Tabla de contenido

[INSTALL 4](#_Toc188100545)

[Configurar VMW para arrancar desde UEFI 4](#_Toc188100546)

[Script que verifica la integridad de la imagen ISO 4](#_Toc188100547)

[Comprobar que el sistema permite UEFI 4](#_Toc188100548)

[Configuración del teclado 4](#_Toc188100549)

[Configuración de la red 4](#_Toc188100550)

[Comprobar que hay conexión 5](#_Toc188100551)

[Crear particiones del Disco Principal (/dev/sda) 5](#_Toc188100552)

[Crear particiones del RAID ZFS (/dev/sdb y /dev/sdc) 5](#_Toc188100553)

[Crear y Cifrar las Particiones 5](#_Toc188100554)

[Cifrado del Root 5](#_Toc188100555)

[Formatear las Particiones Root y Swap 6](#_Toc188100556)

[Preparar los Discos para RAID ZFS con Luks 6](#_Toc188100557)

[Cifrar los Discos para ZFS 6](#_Toc188100558)

[Creación del directorio para LuksAddKey 6](#_Toc188100559)

[Crea un directorio temporal en el sistema base donde estés ejecutando el instalador de Arch Linux: 7](#_Toc188100560)

[Crear un directorio seguro para guardar las claves: 7](#_Toc188100561)

[Generar clave con alta entropía 7](#_Toc188100562)

[Respaldar clave en un dispositivo externo seguro 7](#_Toc188100563)

[Alternativas más seguras: 9](#_Toc188100564)

[Conclusiones: 10](#_Toc188100565)

[Consideraciones 11](#_Toc188100566)

[Verifica las claves activas: 11](#_Toc188100567)

[Respaldo del encabezado de LUKS 12](#_Toc188100568)

[Realizar un respaldo del header: 12](#_Toc188100569)

[Eliminación segura de claves temporales 12](#_Toc188100570)

[Borra las claves del sistema: 12](#_Toc188100571)

[Asegúrate de que las claves no estén en caché: 12](#_Toc188100572)

[Notas interesantes sobre claves: 12](#_Toc188100573)

[Respaldo de clave en dispositivo externo: ¿Es opcional durante la instalación? 14](#_Toc188100574)

[Instalar ZFS y Crear el Pool RAID ZFS 14](#_Toc188100575)

[Crear Datasets ZFS 14](#_Toc188100576)

[Montar las Particiones 15](#_Toc188100577)

[Montar Root y Configurar Swap 15](#_Toc188100578)

[Montar EFI 15](#_Toc188100579)

[Montar el RAID ZFS 15](#_Toc188100580)

[Configurar mirrorlist 16](#_Toc188100581)

[Instala Reflector 16](#_Toc188100582)

[Actualiza la lista de espejos para optimizar las descargas 16](#_Toc188100583)

[Instalación de paquetes base 16](#_Toc188100584)

[Generar File System Tab (fstab) 16](#_Toc188100585)

[Configurar /etc/modprobe.d/zfs.conf para la ARC 16](#_Toc188100586)

[POST-INSTALLATION 17](#_Toc188100587)

[Entrar en el sistema instalado (chroot) 17](#_Toc188100588)

[Configurar el sistema 17](#_Toc188100589)

[Establecer configuraciones regionales y de idioma que usará el sistema 17](#_Toc188100590)

[Poner el sistema en inglés 17](#_Toc188100591)

[Cambiar el idioma del teclado 17](#_Toc188100592)

[Especificar hostname 17](#_Toc188100593)

[Configurar el gestor de arranque con soporte para LUKS 18](#_Toc188100594)

[Configurar HOOKS en mkinitcpio 18](#_Toc188100595)

[Backup de claves UUID y LUKS 18](#_Toc188100596)

[¿Cuándo guardar los UUID? 18](#_Toc188100597)

[Validar UUID antes de configuración de GRUB 19](#_Toc188100598)

[Cómo guardar los UUID de forma segura 19](#_Toc188100599)

[Valida los UUID periódicamente 20](#_Toc188100600)

[Configura GRUB para usar LUKS: 20](#_Toc188100601)

[Configurar UUID de las particiones encriptadas 20](#_Toc188100602)

