**Trabajo Final**

Vamos a utilizar UBUNTU 24.04 para el proyecto. Es una distribución de GNU/LINUX basada en Debian muy utilizada. Utiliza el sistema de paquetes apt y los mismos son administrados por Canonical, quienes luego de un proceso de pruebas extensivo y de pasar por los canales de Unstable, Testing Beta pasan a ser parte del estable que permiten que nuestro sistema tome del repositorio. Además, es una versión que tendrá un kernel LTS (Long Term Support) y evitará inconvenientes con actualizaciones de drivers de dispositivos y funciones nuevas en el sistema de archivos.

En el sistema de archivos utilizaremos BTRFS ya que nos permitirá utilizar snapshots y compresión persistente en todo el sistema de forma automática. Esto será perfecto, no solo para mantener el tamaño de las bases de datos comprimidas todo el tiempo, sino también para que en caso de algún inconveniente con quien administre el sistema, pueda volver rápidamente a una versión anterior del S.O.

El backup local, será simplemente un crontab que estará cargando la base de datos en un archivo externo. Como medio de respaldo remoto, utilizaremos RSYNC de ese mismo archivo en una red, conectada como NAS y en caso de que la institución lo considere necesario, podrá ser simplemente un USB remoto que servirá como espejo en caso de falla de algún disco mecánico o sólido.

**Instalación y puesta a punto**

Si bien, podría utilizarse una pequeña computadora de bajo consumo dedicada a este tipo de tareas como un Raspberry Pi, en este caso, utilizaremos una computadora X86/AMD-64 que se encuentra disponible por la institución que nos lo solicitaria.

Descargaremos una ISO de Ubuntu 24.04 de 64bits con el entorno de escritorio predeterminado GNOME y también la herramienta de creación de discos Ventoy.

El programa Ventoy simplemente permite seleccionar el pendrive a formatear para poder ejecutar las iso que uno necesite para hacer de BOOT. Una vez Ventoy formatee nuestro pendrive, colocaremos la ISO dentro de la partición exFAT creada.

Insertaremos el pendrive y reiniciaremos la PC a instalar el sistema operativo objetivo. Apretaremos las teclas F8 u F12 según corresponda y seleccionaremos nuestro pendrive dentro del selector de la BIOS/UEFI (puede que sea necesario desactivar SECURE BOOT cuando convive en sistemas antiguos).

Ahora comenzará la instalación de UBUNTU. La cual, dejaremos todas las opciones en predeterminado a excepción de cuando pida nuestra región que aclararemos Montevideo/URUGUAY (GMT-3) y la distribución de teclado en Español Latinoamericano.

Tras completar el proceso, el sistema habrá creado las particiones necesarias e instalará los paquete predeterminados de la distribución (Únicamente se hará algo distinto si es necesario que conviva con otro sistema operativo. Adicionalmente, el sistema de archivos predeterminado para Root y Home es EXT4 y es el que utilizaremos en el manual a diferencia del BTRFS sugerido).

Una vez iniciado el sistema, simplemente instalaremos por terminal los paquetes de GIT y MariaDB (la SQL) que nos permitirán descargar la aplicación y utilizarla

*sudo apt install git*

*sudo apt install mariadb-server*

Configuramos el SQL con su usuario y para que sea persistente en sistema con systemd

*sudo systemctl enable mariadb*

*sudo systemctl start mariadb*

*sudo mysql -u prestausuario -p prestacontrasena*

Tras instalarse, descargaremos el repositorio donde se encuentra nuestro programa. Lo haremos en la carpeta Documentos de la carpeta personal

*cd $HOME/Documentos*

*git clone https://github.com/gabrielamarante1/PrestamosSoft*

*cd PrestamosSoft*

Finalmente, daremos permisos de ejecución a todos los scripts ejecutables que utilizaremos.

*chmod +x Inicio.sh backup\_prestamos.sh inicializar\_bd.sh*

**Operación de PrestamosSoft**

### ***Paso 1: Inicializar BD***

*Ejecutamos el script inicializar\_bd.sh que creará la base de datos del programa.*

### ***Paso 2b Crear crontab para respaldos automáticos***

*Este cron, va a crear los respaldos cada medianoche ejecutando el script anterior.*

*crontab -e*

*# Añadir la siguiente línea:*

*0 0 \* \* \* /$HOME/Documentos/PrestamosSoft/backup\_prestamos.sh*

### ***Paso 2c Crear copia de seguridad en soporte remoto***

Insertaremos un pendrive y lo sincronizaremos con rsync.

