectado antes de que se envíe o reciba.

- **SOCK_DGRAM**: Socket de datagrama, no orientado a conexión

Destinados a la comunicación en modo no conectado para el envío de datagramas de tamaño limitado. Las comunicaciones correspondientes tienen la propiedad e. En el dominio del internet su protocolo es UDP. Los datagramas no trabajan con conexiones permanentes, la transmisión por los datagramas es a nivel de paquetes, cada paquete puede tener una ruta distinta provocando que no exista una recepción secuencial.

- **SOCK_RAW**:
Permite el acceso a los protocolos de más bajo nivel, su uso está reservado para el superusuario.
- **SOCK_SEQPACKET**:
Son las comunicaciones que poseen las propiedades 1,2,3,4 y 5, se encuentran en el dominio XNS.

VENTAJAS

Ventajas del Uso de Sockets:

- Optimización del ancho de banda.
- Compatible con casi todos los lenguajes de Programación.
- Está disponible en casi todos los sistemas operativos (Windows, Unix, MacOS).
- Permite intercambiar flujo de datos entre un cliente – servidor de manera fiable y ordenada.
- Se beneficia de la funcionalidad API.

DESVENTAJAS

- Complicado de desarrollo.
- Necesario determinar direcciones ip, puertos y protocolos.

- Es dependiente de la respuesta del servidor.
- En el tipo de sockets no orientado a la conexión, el protocolo UDP no garantiza que los datos lleguen en el mismo orden en que se enviaron.

APLICACIONES

Los sockets se aplican y son usados en los protocolos de Internet como: telnet, ssh, www,... Además, se utilizan en el desarrollo de aplicaciones en red y también en el desarrollo de aplicaciones cliente/servidor y de aplicaciones de comunicación provista por los sistemas tipo Unix.

En la actualidad muchas aplicaciones están implementadas usando sockets de forma interna. Por ejemplo: los gestores de bases de datos relacionales, chats, herramientas para el control de versiones, servicios en la nube, e incluso las páginas que solicitamos en el internet se envían y reciben en los browsers a través de sockets.

CONCLUSIONES

Un socket es un concepto de programación implementado dentro de un sistema Operativo (SO), que permite la comunicación entre diferentes procesos.

Entendemos por comunicación entre procesos a la tarea de intercambiar información requerida entre diferentes aplicaciones para que puedan ejecutarse de manera eficiente.

Un socket emplea el protocolo TCP (Protocolo de control de transmisión) para realizar la conexión y comunicación entre sus terminales, y de esta manera poder completar el intercambio.

Los sockets se diseñan bajo una arquitectura conocida como Cliente/Servidor, esta estructura es una de las más usadas hoy en día pues permite una gran maleabilidad sin perder su foco de funcionamiento.

Debido a su gran uso hoy en día los sockets tienen mucha importancia. A través de ellos y en conjunto con un patrón de diseño podemos implementar un servicio web robusto y extensible, lo que nos deja la puerta abierta a muchas posibilidades.

REFERENCIAS

El interfaz socket. (s. f.). lcc uma. Recuperado 5 de diciembre de 2021, de http://www.lcc.uma.es/%7Eeat/services/i_socket/i_socket.html#link22

¿Qué es un socket? (s. f.). Speedcheck. Recuperado 5 de diciembre de 2021, de <https://www.speedcheck.org/es/wiki/socket/>

de Matteis, L. (2011). Introducción al uso de SOCKETS en Linux. Universidad Nacional del Sur. Recuperado 5 de diciembre de 2021, de https://cs.uns.edu.ar/~ldm/mypage/data/rc/apuntes/introduccion_al_uso_de_sockets.pdf