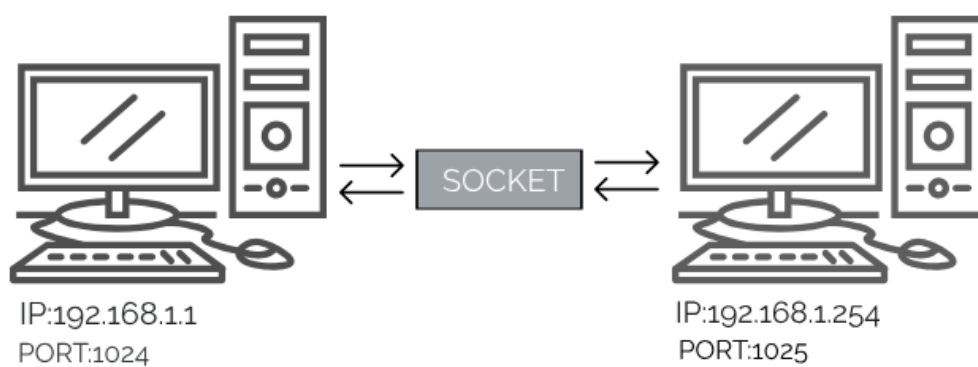


Ejercicios de la UD03



1. **Ejercicios**
2. **Comprobación**
3. **Actividades de aplicación**
4. **Ampliación**
5. **Fuentes de información**

1. Ejercicios

1. Intercambio de mensajes de texto entre cliente y servidor a través de sockets TCP.

Crear una aplicación cliente y una aplicación servidor que intercambien mensajes de texto basando la comunicación en sockets TCP.

La salida generada por la ejecución del servidor es la siguiente:

```
1  (Servidor) Esperando conexiones...
2  (Servidor) Conexión establecida.
3  (Servidor) Abriendo canales de texto...
4  (Servidor) Canales de texto abiertos.
5  (Servidor) Leyendo mensaje...
6  (Servidor) Mensaje leído.
7  (Servidor) Mensaje recibido: Mensaje enviado desde el cliente
8  (Servidor) Enviando mensaje...
9  (Servidor) Mensaje enviado.
10 (Servidor) Cerrando canales de texto.
11 (Servidor) Canales de texto cerrados.
12 (Servidor) Cerrando conexiones...
13 (Servidor) Conexiones cerradas.
```

La salida generada por la ejecución del cliente es la siguiente:

```
1  (Cliente) Estableciendo conexión...
2  (Cliente) Conexión establecida.
3  (Cliente) Abriendo canales de texto...
4  (Cliente) Canales de texto abiertos.
5  (Cliente) Enviando mensaje...
6  (Cliente) Mensaje enviado.
7  (Cliente) Mensaje leído.
8  (Cliente) Mensaje recibido: Mensaje enviado desde el servidor
9  (Cliente) Cerrando canales de texto.
10 (Cliente) Canales de texto cerrados.
11 (Cliente) Cerrando conexiones...
12 (Cliente) Conexiones cerradas.
```

2. Lectura remota de ficheros.

Desarrollar una aplicación en Java que permita leer un fichero de texto ubicado en otro ordenador a través de sockets. Los pasos del proceso serán los siguientes:

- Programa cliente: solicita al usuario el nombre de un fichero incluyendo su ruta completa dentro del sistema de archivos del servidor.
- Programa cliente: envía el nombre y la ruta del fichero al servidor.
- Programa servidor: lee el contenido del fichero y se lo envía al cliente.
- Programa cliente: muestra el contenido por la pantalla.

3. Transferencia de datos por UDP

Desarrollar una aplicación en Java que transmite números desde un cliente a un servidor mediante el uso de sockets UDP. Los pasos del proceso son los siguientes:

- Programa cliente: mediante un bucle genera y envía 10 000 mensajes con el contenido «Mensaje: numero_mensaje» tomando numero_mensaje los valores entre 0 y 9999.
- Programa cliente: cuando ha enviado todos los números manda la cadena «FIN».
- Programa servidor: recibe los mensajes y los almacena en un fichero.
- Programa servidor: cuando recibe la cadena «FIN» termina la ejecución.
- Una vez finalizada la ejecución, comprobar si han llegado todos los datagramas en el mismo orden que se enviaron. Si la ejecución se realiza en un mismo ordenador es probable que lleguen todos los mensajes en el orden correcto. Si la ejecución se realiza en una red de varios ordenadores quizás se produzca alguna pérdida o desorden. Dependerá de muchos factores por lo que cualquier escenario es posible.

4. Cree un servidor que recoja los números que se envían desde el cliente y que devuelva la suma en el momento de enviar un cero.

5. Cree un servidor que devuelva el signo zodiacal europeo y chino a partir de la fecha de nacimiento que envía el cliente.

