

# Documentación Técnica: Script de Administración y Mantenimiento de Sistema

## Introducción

Este proyecto implementa un script en **Bash** que permite a los usuarios automatizar varias tareas de administración y mantenimiento en sistemas Linux. A través de un menú interactivo, el script facilita:

1. **Creación de usuarios** en el sistema con la posibilidad de asignarles directorios y grupos específicos.
2. **Eliminación de archivos temporales y caché** para liberar espacio en disco.
3. **Verificación e instalación de actualizaciones** de paquetes, con la opción de realizar un proceso completo de actualización del sistema.
4. **Generación de un informe de uso de recursos**, registrando la carga de CPU, uso de memoria y espacio en disco.

El propósito de este proyecto es agilizar la administración del sistema y optimizar el uso de los recursos.

## Requisitos Técnicos

### Herramientas y Dependencias

- **Sistema Operativo:** Linux
- **Shell:** Bash (versión 4.0 o superior).
- **Dependencias:**
  - **sudo:** Para ejecutar comandos con privilegios de superusuario.
  - Herramientas estándar de Linux:
    - **useradd:** Para crear nuevos usuarios.
    - **usermod:** Para modificar usuarios existentes.
    - **rm:** Para eliminar archivos y directorios.
    - **top:** Para mostrar el uso de CPU.
    - **free:** Para mostrar el uso de memoria.
    - **df:** Para mostrar el uso del disco.
    - **apt:** Para la gestión de paquetes en distribuciones basadas en Debian.

# Desarrollo

## Explicación Detallada del Código

El script consiste en un menú interactivo que permite al usuario realizar varias acciones de administración y mantenimiento del sistema. A continuación, se describen las funciones clave:

### 1. Menú Principal

El menú interactivo permite al usuario elegir entre las siguientes opciones:

- **Crear nuevo usuario:** Ejecuta el proceso de creación de un nuevo usuario en el sistema.
- **Eliminar archivos temporales:** Limpia los directorios `/tmp` y `~/.cache`, eliminando archivos temporales y de caché.
- **Verificar e instalar actualizaciones:** Verifica si hay actualizaciones disponibles, muestra una lista de las actualizaciones y ofrece la opción de instalarlas.
- **Generar informe de uso de recursos:** Genera un informe sobre el uso de CPU, memoria y espacio en disco.
- **Salir:** Finaliza la ejecución del script.

### 2. Función `crear_usuario`

- Solicita el nombre del nuevo usuario y su directorio personal.
- Si el directorio no se especifica, se establece un valor predeterminado de `/home/nombre_usuario`.
- Usa `useradd` para crear el usuario, y `usermod` para agregarlo a un grupo si es necesario.
- Maneja errores y muestra mensajes de éxito o fracaso.

### 3. Función `eliminar_cache`

- Elimina los archivos de caché y temporales en los directorios `/tmp` y `~/.cache`.
- Informa al usuario sobre el progreso y la eliminación de archivos.

### 4. Función `verificar_actualizaciones`

- Usa `apt update` para verificar si hay paquetes disponibles para actualizar.
- Si se detectan actualizaciones, se muestra una lista y el usuario puede elegir si desea instalarlas.
- Si se elige instalar, el script ejecuta `apt upgrade`
- Al finalizar, limpia el sistema utilizando `apt-get clean`, `apt-get autoclean`, y `apt-get autoremove`.

## 5. Función **generar\_informe**

- Usa los comandos **top**, **free** y **df** para recopilar información sobre el uso de CPU, memoria y espacio en disco.
- Guarda los resultados en un archivo de texto con la fecha y hora actuales.

# Pruebas y Validación

## Cómo Probar las Funcionalidades

### 1. Crear un Usuario:

- Ejecuta el script y selecciona la opción 1.
- Ingresa un nombre de usuario y un directorio personal.
- Verifica que el usuario se haya creado correctamente usando el comando **id nombre\_usuario**.
- Intenta crear el mismo usuario nuevamente para comprobar el manejo de errores.

### 2. Eliminar Archivos Temporales:

- Selecciona la opción 2 y verifica que los archivos de **/tmp** y **~/.cache** hayan sido eliminados.
- Lista los archivos en estos directorios antes y después de la ejecución.

### 3. Verificar e Instalar Actualizaciones:

- Selecciona la opción 3 y verifica que el script detecte las actualizaciones disponibles.
- Verifica que las actualizaciones se instalen correctamente después de seleccionar la opción de instalación.
- Revisa el archivo **actualizacion.log** para confirmar que se registraron las acciones de actualización.

### 4. Generar Informe:

- Selecciona la opción 4 y verifica que se genere un archivo de informe.
- Revisa el contenido del archivo para asegurarte de que la información sea correcta.

## Reflexiones Finales

El desarrollo de este script ha mejorado significativamente la automatización de tareas de administración del sistema, lo que ahorra tiempo y esfuerzo a los administradores de sistemas. Las funciones implementadas permiten mantener el sistema actualizado, optimizar el uso de recursos y gestionar usuarios de manera eficiente.