[Configurar los UUIDs en /etc/default/grub 20](#_Toc188100603)

[Instalar y Configurar GRUB 21](#_Toc188100604)

[Instala el gestor de arranque en el sistema EFI: 21](#_Toc188100605)

[Genera el archivo de configuración de GRUB: 21](#_Toc188100606)

[Optimización y Configuración del RAID ZFS 21](#_Toc188100607)

[Crear Snapshots Automáticos 21](#_Toc188100608)

[Backup a Servidor Remoto o Dispositivo Externo 21](#_Toc188100609)

[Verificar Backup 22](#_Toc188100610)

[Optimizaciones Finales 22](#_Toc188100611)

[Habilitar Servicios Clave 22](#_Toc188100612)

[Configura el borrado de claves y cachés al apagar el sistema: 22](#_Toc188100613)

[Pruebas Finales: 22](#_Toc188100614)

[USERS 23](#_Toc188100615)

[Root password 23](#_Toc188100616)

[Añadir usuario del sistema 23](#_Toc188100617)

[Configuración Sudo 23](#_Toc188100618)

[Habilitación de servicios básicos 23](#_Toc188100619)

[Configuración de Pacman.conf 23](#_Toc188100620)

[Generar carpetas principales [root] 23](#_Toc188100621)

[Generar carpetas principales LaraCanBurn 24](#_Toc188100622)

[Eliminar beeps del sistema 24](#_Toc188100623)

[Desmontar unidades 24](#_Toc188100624)

[INSTALL GRAPHICAL ENVIRONMENT 24](#_Toc188100625)

[Instalar emulador de la terminal 24](#_Toc188100626)

[Iniciar Linux Desktop 24](#_Toc188100627)

[INSTALL ESSENTIAL SYSTEM PACKAGES 25](#_Toc188100628)

[Instalar paquetes básicos 25](#_Toc188100629)

[Instalación de VMWare Tools 25](#_Toc188100630)

[Instalación de servidor de sonido y drivers de sonido 25](#_Toc188100631)

[Drivers 25](#_Toc188100632)

[Servidor 25](#_Toc188100633)

[Utilidades del sistema 25](#_Toc188100634)

[Desarrollo y programación 25](#_Toc188100635)

[Networking 25](#_Toc188100636)

[Seguridad y privacidad 26](#_Toc188100637)

[Comodidades y personalización 26](#_Toc188100638)

[Multimedia 26](#_Toc188100639)

[Compresión y gestión de archivos 26](#_Toc188100640)

[Instalar asistentes de AUR (Arch User Repository) 26](#_Toc188100641)

[Paru 26](#_Toc188100642)

[Yay 26](#_Toc188100643)

[Instalar editor de Código Visual Studio Code 27](#_Toc188100644)

[Instalar minikube 27](#_Toc188100645)

[INSTALL ESSENTIAL ENVIRONMENT PACKAGES 27](#_Toc188100646)

[BIBLIOGRAPHY 27](#_Toc188100647)

[Resultado esperado con lsblk 28](#_Toc188100648)

[Explicación de Cada Componente 28](#_Toc188100649)

[Validación 28](#_Toc188100650)

# INSTALL

## Configurar VMW para arrancar desde UEFI

Options>Advanced>Firmware type>UEFI

## Script que verifica la integridad de la imagen ISO

## Comprobar que el sistema permite UEFI

/sys/firmware/efi/efivars

## Configuración del teclado

loadkeys es

## Configuración de la red

iwctl

station list

station wlan0 get-networks

exit

## Comprobar que hay conexión

ping archlinux.org

ping -c 1 archlinux.org

## Crear particiones del Disco Principal (/dev/sda)

cfdisk /dev/sda

# Seleccionar GPT

# Crear las particiones:

# 1. /dev/sda1 -> EFI (512M, tipo EFI System)

# 2. /dev/sda2 -> Root (resto del espacio, Linux File System)

## Crear particiones del RAID ZFS (/dev/sdb y /dev/sdc)

cfdisk /dev/sdb

cfdisk /dev/sdc

# Seleccionar GPT

# Crear las particiones:

# 1. /dev/sdb -> RAID ZFS Parte 1 (100G)

# 2. /dev/sdc -> RAID ZFS Parte 2 (100G)

## Crear y Cifrar las Particiones

### Cifrado del Root

#### Formateo de EFI

mkfs.vfat -F32 /dev/sda1 # Formatear EFI

#### Configuración de Partición Root con LUKS

cryptsetup luksFormat --type luks2 /dev/sda2

cryptsetup open /dev/sda2 crypt-root

##### Valida:

cryptsetup status crypt-root 🡪 Añadir después de cada cryptsetup open, cryptsetup status para validar los dispositivos mapeados

