
Universidad de San Carlos de Guatemala

Facultad de Ingeniería

1er. Semestre del 2022

Sistemas Operativos 2. Sección "A"

Ing. César Batz Saquimux

Aux. Orfrant Rivera

Aux. Pablo Barillas

Práctica #1

Administración de procesos.

CONTEXTO

Más del 50% de los servidores utilizados para construir los servicios de internet utilizan el sistema operativo Linux. Debido a que es un sistema seguro y de código abierto, las empresas lo utilizan por sus garantías y bajo costo.

OBJETIVOS

1. Poner en práctica los conocimientos sobre el Kernel de Linux.
2. Familiarizarse con la terminal de Linux y comandos de sistema y usuario.
3. Aprender a crear, monitorizar y montar procesos del Kernel de Linux.
4. Optimizar llamadas del sistema mediante el uso del comando strace.

SOLUCIÓN WEB DE MONITORIZACIÓN DE PROCESOS Y MEMORIA

Se deberá implementar un servidor web en **Golang**, creando una API simple para obtener información sobre la memoria y CPU del sistema haciendo uso de módulos del Kernel escritos en C y escribiéndola en archivos dentro de la carpeta /proc para posteriormente leerlos con Golang. También se podrá implementar la funcionalidad de strace y kill process a través de Golang.

Queda a discreción del estudiante cuál tecnología utilizar para la interfaz gráfica que deberá ser visible a través del navegador. Tanto el frontend como el backend pueden ser utilizados de manera local, no es necesario el uso de servidores en la nube. Los datos deben actualizarse cada 3 segundos automáticamente.

ESPECIFICACIONES

Monitor de Memoria

Se deberá crear y montar un módulo del Kernel llamado `ram_mem_g<#Grupo>` con el cuál se debe extraer información sobre el uso de memoria en el sistema. Al instalar el módulo se debe imprimir el mensaje “El grupo <#Grupo> ha instalado el monitor de memoria”. Al removerse el módulo se debe imprimir el mensaje “El grupo <#Grupo> ha removido el monitor de memoria.” El monitor de memoria debe mostrar la información del consumo de RAM del servidor, en el cual se podrá visualizar la siguiente información:

- Total de memoria RAM del servidor (en MB).
- Total de memoria RAM consumida (en MB).
- Porcentaje de consumo de memoria RAM.
- Gráfica en tiempo real del consumo de memoria RAM.

Se debe crear una gráfica que debe mostrarse como un polígono de frecuencias en tiempo real del consumo de memoria RAM del servidor, la librería a utilizar para crear esta gráfica queda a discreción del estudiante.

Administrador de Procesos y Árbol de procesos

Éste mostrará de manera tabulada todos los procesos que están siendo ejecutados en el servidor, así como un resumen de los procesos. La información general que se debe mostrar es la siguiente:

- Total de procesos: Cantidad total de procesos registrados.
- Procesos en ejecución: Cantidad de procesos en estado running.
- Procesos suspendidos: Cantidad de procesos en estado sleeping.
- Procesos detenidos: Cantidad de procesos en estado stopped.
- Procesos zombies: Cantidad de procesos en estado zombie.

Además debe mostrar de manera tabulada la siguiente información de cada proceso:

- PID: identificador del proceso.
- Nombre del proceso.
- Estado: estado en el que se encuentra el proceso.
- Usuario: nombre de usuario que ejecutó el proceso.

Se deberá mostrar un árbol de procesos dinámico, es decir, se debe mostrar una lista de procesos que permita seleccionar uno y desplegar a todos sus hijos con PID y nombre.

El administrador de procesos debe dejar realizar las siguientes operaciones sobre los procesos:

- Se puede utilizar y visualizar de forma clara el resultado de usar el comando strace sobre el proceso seleccionado.
 - Se debe poder filtrar la salida del comando strace para visualizar solo los system call seleccionados
 - Se debe poder ver un histograma con la frecuencia de ejecución de los system call de un proceso.
- Se puede cambiar el estado del proceso utilizando el system call kill

CONSIDERACIONES

- La interfaz de usuario de la solución web debe ser amigable para que su utilización sea cómoda y fácil de utilizar.
- La distribución de Linux a utilizar queda a discreción del estudiante.
- La obtención de la información debe hacerse obligatoriamente por medio de los módulos de kernel y las respectivas estructuras que la contienen. Prohibido el uso de archivos ya existentes en el directorio /proc.
- Cualquier copia total o parcial será reportada a la Escuela de Sistemas para que proceda como corresponde.
- La práctica será realizada por 4 personas utilizando los grupos establecidos en el laboratorio.

ENTREGABLES

- Código fuente de la solución web.
- Código fuente de los módulos desarrollados.
- Manual de usuario y manual técnico.

FORMA DE ENTREGA

Todo los entregables comprimidos en un archivo .zip con el nombre: [SO2]Practica1_G<no_grupo> (Ejemplo: [SO2]Practica1_G15). Todo debe ir bien identificado con los carnets de los 4 integrantes. Es indispensable que se realice la entrega de 2 formas:

- Mediante UEDI subiendo el archivo zip.
- Subiendo todo a un repositorio de GitHub el cuál debe ser privado con el nombre: SO2_1S2022_P1_G<no_grupo>, (Ejemplo: SO2_1S2022_P1_G15). Se tendrán que agregar como colaboradores en el repositorio a los usuarios: **Ojr** y **Parobacal**.

Se podrá calificar bien sea utilizando el archivo enviado a UEDI o clonando el repositorio de GitHub, se verificarán las fechas de última actualización de los archivos.

Fecha de Entrega: Domingo 20 de Febrero hasta las 23:59.

