

1. Introducción

Descripción del Proyecto: Este proyecto consiste en un script en Bash que automatiza tareas de administración de un sistema Linux. El script ofrece un menú interactivo que permite al usuario realizar respaldos de directorios, generar informes sobre el uso de recursos del sistema y crear usuarios nuevos.

Tareas Automatizadas:

- **Respaldo de Directorios:** Comprime un directorio especificado por el usuario y elimina respaldos antiguos para optimizar el espacio en disco.
- **Generación de Informe del Sistema:** Crea un informe sobre el uso de la CPU, la memoria y el disco, y guarda estos datos en un archivo log.
- **Creación de Usuarios:** Permite crear un nuevo usuario en el sistema a través de la terminal.

2. Requisitos Técnicos

Herramientas y Dependencias:

- **Sistema Operativo:** Linux (probado en distribuciones como Ubuntu y Debian).
- **Dependencias:**
 - `tar` para comprimir archivos en el respaldo.
 - `top`, `free`, y `df` para generar informes del sistema.
 - `sudo` para ejecutar comandos de superusuario.
- **Permisos:** El script necesita permisos de superusuario para la creación de usuarios.

3. Desarrollo

Explicación del Código:

- **Variables de Colores y Rutas:** El script define variables de color para mejorar la legibilidad en la terminal y variables para la ruta de los respaldos y el archivo de logs:

```
bash Copiar código

ROJO='\033[0;31m'
CYAN='\033[0;36m'
AMARILLO='\033[0;33m'
BLANCO='\033[0;37m'
NC='\033[0m' # Sin color

BACKUP_DIR="backups"
LOG_FILE="system_report.log"
```

función de Respaldo de Directorio: Esta función le pide al usuario el directorio a respaldar, lo comprime y elimina respaldos antiguos de más de 7 días.

```
bash Copiar código

respaldoBackup() {
    echo -e "${CYAN}Ingrese el directorio a respaldar:${NC}"
    read respaldoDir
    mkdir -p $BACKUP_DIR
    timestamp=$(date +%Y%m%d_%H%M%S)
    tar -czf "$BACKUP_DIR/backup_$timestamp.tar.gz" "$respaldoDir"

    if [ $? -eq 0 ]; then
        echo -e "${BLANCO}Respaldo creado en: $BACKUP_DIR/backup_$timestamp.tar.gz${NC}"
        find $BACKUP_DIR -type f -mtime +7 -exec rm {} \;
        echo -e "${ROJO}Backups antiguos eliminados.${NC}"
    else
        echo -e "${ROJO>Error al crear el respaldo.${NC}"
    fi
}
```

Función de Informe del Sistema: Esta función genera un informe sobre el uso de la CPU, memoria y disco, y lo guarda en `system_report.log`.

```
bash Copiar código

crearUsuario() {
    echo -e "${AMARILLO}Como te llamas?${NC}"
    read nombreUser
    sudo adduser --allow-bad-names $nombreUser

    if [ $? -eq 0 ]; then
        echo -e "${BLANCO}Usuario creado: $nombreUser${NC}"
    else
        echo -e "${ROJO>Error al crear el usuario.${NC}"
    fi
}
```

Menú Interactivo: El menú interactivo presenta al usuario las opciones disponibles y ejecuta la función seleccionada.

```
bash Copiar código

while true; do
    echo -e "${AMARILLO}Elegi una opción:${NC}"
    echo "1. Respaldo de directorios"
    echo "2. Generar informe del sistema"
    echo "3. Crear un nuevo usuario"
    echo "4. Salir"
    read -p "Opción: " option

    case $option in
        1) respaldoBackup ;;
        2) informe ;;
        3) crearUsuario ;;
        4) echo -e "${ROJO}Saliendo...${NC}" ; exit ;;
        *) echo -e "${ROJO}ERROR. Elegi una opcion valida.${NC}" ;;
    esac
done
```

Pruebas del Script:

- **Prueba 1: Respaldo de Directorios:**
 - Ejecuta la opción de respaldo e ingresa un directorio válido. Verifica que el archivo comprimido aparece en el directorio `backups` y que los respaldos antiguos se eliminan correctamente.
- **Prueba 2: Generación de Informe del Sistema:**
 - Selecciona la opción de informe. Luego, verifica que el archivo `system_report.log` contiene información de CPU, memoria y disco.
- **Prueba 3: Creación de Usuario:**
 - Elige la opción para crear un usuario, ingresa un nombre, y verifica que el usuario aparece en el sistema.

Capturas de Pantalla (opcional): Puedes incluir capturas de pantalla del menú interactivo, del archivo `system_report.log` y del directorio `backups` con los archivos generados.

5. Reflexiones Finales

- **Dificultades Encontradas:**
 - Algunos comandos pueden necesitar permisos de superusuario, lo que requiere ejecutar el script con `sudo`.
 - La administración de respaldos antiguos puede ser compleja si se tienen muchos archivos grandes.

- **Posibles Mejoras:**

- Programar la ejecución automática del script usando **cron**.
- Añadir una opción para restaurar respaldos.
- Agregar mensajes más detallados y una validación más robusta de entrada de datos.