#### Configurar LVM:

pvcreate /dev/mapper/crypt-root

vgcreate vol /dev/mapper/crypt-root

lvcreate -L 8G vol -n swap # Crear LVM para swap

lvcreate -l +100%FREE vol -n root # Resto para root

### Formatear las Particiones Root y Swap

mkswap /dev/mapper/vol-swap # Formatear el swap

mkfs.ext4 /dev/mapper/vol-root # Formatear la raíz

## Preparar los Discos para RAID ZFS con Luks

### Cifrar los Discos para ZFS

#### Cifrar las particiones con LUKS:

cryptsetup luksFormat --type luks2 /dev/sdb

cryptsetup luksFormat --type luks2 /dev/sdc

#### Abrir particiones cifradas:

cryptsetup open /dev/sdb crypt-zfs1

cryptsetup open /dev/sdc crypt-zfs2

#### Validar estado:

cryptsetup status crypt-zfs1

cryptsetup status crypt-zfs2

## Creación del directorio para LuksAddKey

En este paso, estamos hablando de **usar un archivo como clave adicional para desbloquear las particiones cifradas con LUKS**. Este archivo se genera con alta entropía (aleatoriedad) y sirve como un método alternativo (o adicional) para desbloquear la partición, en lugar de usar únicamente una contraseña manual.

* **¿Por qué hacerlo?**:
  + Aumenta la seguridad y flexibilidad: Puedes desbloquear particiones automáticamente en un entorno seguro al usar un archivo clave (por ejemplo, un servidor o hardware seguro).
  + Sirve como respaldo: Si pierdes la contraseña principal, el archivo clave puede ser usado para recuperar acceso.

### Crea un directorio temporal en el sistema base donde estés ejecutando el instalador de Arch Linux:

### Crear un directorio seguro para guardar las claves:

mkdir -p /mnt/etc/luks-keys

### Generar clave con alta entropía

En caso de olvidar la contraseña principal, puedes usar la clave adicional para acceder a la partición cifrada:

##### Clave Luks sda2

openssl rand -base64 64 > /mnt/etc/luks-keys/root.key

chmod 600 /mnt/etc/luks-keys/root.key

##### Clave Luks sdb

head -c 64 /dev/random > /mnt/etc/luks-keys/sdb.key

chmod 600 /mnt/etc/luks-keys/sdb.key

#### Alternativa con OpenSSL

openssl rand -base64 64 > /mnt/etc/luks-keys/sdb.key

chmod 600 /mnt/etc/luks-keys/sdb.key

### Respaldar clave en un dispositivo externo seguro

**Opciones comunes para --cipher-algo**

1. **aes-xts-plain64 (Recomendado)**
   * **Propósito**: Es el estándar más utilizado para cifrado de discos completos.
   * **Ventajas**:
     + Muy seguro.
     + Eficiente para operaciones de lectura/escritura en sistemas modernos.
     + Compatible con la mayoría de los sistemas y hardware.
   * **Tamaño de clave recomendado**: 256 bits (AES-256).
   * **Ejemplo**:

cryptsetup luksFormat --cipher aes-xts-plain64 --key-size 256 --hash sha256 /dev/sda2

