TEC | Tecnológico de Costa Rica

Instituto Tecnológico de Costa Rica
Escuela de computación

Tarea 1 de Sistema Operativos

Semestre II

Realizado por:
Jonathan Guillén 201261579

Mayo 2021

Índice general

Capítulo 1

Introducción

En esta tarea consiste en realizar un rastreador de System calls. Los System Calls, o syscalls, son llamadas a funciones internas de las computadoras que trabajan a bajo nivel. El objetivo de esta tarea es imprimir un mensaje con una descripción de los syscalls que son usados en un procedimiento que se llame junto al rastreador de syscalls.

Capítulo 2

Ambiente de Desarrollo

2.1. Lenguaje

El lenguaje usado para la tarea es Rust, un lenguaje nuevo y con un futuro muy prometedor por su potencia y capacidad al momento de programar.

2.2. Ambiente

El ambiente de desarrollo que se utilizo es IntelliJ IDEA, con un plugin que permite correr código en Rust.

2.3. Manejo de versiones

Para este proyecto se decidio el uso de github como herramienta para manejo de versiones. Para poder tener la tarea en cualquier maquina para ejecuratar proyecto se puede obtar por descargar el proyecto desde la pagina o bien clonando el proyecto.

Para descargar el proyecto se puede hacer por medio de la pagina de Github, este este link. Este le descargara el proyecto comprimido en un archivo .zip.

En caso que no se pueda o no se quiera descargar el proyecto desde la pagina se puede clonar el proyecto. Para esto se tiene que acceder a la terminal de linux, despues moverse a la carpeta que quiere clonar el proyecto e ingresar el siguiente comando:

```
git clone
    git@github.com:JhonnyGuillen/Tarea1-Operativos-Rastreador-Syscalls.git
```

Listing 2.1: Link para descargar el proyecto.

Capítulo 3

Estructuras de datos usadas y funciones

En el desarrollo de esta tarea se pensaba usar estructuras para el manejo de la detección de los syscalls, pero a causa de no encontrar fuentes para manejar los syscalls no se pudieron utilizar. Al final la única estructura utilizada en la tarea es la de un parser para el manejo del arranque del rastreador.

Capítulo 4

Instrucciones para ejecutar el programa

4.1. Pre-requisitos

Para la presente tarea es necesario contar con Rust instalado, por lo que es un requisito correr los siguientes comandos en una terminal de Ubuntu.

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

Listing 4.1: Comandos para instalar Rust en Linux.

4.2. Compilación

Para compilar la tarea se debe tener instalado el lenguaje mencionado en pre requisitos en el comando de construcción por lo que el comando completo seria el siguiente.

```
rustc rastreador.rs
```

Listing 4.2: Comando para compilar la tarea.

4.3. Ejecución

Para ejecutar el archivo compilado se debe usar el siguiente comando.

```
./rastreador [-v/-V] 'nombre del programa' [argumentos del programa]
```

Listing 4.3: Comando para ejecutar el proyecto.

Capítulo 5

Actividades realizadas por estudiante

Cuadro 5.1: Tabla de Actividades.

Dia	**Actividad**	**Duracion**
02 de mayo 2020	Instalación de Rust, plugin en IntelliJ y búsqueda de tutoriales	2 Horas
04 de mayo 2020	Practica con el lenguaje y familiarización con la sintaxis	3 Horas
07 de mayo 2020	Implementación del parser de arranque	2 Horas
08 de mayo 2020	Búsqueda de manejo de syscalls	4 Horas
09 de mayo 2020	Continuación de búsqueda de manejo de syscalls	2 Horas
11 de mayo 2020	Revisión del manual sobre manejo de syscalls	3 Horas
14 de mayo 2020	Continuación de revisión del manual sobre manejo de syscalls	3 Horas
15 de mayo 2020	Desarrollo de la documentación de la Tarea	3 Horas
16 de mayo 2020	Creación del repositorio de la tarea	1 Horas

Capítulo 6

Autoevaluación

La tarea corre y tiene la estructura de como debería tener, pero al momento que no se pudo encontrar como manejar el tema de detectar los syscalls si afecto el desarrollo de este. El Código no está completo por la razón ya explicada. también solo hay un reporte de commit en el git ya que, sinceramente, se me olvido el apartado de git hasta el último día.

Personalmente me pondría una nota de 50, porque en si no cumple la idea de la asignación, pero si me esforcé en hacer la tarea, aunque me afecto la poca documentación y pocos ejemplo que hay en internet.

Capítulo 7

Lecciones Aprendidas

- Crear el repositorio desde el primer día de la asignación de la tarea.
- Preguntar a profesor por más información del lenguaje, la escases de documentación y tutoriales que no sean hola mundo afecto demasiado.
- Planificar mejor los tiempos ya que afecto con el desarrollo de un paper.

Capítulo 8

Bibliografía

Documentación de Rust: https://www.rust-lang.org/learn