Utilizaremos en la terminal el siguiente comando para ver los dispositivos montados

*df -h*

Y con la ruta conseguida, ejecutaremos la sincronización hasta que muestre el progreso completado.

*rsync -av --progress ~/Documentos/PrestamosSoft/ /media/$user/soporte\_remoto/PrestamosSoft*

### ***Paso 3: Operación principal de usuarios y grupos***

Este script en Bash está diseñado para facilitar la administración de usuarios y grupos en un sistema Linux, además de permitir la ejecución de un script de backup. El script está dividido en diferentes módulos de gestión, incluyendo:

* Gestión de usuarios
* Gestión de grupos
* Ejecución de backups

El script permite la creación, modificación y eliminación de usuarios y grupos, así como la visualización de los usuarios y grupos existentes. Además, incluye una opción para ejecutar un script de respaldo de datos.

*Ejecuta el script principal:*

*./Inicio.sh*

Al ejecutar el script, aparecerá el menú principal con las siguientes opciones:

diff

Copiar código

=== MENÚ PRINCIPAL ===

1) USUARIOS

2) GRUPOS

3) BACKUP

4) Salir

Seleccioná una opción:

En este menú, el usuario puede elegir entre gestionar **usuarios**, **grupos**, **ejecutar un backup** o **salir** del script.

#### **Gestión de Usuarios**

Al seleccionar la opción "1) USUARIOS", se accede al submenú de gestión de usuarios. Este submenú ofrece las siguientes opciones:

diff

Copiar código

=== GESTIÓN DE USUARIOS ===

1) Crear usuario

2) Modificar usuario

3) Eliminar usuario

4) Listar usuarios

5) Volver al menú principal

Seleccioná una opción:

**1) Crear usuario**:

* Permite crear un nuevo usuario en el sistema.
* Solicita el nombre de usuario y el shell deseado (si no se especifica, se asigna /bin/bash por defecto).
* Si el usuario se crea correctamente, se muestra un mensaje de éxito.

**2) Modificar usuario**:

* Permite modificar un usuario existente.
* Se puede cambiar el nombre de usuario y/o el shell.
* Si se cambian ambos datos, el script aplica ambos cambios.

**3) Eliminar usuario**:

* Permite eliminar un usuario del sistema.
* Elimina al usuario junto con su directorio home y sus archivos.

**4) Listar usuarios**:

* Muestra una lista de todos los usuarios actuales del sistema, utilizando el comando compgen -u, que enumera todos los nombres de usuario registrados.

**5) Volver al menú principal**:

* Vuelve al menú principal para elegir otra opción.

#### **Gestión de Grupos**

Al seleccionar la opción "2) GRUPOS", se accede al submenú de gestión de grupos. Este submenú ofrece las siguientes opciones:

diff

Copiar código

=== GESTIÓN DE GRUPOS ===

1) Crear grupo

2) Modificar grupo

3) Eliminar grupo

4) Listar grupos

5) Volver al menú principal

Seleccioná una opción:

**1) Crear grupo**:

* Permite crear un nuevo grupo en el sistema.
* Solicita el nombre del grupo y lo crea utilizando el comando groupadd.

**2) Modificar grupo**:

* Permite cambiar el nombre de un grupo existente.
* Solicita el nombre del grupo original y el nuevo nombre.

**3) Eliminar grupo**:

* Permite eliminar un grupo del sistema.
* Elimina el grupo seleccionado utilizando el comando groupdel.

**4) Listar grupos**:

* Muestra una lista de todos los grupos existentes en el sistema utilizando el comando compgen -g.

**5) Volver al menú principal**:

* Vuelve al menú principal.

#### **Backup**

Al seleccionar la opción "3) BACKUP", el script intenta ejecutar un script de respaldo (backup). El script buscará un archivo llamado backup\_prestamos.sh en el mismo directorio donde se encuentra el script principal y verificará si tiene permisos de ejecución.

**Si el archivo existe y tiene permisos de ejecución**:

* El script backup\_prestamos.sh se ejecutará.

**Si el archivo no existe o no tiene permisos de ejecución**:

* El script mostrará un mensaje de error indicando que el archivo no existe o no tiene permisos de ejecución.

#### **Salir**

Al seleccionar la opción "4) Salir", el script terminará su ejecución y volverá a la línea de comandos.