1. **aes-cbc-essiv:sha256 (Compatibilidad)**
   * **Propósito**: Más antiguo, usado en sistemas con hardware más viejo.
   * **Ventajas**:
     + Adecuado si el hardware no soporta XTS.
     + Combina cifrado AES con un método de inicialización seguro (ESSIV).
   * **Desventajas**:
     + Menos eficiente que AES-XTS.
   * **Ejemplo**:

cryptsetup luksFormat --cipher aes-cbc-essiv:sha256 /dev/sda2

1. **chacha20-poly1305 (Rendimiento en CPUs débiles)**
   * **Propósito**: Alternativa moderna para dispositivos con poca capacidad de cálculo (como móviles o Raspberry Pi).
   * **Ventajas**:
     + Más rápido que AES en hardware sin aceleración.
     + Asegura buena seguridad con rendimiento eficiente.
   * **Ejemplo**:

cryptsetup luksFormat --cipher chacha20-poly1305 /dev/sda2

### Alternativas más seguras:

**Opciones Avanzadas para Mayor Seguridad**

1. **aes-xts-plain64 con claves de 512 bits**
   * **Por qué es más seguro**: Aumenta el tamaño de la clave, dificultando aún más un ataque de fuerza bruta.
   * **Requisitos**:
     + Hardware moderno con soporte para claves más largas.
     + Un impacto mínimo en el rendimiento.
   * **Comando**:

cryptsetup luksFormat --cipher aes-xts-plain64 --key-size 512 --hash sha512 /dev/sda2

1. **serpent-xts-plain64 o twofish-xts-plain64 (Criptografía alternativa)**
   * **Por qué es más seguro**:
     + Son algoritmos alternativos a AES que ofrecen un nivel similar de seguridad pero reducen la dependencia de un único estándar (AES).
     + Evitan problemas potenciales si AES alguna vez es vulnerado.
   * **Impacto**:
     + Generalmente más lentos que AES.
     + Requieren soporte en el kernel y compatibilidad en el sistema.
   * **Comandos**:
     + Usar Serpent:

cryptsetup luksFormat --cipher serpent-xts-plain64 --key-size 256 /dev/sda2

* + - Usar Twofish:

cryptsetup luksFormat --cipher twofish-xts-plain64 --key-size 256 /dev/sda2

1. **Cifrado combinado (AES y otro algoritmo)**
   * Algunos sistemas avanzados permiten combinar algoritmos, como cifrar con AES y luego aplicar Serpent o Twofish como una capa adicional.
   * **Ejemplo**:
     + Primero, cifra con AES:

cryptsetup luksFormat --cipher aes-xts-plain64 --key-size 256 /dev/sda2

* + - Luego, aplica otra capa de cifrado con otro algoritmo utilizando dm-crypt.

1. **Chacha20 con Poly1305 + SHA3**
   * Una opción moderna que combina un algoritmo rápido y seguro (Chacha20) con una autenticación robusta (Poly1305) y un hash moderno (SHA3).
   * **Comando**:

cryptsetup luksFormat --cipher chacha20-poly1305 --hash sha3-512 /dev/sda2

**¿Qué más puedes añadir para seguridad adicional?**

1. **PBKDF (Key Derivation Function) con más iteraciones**
   * **Por qué es útil**: Aumenta el tiempo necesario para derivar una clave, lo que hace más lento cualquier intento de descifrado de fuerza bruta.
   * **Ejemplo**:

cryptsetup luksFormat --iter-time 5000 /dev/sda2

* + - Aquí --iter-time especifica 5000 ms para calcular las claves.

1. **Utilizar LUKS2 con Argon2 (por defecto en sistemas modernos)**
   * **Por qué es útil**: Argon2 es un método moderno de derivación de claves que es resistente a ataques de hardware.
   * **Ejemplo**:

cryptsetup luksFormat --type luks2 --pbkdf argon2id /dev/sda2

1. **Cifrado en capas**
   * Puedes aplicar LUKS encima de un sistema ZFS cifrado, combinando los métodos de cifrado de ambos sistemas para aumentar la seguridad.

### Conclusiones:

**Recomendación**

Para un sistema con alto rendimiento y máxima seguridad:

1. **Cifrado:** aes-xts-plain64 con clave de 512 bits.
2. **PBKDF:** Usa --pbkdf argon2id.
3. **Hash:** Cambia a sha512 o sha3-512.
4. **Iteraciones:** Aumenta --iter-time a 5000 ms o más, dependiendo de tu hardware.