2. Comprobación

1. ¿Cuál de los siguientes protocolos de internet corresponde a la capa de internet?
 - a) TCP
 - b) UDP.
 - c) HTTP.
 - d) IP.
2. ¿Qué característica no es propia del protocolo `TCP`?
 - a) Está orientado a conexión.
 - b) Garantiza que los paquetes lleguen de forma ordenada.
 - c) Cada paquete puede llegar a su destino por una ruta distinta.
 - d) Garantiza que todos los paquetes se entreguen al destinatario.
3. Para qué es más apropiado utilizar el protocolo `UDP`?
 - a) Para la transferencia de páginas web.
 - b) Para comunicaciones de voz.
 - c) Para transferir ficheros.
 - d) Para enviar correos electrónicos.
4. ¿Qué método de la clase `InetAddress` proporciona la dirección `IP` en modo texto?
 - a) `getByName`.
 - b) `getHostAddress`.
 - c) `getHostName`.
 - d) `getByAddress`.
5. ¿Qué método de la clase `ServerSocket` queda a la espera de recibir peticiones?
 - a) `accept`.
 - b) `bind`.
 - c) `close`.
 - d) `isBound`.
6. ¿Qué método de la clase `Socket` proporciona un stream de lectura?
 - a) `connect`.
 - b) `bind`.
 - c) `getInputStream`.
 - d) `getOutputStream`.
7. ¿Qué método de `DatagramSocket` permite enviar un datagrama?
 - a) `flush`.
 - b) `send`.
 - c) `getOutputStream`.
 - d) `connect`.
8. ¿Qué información hay que proporcionar obligatoriamente a un socket servidor?
 - a) La dirección IP.
 - b) El nombre del host.
 - c) El puerto de escucha de peticiones.
 - d) La dirección IP del cliente.

9. ¿Qué información hay que proporcionar obligatoriamente a un socket cliente?
- a) La dirección IP o el nombre del servidor.
 - b) La dirección IP o el nombre del cliente.
 - e) El puerto de comunicación utilizado por el cliente.
 - d) El número de hilos admitidos.
10. ¿Cuál es el primer paso para crear un servidor de sockets TCP?
- a) Indicar al socket servidor que se quede a la espera de peticiones.
 - b) Aceptar el establecimiento de la conexión.
 - c) Crear un socket de tipo servidor asociado a una dirección y un puerto.
 - d) Intercambiar datos con el cliente.

3. Actividades de aplicación

1. Enumera las capas del modelo TCP/IP.
2. Indica en qué se diferencian las direcciones IPv4 y las IPv6.
3. Explica en qué se diferencia un mensaje de un paquete en comunicaciones en red.
4. Haz una descripción del concepto de datagrama.
5. Averigua qué puerto utilizan por defecto los servidores de base de datos MySQL.
6. Dado un URL averigua la IP utilizando los métodos de la clase InetAddress.
7. Dada una IP averigua la URL del host utilizando los métodos de la clase InetAddress.
8. Explica las diferencias entre los sockets de tipo TCP y los de tipo UDP.
9. Explica por qué la web utiliza el protocolo TCP y no puede utilizar UDP.
10. Enumera los pasos que hay que seguir para crear un servidor de sockets TCP.
11. Enumera los pasos que hay que seguir para crear un cliente de sockets UDP.
12. Describe por qué un servidor web debe tener múltiples hilos para atender las peticiones de los clientes.
13. Explica por qué elegirías utilizar UDP como protocolo para desarrollar un juego en red.

4. Ampliación

1. Profundiza en el conocimiento del modelo de interconexión de sistemas abiertos (OSI). Busca información referente a la aparición de las direcciones IPv6. Averigua cuándo aparecieron y empezaron a utilizarse. Descubre si están implementadas al 100%.
2. Confirma que los siguientes lenguajes de programación permiten la programación de sockets TCP: Java, Python, PHP, C#, C++ y Kotlin.
3. Los sockets UDP utilizan datagramas y estos tienen un límite de tamaño. Descubre cuál es el tamaño mínimo y máximo.
4. La clase MulticastSocket de Java es una subclase de DatagramSocket. Averigua para qué sirve.
5. Los WebSockets permiten establecer una comunicación bidireccional en tiempo real entre un servidor y un navegador web. Profundiza en el conocimiento de esta tecnología.

5. Fuentes de información

- [Wikipedia](#)
- [Programación de servicios y procesos - FERNANDO PANIAGUA MARTÍN \[Paraninfo\]](#)
- [Programación de Servicios y Procesos - ALBERTO SÁNCHEZ CAMPOS \[Ra-ma\]](#)
- [Programación de Servicios y Procesos - M^a JESÚS RAMOS MARTÍN - \[Garceta\] \(1^a y 2^a Edición\)](#)
- [Programación de servicios y procesos - CARLOS ALBERTO CORTIJO BON \[Síntesis\]](#)
- [Programació de serveis i processos - JOAR ARNEDO MORENO, JOSEP CAÑELLAS BORNAS i JOSÉ ANTONIO LEO MEGÍAS \[IOC\]](#)
- GitHub repositories:
 - <https://github.com/ajcpro/psp>
 - <https://oscarmaestre.github.io/servicios/index.html>
 - <https://github.com/juanro49/DAM/tree/master/DAM2/PSP>
 - https://github.com/pablohs1986/dam_psp2021
 - <https://github.com/Perju/DAM>
 - <https://github.com/eldiegoch/DAM>
 - <https://github.com/eldiegoch/2dam-ppp-public>
 - <https://github.com/franlu/DAM-PSP>
 - <https://github.com/ProgProcesosYServicios>
 - <https://github.com/joseluisgs>
 - https://github.com/oscarovillo/dam2_2122
 - https://github.com/PacoPortillo/DAM_PSP_Tarea02_La-Cena-de-los-Filosofos