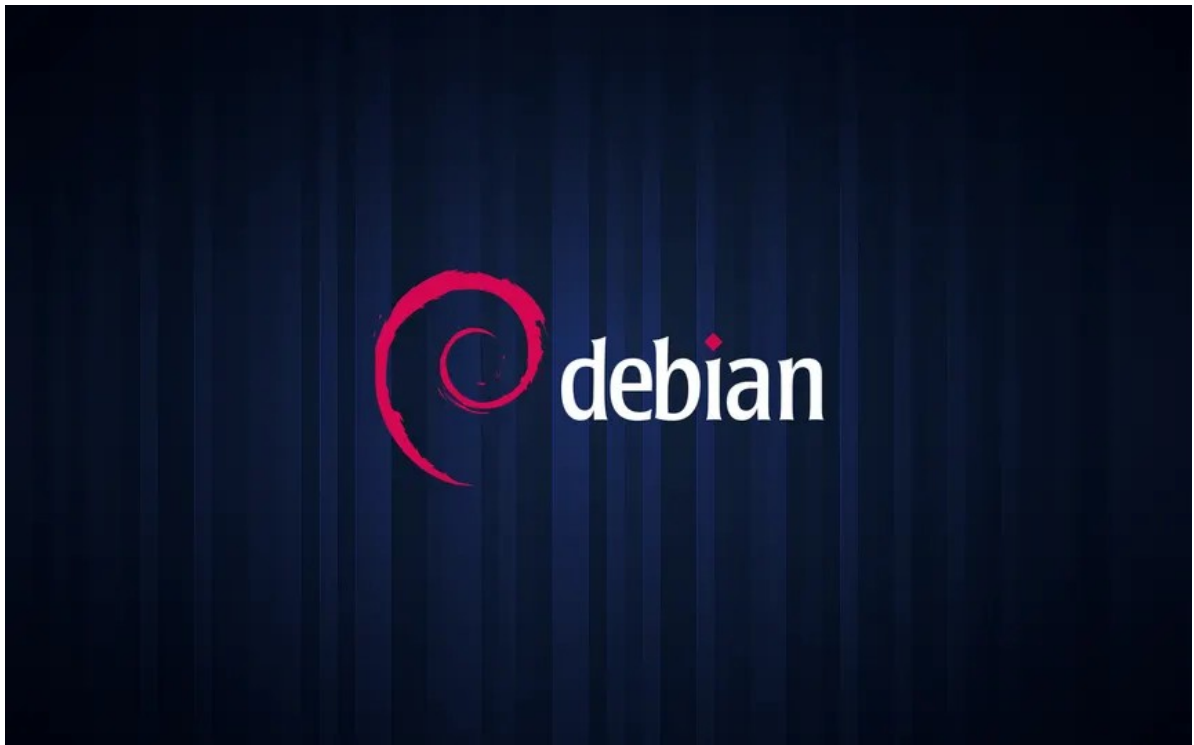


Linux-sysadmin-tools

Scripts de automatización y administración Linux para estudiantes de FP
Linux admin & automation scripts for vocational students



Índice / Table of Contents

1. Objetivo del proyecto / Project Objective
2. Estructura del proyecto / Project Structure
3. Scripts
 - 3.1 set-ip.sh
 - 3.2 samba-setup.sh
 - 3.3 create-users.sh
 - 3.4 backup.sh
4. Ejemplo de archivo de usuarios / Example Users File
5. Requisitos / Requirements
6. Subida a GitHub / GitHub Push
7. Uso de los scripts / How to Use the Scripts

Objetivo del proyecto / Project Objective

Este proyecto consiste en un conjunto de scripts en Bash diseñados para facilitar la administración de sistemas Linux y automatizar tareas comunes. Está orientado especialmente a estudiantes de Formación Profesional (FP) y administradores principiantes que buscan aprender y aplicar buenas prácticas en entornos reales.

Los scripts permiten:

- Configurar una dirección IP estática
- Instalar y configurar un servidor Samba
- Crear múltiples usuarios desde un archivo de texto
- Realizar copias de seguridad comprimidas con `tar`

This project is a collection of Bash scripts designed to simplify Linux system administration and automate common tasks. It is especially aimed at vocational students and beginner sysadmins who want to learn and apply best practices in real-world environments.