Comando final recomendado:

cryptsetup luksFormat --cipher aes-xts-plain64 --key-size 512 --hash sha512 --iter-time 5000 --pbkdf argon2id /dev/sda2

### Consideraciones

* **Hardware:** Asegúrate de que tu sistema tiene soporte para claves largas (512 bits) y que el rendimiento no se vea muy afectado.
* **Recuperación:** Si aplicas configuraciones avanzadas, documenta todo y realiza respaldos del encabezado de LUKS (header).
* **Compatibilidad:** Las configuraciones avanzadas como serpent o twofish pueden no ser compatibles con todos los sistemas.

En este caso, vamos a utilizar como prueba el primer tipo:

cp /mnt/etc/luks-keys/sda.key /media/usb/backup\_keys/sda.key

gpg --symmetric --cipher-algo AES256 /media/usb/backup\_keys/sda.key

cp /mnt/etc/luks-keys/sdb.key /media/usb/backup\_keys/sdb.key

gpg --symmetric --cipher-algo AES256 /media/usb/backup\_keys/sdb.key

#### Añadir clave a las particiones:

cryptsetup luksAddKey /dev/sda2 /mnt/etc/luks-keys/root.key

cryptsetup luksAddKey /dev/sdb /mnt/etc/luks-keys/sdb.key

cryptsetup luksAddKey /dev/sdc /mnt/etc/luks-keys/sdb.key

Una vez que entres en chroot, el archivo estará disponible en /etc/luks-keys/sdb.key.

## Verifica las claves activas:

cryptsetup luksDump /dev/sda2

cryptsetup luksDump /dev/sdb

cryptsetup luksDump /dev/sdc

## Respaldo del encabezado de LUKS

### Realizar un respaldo del header:

cryptsetup luksHeaderBackup /dev/sda2 --header-backup-file /media/usb/luks\_sda2\_header.img

cryptsetup luksHeaderBackup /dev/sdb --header-backup-file /media/usb/luks\_sdb\_header.img

cryptsetup luksHeaderBackup /dev/sdc --header-backup-file /media/usb/luks\_sdc\_header.img

**Notas importantes sobre el respaldo del encabezado:**

* **Qué es el header:** El encabezado de LUKS contiene la información necesaria para descifrar una partición. Sin este archivo, las claves y el acceso a los datos cifrados pueden perderse permanentemente.
* **Dónde guardarlo:** Almacena el encabezado en un medio externo cifrado y seguro (como una USB cifrada o un gestor de contraseñas).
* **Uso del respaldo:** En caso de corrupción del encabezado, puedes restaurarlo:

cryptsetup luksHeaderRestore /dev/sda2 --header-backup-file /media/usb/luks\_sda2\_header.img

## Eliminación segura de claves temporales

### Borra las claves del sistema:

shred -u /mnt/etc/luks-keys/\*

### Asegúrate de que las claves no estén en caché:

echo 3 > /proc/sys/vm/drop\_caches

## Notas interesantes sobre claves:

#### Notas

* Puedes crear el directorio para las claves **en cualquier momento antes de usar luksAddKey**, ya sea en el sistema base o dentro del chroot.
* El directorio y la clave no dependen de los usuarios del sistema.
* Guardar el archivo dentro del sistema raíz (/etc/luks-keys) asegura que esté disponible después de la instalación, especialmente si deseas automatizar el desbloqueo de particiones.

#### Clave en un dispositivo hardware

* Si tienes un dispositivo físico como una USB o un token, puedes almacenar la clave ahí y usarla con:

luksAddKey /dev/sda2 /media/usb/clave

luksAddKey /dev/sdb /media/usb/clave

**Uso de cryptroot /dev/sda2 /media/usb/keyfile luks**

* **¿Qué significa /media/usb?**  
  Es el directorio donde tu USB debe estar montado. Si el USB se monta automáticamente en otro lugar (por ejemplo, /run/media/usuario/USB), debes sustituir /media/usb por esa ruta específica.

##### Pasos para garantizar que el USB esté correctamente montado:

* + Conecta el USB y ejecuta:

lsblk

Esto mostrará las particiones disponibles y su punto de montaje.

