# Prompts Progresivos - Script usuarios.sh

## Prompt 1 — versión inicial: instalar SSH y crear un usuario básico

Generá un script Bash llamado usuarios.sh para Ubuntu Server que realice lo siguiente mínimo:

1. Verifique que el script se ejecute como root; si no, muestre un mensaje y termine con error.
2. Instale openssh-server (usando apt) y arranque/habilite el servicio ssh.
3. Pida por consola un nombre de usuario y cree ese usuario con useradd -m -s /bin/bash.
4. Asigne una contraseña interactiva usando passwd o chpasswd.
5. Muestre un mensaje final indicando cómo conectarse vía SSH (ssh usuario@IP\_SERVIDOR) y deje comentarios básicos en el script explicando cada paso.

Incluí manejo simple de errores (por ejemplo: si useradd falla, mostrar mensaje) y comentarios en español en el encabezado del archivo describiendo el objetivo del script (crear usuarios y habilitar SSH).

## Prompt 2 — versión intermedia: creación masiva desde CSV y grupos

Mejorá usuarios.sh para soportar creación masiva de usuarios desde un CSV (usuarios.csv) con cabecera y columnas usuario,clave,rol. Requerimientos:

1. El script debe comprobar que existe usuarios.csv y salir con mensaje si no.
2. Leer el CSV saltando la cabecera (por ejemplo tail -n +2) e iterar cada línea con IFS=',' read -r usuario clave rol.
3. Sanitizar/limpiar las variables (quitar comillas).
4. Para cada entrada: crear el usuario con useradd -m -s /bin/bash, asignar la contraseña con chpasswd.
5. Crear el grupo rol si no existe (getent group + groupadd) y añadir el usuario al grupo (usermod -aG).
6. Mostrar mensajes claros por cada usuario creado y manejar errores por usuario (no detener todo si falla uno).
7. Documentá en la cabecera el formato requerido del CSV y un ejemplo de uso.

Generá el script listo para probar con un CSV de ejemplo.

## Prompt 3 — versión avanzada: seguridad SSH, UFW y generación de claves

Extendé usuarios.sh para convertirlo en una herramienta de provisioning segura para servidores. Añadí:

1. Instalación y configuración de ufw: instalar ufw, permitir ssh y habilitar el firewall (ufw --force enable).
2. Asegurar parámetros de SSH: editar /etc/ssh/sshd\_config (usando sed) para ajustar PasswordAuthentication y PubkeyAuthentication según lo necesario, y reiniciar el servicio SSH (systemctl restart ssh).
3. Para cada usuario creado, generar un par de claves SSH sin passphrase (ssh-keygen -t rsa -b 2048 -f /home/usuario/.ssh/id\_rsa -q -N "") ejecutado como ese usuario (sudo -u usuario ...).
4. Copiar la clave pública a /home/usuario/.ssh/authorized\_keys, fijar propietarios (chown -R) y permisos (chmod 700 para .ssh, chmod 600 para authorized\_keys).
5. Asegurar que los pasos se ejecutan como root (comprobación inicial), e informar por pantalla por cada acción (instalación, firewall, creación de usuario, generación de claves).
6. Continuar leyendo desde el CSV (saltando cabecera) y procesar todos los usuarios.
7. Mensajes finales indicando que los usuarios ya pueden acceder por SSH y ejemplo de conexión.

Entregá el script con comentarios y manejo básico de errores en cada bloque (instalación, configuración SSH, creación/permiso de .ssh).

## Prompt 4 — versión final: automatización completa (script que subiste)

Generá la versión final de usuarios.sh (lista para entregar) que combine todo lo anterior en un flujo automatizado para Ubuntu Server. Debe incluir:

1. Verificación de ejecución como root (si no, salir con código 1).
2. Variables: USERS\_FILE="usuarios.csv" y comprobación de existencia del archivo (salida con mensaje si falta).
3. Instalación automática de dependencias: apt update -y y apt install -y openssh-server ufw.
4. Habilitar y arrancar el servicio SSH (systemctl enable/start ssh).
5. Configurar UFW permitiendo ssh y habilitar firewall (ufw --force enable).
6. Ajustes en /etc/ssh/sshd\_config con sed para activar/desactivar PasswordAuthentication y PubkeyAuthentication según política, y reiniciar ssh.
7. Lectura del CSV saltando la cabecera (tail -n +2 "$USERS\_FILE") e iteración while IFS=',' read -r usuario clave rol; do ... done. Para cada usuario:
   * limpiar variables (quitar comillas), crear usuario con useradd -m -s /bin/bash, asignar contraseña con chpasswd, crear el grupo rol si no existe y agregar el usuario (usermod -aG).
   * generar par de claves SSH para el usuario con sudo -u "$usuario" ssh-keygen -t rsa -b 2048 -f "/home/$usuario/.ssh/id\_rsa" -q -N "", asegurar .ssh y authorized\_keys y fijar permisos/propietarios correctos.
8. Mensajes claros por cada paso y un mensaje final: Proceso completado. Los usuarios ya pueden acceder vía SSH. Ejemplo de conexión: ssh usuario@IP\_SERVIDOR.
9. Comentarios en la cabecera que expliquen uso, formato del CSV (usuario,clave,rol), y que el script instala paquetes necesarios.

Generá el script completo en Bash listo para ejecutarse en una VM Ubuntu Server y con suficientes comentarios para que el profesor entienda cada cambio/iteración.