The scripts allow you to:

- Configure a static IP address
- Install and set up a Samba server
- Create multiple users from a text file
- Perform compressed backups using `tar`

Estructura del proyecto / Project Structure

```
linux-sysadmin-tools/  
├── scripts/  
│   ├── set-ip.sh  
│   ├── samba-setup.sh  
│   ├── create-users.sh  
│   └── backup.sh  
├── examples/  
│   └── usuarios.txt  
└── README.md
```

Scripts

3.1 set-ip.sh

```
#!/bin/bash
# Configura IP estática con input del usuario | Set static IP with user input

if [ $(id -u) -ne 0 ]; then
    echo "Ejecuta como root"
    exit 1
fi

read -p "Introduce el nombre de la interfaz (ej: eth0): " IFACE
read -p "Introduce la IP estática (ej: 192.168.1.100): " IP
read -p "Introduce la puerta de enlace (ej: 192.168.1.1): " GATEWAY
read -p "Introduce el DNS (ej: 8.8.8.8): " DNS

if [ -z "$IFACE" ] || [ -z "$IP" ] || [ -z "$GATEWAY" ] || [ -z "$DNS" ]; then
    echo "Error: Todos los campos son obligatorios."
    exit 1
fi

cat <<EOF > /etc/network/interfaces
auto $IFACE
iface $IFACE inet static
    address $IP
    netmask 255.255.255.0
    gateway $GATEWAY
    dns-nameservers $DNS
EOF

systemctl restart networking
echo "IP configurada en $IFACE: $IP"
```

3.2 samba-setup.sh

```
#!/bin/bash
# Instala y configura Samba con input del usuario | Install and configure Samba
interactively

if [ $(id -u) -ne 0 ]; then
    echo "Ejecuta como root"
    exit 1
fi

read -p "Introduce el nombre del usuario Samba (ej: root): " SAMBA_USER
read -sp "Introduce la contraseña para $SAMBA_USER: " SAMBA_PASS
echo

if [ -z "$SAMBA_USER" ] || [ -z "$SAMBA_PASS" ]; then
    echo "Error: Usuario y contraseña son obligatorios."
    exit 1
fi

apt update && apt install -y samba
cp /etc/samba/smb.conf /etc/samba/smb.conf.bak

cat <<EOF > /etc/samba/smb.conf
[global]
    workgroup = WORKGROUP
    server string = Servidor Samba
    security = user

[Compartido]
    path = /srv/samba
    browsable = yes
    read only = no
    guest ok = no
EOF

mkdir -p /srv/samba
chmod 2770 /srv/samba
chown root:root /srv/samba

(echo "$SAMBA_PASS"; echo "$SAMBA_PASS") | smbpasswd -a $SAMBA_USER -s
systemctl restart smbd
echo "Samba instalado y configurado con usuario $SAMBA_USER"
```

3.3 create-users.sh

```
#!/bin/bash
# Crea usuarios desde archivo con input del usuario | Create users from file
interactively

if [ $(id -u) -ne 0 ]; then
    echo "Ejecuta como root"
    exit 1
fi

read -p "Introduce la ruta del archivo de usuarios (ej: examples/usuarios.txt):"
USER_FILE

if [ -z "$USER_FILE" ]; then
    echo "Error: Debes introducir un archivo."
    exit 1
fi

if [ ! -f "$USER_FILE" ]; then
    echo "Error: Archivo no encontrado: $USER_FILE"
    exit 1
fi

while IFS=: read -r USER PASS; do
    if id "$USER" &>/dev/null; then
        echo "Usuario $USER ya existe"
    else
        useradd -m -s /bin/bash "$USER"
        echo "$USER:$PASS" | chpasswd
        echo "Usuario $USER creado"
    fi
done < "$USER_FILE"
```

3.4 backup.sh

```
#!/bin/bash
# Backup automático con tar usando input del usuario | Automated backup using
tar interactively

read -p "Introduce la ruta de origen (ej: /home/usuario): " SRC
read -p "Introduce la ruta de destino para el backup (ej: /tmp/backups): " DEST

if [ -z "$SRC" ] || [ -z "$DEST" ]; then
    echo "Error: Debes indicar rutas de origen y destino."
    exit 1
fi

if [ ! -d "$SRC" ]; then
    echo "Error: Directorio de origen no encontrado: $SRC"
    exit 1
fi

mkdir -p "$DEST"
DATE=$(date +%F_%H-%M-%S)
tar -czvf "$DEST/backup_$DATE.tar.gz" "$SRC"

echo "Backup realizado correctamente: $DEST/backup_$DATE.tar.gz"
```


Ejemplo de archivo de usuarios / Example Users File

examples/usuarios.txt

```
juan:Cisco123*  
maria:Cisco123*  
pedro:Cisco123*
```

Formato: usuario:contraseña

Requisitos / Requirements

- Linux basado en Debian.
- Ejecutar scripts con root cuando sea necesario.
- Paquetes: `samba` y `cron` (opcional para backups automáticos).
- Dar permisos de ejecución:

```
chmod +x scripts/*.sh
```

Subida a GitHub / GitHub Push

```
cd ~/linux-sysadmin-tools
git add .
git commit -m "Scripts interactivos completos"
git push origin main
```

- Usuario
- Contraseña: token personal GitHub
- También se puede usar SSH para no introducir token cada vez.

Uso de los scripts / How to Use the Scripts

```
# Configurar IP
sudo ./scripts/set-ip.sh

# Configurar Samba
sudo ./scripts/samba-setup.sh

# Crear usuarios desde archivo
sudo ./scripts/create-users.sh

# Realizar backup
./scripts/backup.sh
```

Todos los scripts son interactivos y pedirán los datos necesarios mediante prompts.