* + Si el USB no está montado automáticamente:

mkdir -p /media/usb

mount /dev/sdX1 /media/usb # Reemplaza sdX1 con el dispositivo correspondiente

* + Luego, utiliza la ruta correcta para agregar la clave:

cryptsetup luksAddKey /dev/sda2 /media/usb/keyfile

#### ¿Cómo usar esta clave?

Después de añadir la clave con luksAddKey, puedes usarla para desbloquear la partición con el comando:

cryptsetup open --key-file /ruta/a/clave /dev/sdb crypt-zfs1

Esto desbloquea /dev/sdb utilizando la clave almacenada en /ruta/a/clave.

#### Consideraciones de Seguridad

* **Protección del archivo:** La clave debe estar protegida (por ejemplo, con permisos 600) para evitar que otros usuarios accedan a ella.
* **Respaldo seguro:** Si pierdes todas las claves configuradas en LUKS, perderás acceso a la partición cifrada. Es buena práctica guardar una copia del archivo de clave en un lugar seguro (como un gestor de contraseñas o almacenamiento externo cifrado).

### **Respaldo de clave en dispositivo externo: ¿Es opcional durante la instalación?**

Sí, respaldar la clave en un dispositivo externo es **opcional durante la instalación**. Puedes realizar este paso después de que el sistema esté completamente configurado. Sin embargo, **se recomienda hacerlo inmediatamente después de crear y probar el acceso a la partición cifrada**, para prevenir pérdida de datos en caso de fallos.  
Si decides realizarlo más tarde, asegúrate de que la clave está almacenada de forma segura dentro del sistema en el directorio /etc/luks-keys/.

### Instalar ZFS y Crear el Pool RAID ZFS

sudo pacman -S zfs-dkms zfs-utils

zpool create -f -o ashift=12 raidz raidz /dev/mapper/crypt-zfs1 /dev/mapper/crypt-zfs2

#### Verificar estado:

zpool status raidz

### Crear Datasets ZFS

#### Configurar Volúmenes dentro del Pool

zfs create raidz/root

zfs create raidz/data

#### Habilitar Optimización y Seguridad

**Habilitar compresión y desactivar atime:**

zfs set compression=lz4 raidz # Habilitar compresión

zfs set atime=off raidz # Desactivar atime

**Activar cifrado para un dataset específico**:

zfs set encryption=aes-256-gcm -o keyformat=passphrase raidz/securedata # Esto asegura que los backups realizados con zfs send sigan cifrados si el dataset está cifrado.

**Activar deduplicación:**

zfs set dedup=on raidz # Esto puede ser útil si manejas datos repetitivos.

**Descargar la clave de cifrado para asegurar que no quede en memoria**:

zfs unload-key raidz # Para asegurar que no quede información en memoria tras apagar y descargar clave de cifrado

***Nota:***

* **Establecer políticas de snapshots automáticos**: Usa herramientas como zfs-auto-snapshot para automatizar snapshots en lugar de crearlos manualmente.

## Montar las Particiones

### Montar Root y Configurar Swap

mount /dev/mapper/vol-root /mnt

swapon /dev/mapper/vol-swap

### Montar EFI

mkdir -p /mnt/boot/efi

mount /dev/sda1 /mnt/boot/efi

### Montar el RAID ZFS

mkdir -p /mnt/data

zfs mount raidz/data /mnt/data

## Configurar mirrorlist

### Instala Reflector

sudo pacman -S reflector

### Actualiza la lista de espejos para optimizar las descargas

sudo reflector --verbose --latest 10 --sort rate --save /etc/pacman.d/mirrorlist

## Instalación de paquetes base

pacstrap /mnt base linux-zen linux-zen-headers sof-firmware base-devel grub efibootmgr nano vim networkmanager lvm2 cryptsetup

## Generar File System Tab (fstab)

genfstab -U /mnt > /mnt/etc/fstab

cat /mnt/etc/fstab

## Configurar /etc/modprobe.d/zfs.conf para la ARC

ARC (Adaptive Replacement Cache) utiliza la RAM para acelerar operaciones de lectura/escritura en ZFS. Con 16GB de RAM en tu sistema y 8GB de swap, ajustar el uso del ARC puede mejorar el rendimiento.

**Beneficios de configurar ARC:**

1. **Control del consumo de memoria RAM:**
   * Por defecto, ZFS utiliza hasta el 50% de la memoria RAM para ARC. Esto puede ser excesivo en sistemas con alta demanda de memoria para otras tareas.
2. **Optimización para sistemas de uso mixto:**
   * Al limitar el ARC, aseguras que otras aplicaciones tengan suficiente memoria disponible.

