

Sistemas Distribuidos

RPC

Pregunta 1.1 (0.5 pts): ¿Porque sigue levantado el prortmapper del worker?

El comando `sudo systemctl disable rpcbind.service`, permite impedir que el servicio se inicie automáticamente. Este comando elimina el enlace simbólico que indicaba que el servicio debía iniciarse automáticamente.

Esta modificación no se realiza en la sesión en la que se ha realizado el comando, sino que se debe reiniciar la máquina para que tenga efecto, `sudo reboot`.

Pregunta 1.2 (1 pt): ¿Cuál es el resultado que aparece ahora y por qué?

```
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[alumno@worker ~]$ rpcinfo -p worker
rpcinfo: can't contact portmapper: RPC: Remote system error - Connection refused
[alumno@worker ~]$ sudo rpcinfo -p masterM
[sudo] password for alumno:
  program vers proto  port  service
    100000   4   tcp    111   portmapper
    100000   3   tcp    111   portmapper
    100000   2   tcp    111   portmapper
    100000   4   udp    111   portmapper
    100000   3   udp    111   portmapper
    100000   2   udp    111   portmapper
[alumno@worker ~]$
```

RPC no puede crear una conexión con la máquina *worker* ya que hemos deshabilitado el `rpcbind.service` y reiniciado la máquina impidiendo su inicio automático.

Pregunta 1.3 (0.5 pts): ¿En qué máquina está levantado el servicio que está respondiendo al rpcinfo ?

Está levantado en la máquina Master a la que hemos asignado el nombre de *masterM* dentro del archivo `/etc/hosts` ya que no lo hemos deshabilitado como hemos hecho con la máquina Worker.

Pregunta 2.1 (1.5 pts): ¿Cómo podemos verificar si las nuevas informaciones que aparecen en la salida del comando de diagnóstico del “port mapper” son las del programa ejemplo compilado y ejecutado?

```
alumno@master:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[alumno@master ~]$ sudo rpcinfo -p masterM  
[sudo] password for alumno:  
  program vers proto  port  service  
    100000    4  tcp    111  portmapper  
    100000    3  tcp    111  portmapper  
    100000    2  tcp    111  portmapper  
    100000    4  udp    111  portmapper  
    100000    3  udp    111  portmapper  
    100000    2  udp    111  portmapper  
 6000000000    1  udp   37974  
 6000000000    1  tcp   42873  
[alumno@master ~]$
```

Comparamos los resultados del `rpcinfo -p masterM` de antes y de después de la ejecución del programa compilado. Como se puede observar, hay un par de nuevos puertos ocupados con un ID diferente que se pueden atribuir al programa nuevo. Además si se observa el fichero `date.x`, se puede ver que el identificador del programa es 600000000 que coincide con el identificador de programa que nos devuelve `rpcinfo`

Pregunta 2.2 (2 pts): ¿Cual es el resultado y por qué?

```
[alumno@worker exampleRPC]$ ./rdate masterM  
FechaHora sobre el host masterM = 1611592836  
FechaHora sobre el host masterM = Mon Jan 25 16:40:36 2021  
[alumno@worker exampleRPC]$
```

Resultado antes de parar el dateproc

```
[alumno@worker exampleRPC]$ ./rdate masterM  
masterM: RPC: Timed out  
[alumno@worker exampleRPC]$
```

Resultado después de parar el dateproc

Como se observa en las imágenes anteriores, el resultado después de parar el programa es de un time out ya que al haber hecho esto, no hay puertos disponibles donde se pueda enviar la solicitud del Worker para saber la fecha y hora del Máster. En cambio antes de para el dateproc el programa se ejecuta correctamente y devuelve la fecha y la hora del masterM.

Pregunta 2.3 (0.5 pts): ¿Cual es identificador del programa dateproc y el puerto(s) en el que está a la escucha?

```
6000000000    1    udp    37974
6000000000    1    tcp    42873
[alumno@master ~]$
```

Según el *rpcinfo -p masterM* anterior, el identificador es el 6000000000 y los puertos son el 37974 (udp) y el 42873 (tcp).

Pregunta 3.1 (1 pt): ¿Cuales son los ficheros de los códigos fuente originales de nuestro ejemplo y cuáles son los ficheros de código fuentes generados automáticamente por rpcgen?

Los ficheros originales del código fuente del ejemplo son los siguientes: dateproc.c, date.x y rdate.c, además de estos ficheros se incluye el Makefile que ejecuta el rpcgen y el compilador gcc.

Cuando se ejecuta el comando *rpcgen date.x* se crean los siguientes archivos: *date_clnt.c*, *date.h*, *date_svc.c*

Pregunta 3.2: Basándose en los códigos fuente del ejemplo colgado en el Campus Virtual, modifíquelos añadiéndole dos funcionalidades más:

3.2.1 (2 pts) Una que devuelva en valor numérico (timestamp) la diferencia del reloj pasado como parámetro con respecto al reloj del servidor.

3.2.2 (1 pts) Otra que haga lo mismo que la anterior, pero que muestre el resultado en formato texto usando para la conversión timestamp a texto el procedimiento correspondiente del servidor.