



Temas:

- Interfaz de Socket.
- Desarrollo de aplicaciones distribuidas.
- Llamadas a Sistema.
- Programación en C.

Enunciado

Se desea crear una aplicación distribuida de mensajería la cual consta de un servidor el cual envía mensajes a todas las aplicaciones cliente instaladas en la red local.

Las aplicaciones tienen que hacer lo siguiente:

- Servidor:

Al iniciarse la aplicación debe mostrar por pantalla un mensaje de bienvenida. A continuación debe abrir un puerto efímero aleatorio entre el 50.000 y el 50.100 (TCP o UDP según su criterio de diseño). Luego, la aplicación debe solicitar el ingreso de un mensaje vía teclado (de hasta 200 caracteres), enviar el mismo a todos los hosts de la subred y quedar a la espera de la escritura de un nuevo mensaje.

- Cliente:

Al iniciarse la aplicación debe mostrar por pantalla un mensaje de bienvenida. A continuación debe abrir el puerto registrado 2.500 (TCP o UDP según su criterio de diseño) y quedar a la espera de la recepción de un mensaje por parte del servidor. Una vez llegado un mensaje, la aplicación debe mostrar por pantalla la siguiente información:

- Fecha y hora de la PC cliente.
- IP y puerto desde la que llega el mensaje.
- Mensaje.

En ambos casos debe realizar el control de error de los ingresos de datos, funciones y las llamadas a sistema utilizadas, mostrando mensajes informativos en caso de falla.

Consideraciones:

- Debe usar IPv4 como protocolo de Internet.
- Para evitar problemas de comunicación entre las aplicaciones, puede clonar la máquina virtual Ubuntu (disponible en Classroom) y hacer correr una aplicación en cada una para probar el laboratorio. En este caso, la configuración de red de cada máquina virtual debe ser “Red interna” (Nombre de la red “LAN”) y se le debe asignar una IP estática a cada una dentro de una misma red (para su comodidad puede usar la red 192.168.1.0/24).
- Contraseña del ROOT de la MV Ubuntu: tcpip2020



Actividades

- 1) Realice un diagrama de flujo de la interacción Cliente-Servidor del laboratorio.
- 2) Codifique ambas aplicaciones usando lenguaje C.

Cuestionario

- 1) ¿Qué tipo de Socket decidió usar para el desarrollo de las aplicaciones? ¿Por qué?
- 2) ¿Qué tipo de servidor decidió desarrollar? ¿Por qué?
- 3) ¿Qué rutinas de envío y recepción de mensajes usó? ¿Por qué?
- 4) Describa las dificultades que tuvo al desarrollar el laboratorio.

Debe entregar los ejecutables y los códigos fuente de ambas aplicaciones y un informe conteniendo los diagramas de flujo de las aplicaciones y las respuestas al cuestionario.