**Ejemplo de configuración en /etc/modprobe.d/zfs.conf:**

Para un sistema con 16GB de RAM, puedes limitar ARC de la siguiente manera:

options zfs zfs\_arc\_max=8589934592 # 8GB (mitad de la RAM)

options zfs zfs\_arc\_min=2147483648 # 2GB (mínimo garantizado para ZFS)

**Cómo ajustarlo:**

* Si planeas usar intensivamente ZFS, mantén un límite alto para zfs\_arc\_max (ej., 8GB).
* Si tu sistema necesita más RAM para otras aplicaciones, reduce el límite (ej., 4GB para zfs\_arc\_max).
* Prueba y ajusta según el uso del sistema usando:

arcstat

* Para recargar las configuraciones en caso de ajustarlo nuevamente haremos:  
  modprobe -r zfs && modprobe zfs

# POST-INSTALLATION

## Entrar en el sistema instalado (chroot)

arch-chroot /mnt

## Configurar el sistema

ln -sf /usr/share/zoneinfo/Europe/Madrid /etc/localtime  crear un enlace simbólico para la zona horaria

date (check)

hwclock --systohc  sincronizar el reloj del sistema

## Establecer configuraciones regionales y de idioma que usará el sistema

nano /etc/locale.gen

Para filtrar, vamos a hacer  Ctrl W

Filtraremos por en\_US y descomentamos la siguiente línea  en\_US.UTF-8 UTF-8

Filtraremos nuevamente por es\_ES y haremos lo mismo  es\_ES.UTF-8 UTF-8

## Poner el sistema en inglés

echo "LANG=en\_US.UTF-8" > /etc/locale.conf

## Cambiar el idioma del teclado

echo "KEYMAP=es" > /etc/vconsole.conf

Guardamos y aplicamos los cambios con  locale-gen

## Especificar hostname

echo "ArchLinux" > /etc/hostname

## Configurar el gestor de arranque con soporte para LUKS

### Configurar HOOKS en mkinitcpio

#### Edita el archivo de configuración de mkinitcpio para añadir soporte a LUKS y LVM:

nano /etc/mkinitcpio.conf

#### Modifica la línea HOOKS para que incluya:

#### HOOKS=(base udev autodetect modconf block encrypt lvm2 filesystems keyboard fsck)

#### Regenera las imágenes del kernel:

sudo mkinitcpio -P linux-zen

## Backup de claves UUID y LUKS

* Respaldo del encabezado de LUKS
* Eliminación segura de claves temporales
* Configurar respaldo seguro de UUID y LUKS
* Restaurar encabezado de LUKS en caso de corrupción
* Validaciones finales

### ¿Cuándo guardar los UUID?

**Después de crear particiones y cifrarlas (LUKS, ZFS, EFI):**

* Asegúrate de guardar tanto los UUID iniciales como los UUID de los dispositivos mapeados tras usar cryptsetup luksOpen.
* En este momento, los UUID están asignados y no cambiarán mientras no reformatees las particiones.

#### Guardar en un archivo temporal:

blkid > /mnt/etc/luks-keys/UUIDs\_backup.txt

#### Cifrar el archivo con GPG:

gpg --symmetric --cipher-algo AES256 /mnt/etc/luks-keys/UUIDs\_backup.txt

#### Almacenar en un dispositivo externo cifrado:

cp /mnt/etc/luks-keys/UUIDs\_backup.txt.gpg /media/usb/secure\_backup/

Opcional: Almacenar en un gestor de contraseñas seguro (KeePassXC, Bitwarden, etc.).

### Validar UUID antes de configuración de GRUB

Usa blkid para verificar los UUID guardados y asegúrate de que coincidan con las configuraciones en /etc/default/grub.

**Tras abrir particiones cifradas con cryptsetup luksOpen**

blkid > /media/usb/UUIDs\_backup.txt

Esto guardará la información de todas las particiones en un archivo que puedes copiar a un medio externo.

