

SISTEMAS OPERATIVOS II

Tercer curso Ingeniería Técnica Informática de sistemas. Curso 2006-2007

Práctica 4: Procesos en UNIX: Credenciales, señales

Continuar la codificación de un intérprete de comandos (shell) en UNIX. Al igual que en la práctica anterior

- los argumentos entre corchetes [] son opcionales.
- Los argumentos separados por | indican que debe ir uno u otro, pero no ambos simultaneamente.
- Cuando el shell no pueda ejecutar una acción por algún motivo, debe indicarlo con un mensaje como el que se obtiene con `sys_errlist[errno]` o con `perro()`
- El shell no debe dilapidar memoria. La memoria que se asigna (con `malloc()`) y deja de utilizarse debe ser liberada.
- La información que se muestra en pantalla no debe incluir en ningún caso líneas en blanco.
- El shell leerá de su entrada estándar y escribirá en su salida estándar, de manera que podría ser ejecutado un archivo de comandos invocando el shell con su entrada estándar redireccionada a dicho archivo.

En esta práctica el shell, además se incorporarán las funciones para comprobar el funcionamiento de los manejadores de señales y de las credenciales. Todo el manejo de señales se hará con la interface de System V R4 (*sigaction*, *sigprocmask* ...). Los nombres de las señales se especificarán sin el SIG.

seninfo [S1] [S2] ... No da información del estado de cada señal: manejada (con la dirección del manejador, los flags y la máscara asociada), ignorada o acción por defecto, así como de si está enmascarada o no. Ejemplo

```
#seninfo INT HUP SEGV
INT Enmascarada manejador 0x30045a00 SA_RESTART SA_NODEFER. mascara asociada S
HUP No enmascarada Accion por defecto
SEGV No enmascarada Ignorada
```

contador [-r] [S1] [S2] ... Nos informa de los contadores de las señales S1 S2 ..., con -r resetea los contadores de dichas señales. Si no se especifican señales lo hace para todas (tanto mostrar los contadores como resetearlos en su caso)

pila [*tam*] Establece una pila alternativa para la ejecución de las señales de tamaño. Si no se especifica tamaño nos informa del tamaño de la pila alternativa y de su dirección *tam*.

maxmanejador [*n*] Establece el número máximo de veces que un manejador se reenvía la señal. Si no se especifica *n* nos indica a cuanto está establecido dicho valor. Un valor 0 indica que no ha límite predefinido

- **manejador** [-v] [-r] [-p] [-t] [-f] [-dN] [-mSEN1] [-mSEN2] ... **SIG** [**SIG2**] Instala un manejador para las señales SIG1, (SIG2 ...). El manejador incrementará un contador que indica cuantas veces se ha ejecutado. Además se admitirán las siguientes opciones

-mSEN El manejador se ejecuta con la señal SEN enmascarada (miembro *sa.mask* de la estructura *sigaction*)

-dN El manejador se queda en espera N segundos (con la llamada *sleep*). De esta presente debe ser la última instrucción dentro del manejador

-v El manejador al ejecutarse debe imprimir en pantalla la señal que se ha recibido, cuantas veces se ha ejecutado así como la dirección del parámetro que recibe. De no indicarse -v el manejador NO DEBE IMPRIMIR NADA en pantalla.

-r El manejador reenvía al proceso la señal para la cual es manejador. En caso de reenviarse la señal, debe hacerse después de imprimir en pantalla (en caso de que se imprima) y antes de quedarse en espera (en caso de que se haya especificado -sN)

-t El manejador se instala con el flag SA_RESETHAND

-p El manejador se ejecutará en la pila alternativa (se instala con el flag SA_ONSTACK)

-f El manejador se instala con el flag SA_NODEFER

enmascarar [-n] [S1] [S2] ... Enmascara (mediante *sigprocmask* las señales S1 S2 Si no se especifican señales nos informa de las que están enmascaradas. Con -n las desenmascara.

ignorar [S1] [S2] ... Ignora las señales S1 S2 Si no se especifican señales nos informa de las que están ignoradas.

defecto [S1] [S2] ... Pone las señales S1 S2 ... a su acción por defecto. Si no se especifican señales nos informa de las que están a su acción por defecto.

bucle Hace que el shell entre en un bucle infinito. Instala un manejador para

SIGINT que permite salir del bucle pulsando control-c para seguir ejecutando el shell.

segmentation produce un fallo de segmentacin en el shell. (No vale enviar SIGSEGV, tiene que ser un fallo de segmentacin de verdad).

fpe produce una excepci3n de la unidad en punto flotante en el shell. (No vale enviar SIGFPE, tiene que ser un fallo de segmentacin de verdad).

uid [id] . Si no se especifica uid informa de las credenciales (real y efectiva) de usuario del proceso del shell (informar3a tanto de la credencial num3rica como del login asociado). Si se especifica id establece la credencial efectiva del shell a *id.id* puede representar tanto el valor num3rico de la credencial como el login asociado. Ejemplo

```
# uid root
Imposible cambiar credencial: not owner
# uid 15432
Imposible cambiar credencial: not owner
# uid insaaa00
Credencial efectiva de usuario establecida insaaa00 (12001)
#uid
Credencial real de usuario insbbb00 (12007)
credencial efectiva de usuario insaaa00 (12001)
```

infouser id Informa del usuario *id* (*id* puede ser un n3mero de usuario o un login). Informa del login, uid, gid, nombre, directorio HOME y shell.

Informaci3n detallada de las llamadas al sistema y las funciones de la librer3a debe obtenerse con man (sigaction, sigprocmask, sigaltstack, setuid, getpwent ...)

FORMA DE ENTREGA Como en pr3cticas anteriores

FECHA DE ENTREGA VIERNES 19 ENERO 2007