SISTEMAS OPERATIVOS II

Tercer curso Ingeniería Informática. Curso 2004-2005

Práctica 4: Procesos en UNIX. Señales.

1. Realizar un programa, *kil*, que recibe como argumentos en nombre de una señal (sin el SIG), el pid de un proceso, el número de veces que ha de enviarle dicha señal (0 para indicarle que se la envie continuamente o hasta que el proceso termine) y (opcionalmente) el retardo (en segundos) entre que se la envía una vez y la siguiente. El programa NO DEBE IMPRIMIR NADA en pantalla. Ejemplo

   ```
   %kil INT 435 100 1
   ```

   Le envia al proceso con pid 435 100 señales SIGINT dejando pasar un segundo después de enviársela cada vez.
   En caso de que no se especifique el cuarto argumento se entenderá que es 0, y NO SE LLAMARA a *sleep* entre una llamada a *kill* y la siguiente.

2. Añadir al shell de la práctica anterior las siguientes funciones de manejo de las señales.

manj A|B [op] SEN1... Instala un manejador (del tipo A o B) para las señales SEN1, SEN2... (sin el SIG). Opciones se refiere a las opciones del manejador (son distintas para cada tipo). Los manejadores se instalarán mediante sigaction. Los tipos de manejador son:

- tipo A Permanente. Cada vez que se ejecute, el manejador incrementa un contador que indica cuantas veces se ha recibido la señal. Admite las siguientes opciones
  - -t el manejador es temporal (se instala con SA_RESETHAND)
  - -v el manejador imprime un mensaje indicando que se ha recibido la señal. (Si no se especifica -v NO DEBE IMPRIMIR NADA).
- tipo B Permanente. Cada vez que se ejecute, el manejador incrementa un contador que indica cuantas veces se ha recibido la señal y además envía al propio proceso la señal para la cual es manejador
  - -f el manejador se instala con SA_NODEFER.
  - -v el manejador imprime un mensaje que indica que ha re-

cibido la señal, la dirección del parámetro que recibe el manejador así como cuantas veces se ha recibido dicha señal (el valor del contador para esa señal).

-p el manejador se instala para ser ejecutado en la pila alternativa

Ejemplos

```
#manejador A -t -v HUP INT SEGV FPE USR1
#manejador A USR2
#manejador B -f -v -p SEGV ILL
#manejador B -v BUS
```

pilaalternativa [bytes] Establece una pila alternativa, de tamaño *bytes* para el manejo de las señales. Si no se especifica *bytes* informa de la pila actual.

ign [SEN1 SEN2. . .] Ignora las señales SEN1 SEN2. . . . Si no se especifican señales nos da la información de las señales (comando siginfo)

masc [-m|-u|-s] [S1. . .] Con -m enmascara las señales S1 S2. . . Con -u las desenmascara. Con -s establece el conjunto de señales enmascaradas a S1 S2. . . . Debe usarse sigprocmask. Si no se especifica opción nos da información de las señales (comando siginfo).

default [SEN1 SEN2. . .] Pone las señales SEN1 SEN2. . . a su acción por defecto. Si no se especifican señales pone todas a si accion por defecto.

siginfo [SEN1 SEN2. . .] Para las señales SEN1 SEN2. . . indica si están enmascaradas, ignoradas, si tienen instalado un manejador, la dirección del manejador y los flags asociados a ese manejador. Si no se especifica ninguna señal lo indicará para todas. Una sola línea por señal. Ejemplo

```
#senales INT HUP SEGV
SIGINT enmascarada manejador SI (0x12450) flags SA_RESETHAND
SIGHUP  enmascarada manejador NO (SIG_DFL) flags
SIGSEGV no enmascarada manejador SI (0x12700) flags SA_NODEFER
```

cont [-c] [SEN1 SEN2. . .] Indica cuantas señales SEN1 SEN2. . . se han recibido. Si no se especifican parametros lo indica de todas. Con -c pone los contadores a 0 de las señales que se especifican, o de todas si no se especifica ninguna.

bucle Hace que el shell entre en un bucle infinito. Instala un manejador para SIGINT (dinstindo de los instalados por hand1 y hand2) que permite salir del bucle pulsando control-c para seguir ejecutando el shell (también se desenmascara SIGINT si es necesario).

fault Produce un fallo de segmentación en el shell. (No vale enviar SIGSEGV, tiene que ser un fallo de segmentación de verdad).

Información detallada de las llamadas al sistema y las funciones de la librería debe obternerse con man (sigaction, sigprocmask, str2sig

Los manejadores deben instalarse con *sigaction* y la señales enmascararse y desenmascararse con *sigprocmask*. SEN representa el nombre de una señal sin el SIG (ejemplo: INT, SEGV, HUP), es decir tal como lo hacen las funciones *strsignal, sig2str y str2sig*. El estado de las señales debe obtenerse con *sigaction* y *sigprocmask*, ya que almacenarlo en variables del programa podría hacer que tras una llamada a *fork()* o *exec* no fuese correcto.

Debe tenerse en cuenta que las señales interrumpen las llamadas al sistema (muy notorio en las funciones que leen de teclado) haciendo que éstas devuelvan -1 con errno=EINTR y sin realizar la tarea para la que fueron invocadas. Es necesario por tanto, que en el caso de que una llamada al sistema sea interrumpida por una señal la reiniciemos, bien directamente después de comprobar que ha sido interrumpida o bien indicándole al sistema que lo haga automáticamente (usando el flag adecuado en *sigaction*).

Pueden usarse las utilidades de /usr/proc/bin para comprobar la veracidad de lo reportado por el programa.

FORMA DE ENTREGA

Debe entregarse sólo el código correspondiente al intérprete de comandos. El programa auxiliar *kil* no debe entregarse. El código del intérprete de comandos se entregará como en prácticas anteriores.

FECHA DE ENTREGA JUEVES 20 MAYO 2005