**Tras abrir las particiones cifradas con cryptsetup luksOpen**:

* Los dispositivos mapeados (/dev/mapper/<nombre>) tendrán UUID diferentes. Es útil guardarlos también.

blkid > /media/usb/UUIDs\_backup\_decrypted.txt

**Antes de configurar el GRUB**:

* Esto asegura que los UUID configurados en el archivo /etc/default/grub son correctos.
* En este momento, revisa que coincidan con las particiones que vas a configurar.

### Cómo guardar los UUID de forma segura

1. **Dispositivo externo seguro**:
   * Guarda los archivos de backup (UUIDs\_backup.txt y UUIDs\_backup\_decrypted.txt) en una USB cifrada o en un almacenamiento externo.
2. **Gestor de contraseñas**:
   * Copia los UUID al gestor de contraseñas que utilices (por ejemplo, Bitwarden o KeePassXC). Puedes crear un ítem con los siguientes datos:
     + Nombre de partición (ej. /dev/sda2)
     + UUID asignado
3. **Impresión física cifrada** *(opcional)*:
   * Si prefieres tener un respaldo físico, imprime los UUID cifrándolos previamente con una clave que solo tú conozcas. Puedes usar herramientas como OpenSSL:

echo "UUID\_de\_la\_partición" | openssl enc -aes-256-cbc -salt -out UUID\_encrypted.txt

1. **En un sistema de backups cifrado**:
   * Si ya tienes configurado un RAID o un servidor remoto seguro, puedes guardar el archivo ahí, asegurándote de que el transporte y almacenamiento estén cifrados.

### Valida los UUID periódicamente

Antes de finalizar la instalación, realiza pruebas para asegurarte de que los UUID son correctos:

1. Desbloquea las particiones.
2. Monta las particiones.
3. Verifica con blkid que los UUID coincidan con tu archivo de backup.

## Configura GRUB para usar LUKS:

### Configurar UUID de las particiones encriptadas

**Obtener los UUIDs de las particiones**:

Ejecuta el siguiente comando para obtener el UUID de cada partición (también de las particiones cifradas y ZFS):

blkid -o value -s UUID

##### Esto devolverá algo similar a lo siguiente:

/dev/sda1: UUID="123e4567-e89b-12d3-a456-426614174000"

/dev/sda2: UUID="abcdef01-2345-6789-abcd-ef0123456789"

/dev/mapper/cryptroot: UUID="d4e5f678-00a3-4b84-82de-c94791f2d123"

/dev/mapper/crypt-zfs1: UUID="ac9d812f-5783-4f62-b902-5c16adf0c840"

/dev/mapper/crypt-zfs2: UUID="d4e5f678-00a3-4b84-82de-c94791f2d123"

### Configurar los UUIDs en /etc/default/grub

blkid -o value -s UUID /dev/sda2 # Obtener UUID para cryptdevice

blkid -o value -s UUID /dev/mapper/vol-root # Obtener UUID para root

nano /etc/default/grub

**Configurar:**

GRUB\_CMDLINE\_LINUX="cryptdevice=UUID=<UUID\_de\_LUKS>:cryptroot root=UUID=<UUID\_de\_Root>"

Modifica la línea GRUB\_CMDLINE\_LINUX para incluir los UUID de las particiones cifradas y de root. Asegúrate de que coincidan con los UUID que obtuviste con blkid:

GRUB\_CMDLINE\_LINUX="cryptdevice=UUID=abcdef01-2345-6789-abcd-ef0123456789:cryptroot root=UUID=123e4567-e89b-12d3-a456-426614174000"

## Instalar y Configurar GRUB

### Instala el gestor de arranque en el sistema EFI:

sudo grub-install --target=x86\_64-efi --efi-directory=/boot/efi --bootloader-id=GRUB

#### Desactiva la detección de otros sistemas operativos:

Poner GRUB\_DISABLE\_OS\_PROBER en True, si solo tienes un sistema operativo y no necesitas detectar otros sistemas operativos. En cambio, si lo pones en False, es porque tienes múltiples sistemas operativos instalados y quieres que GRUB los detecte y añada al menú de arranque.

GRUB\_DISABLE\_OS\_PROBER=true

### Genera el archivo de configuración de GRUB:

sudo grub-mkconfig -o /boot/grub/grub.cfg

## Optimización y Configuración del RAID ZFS

### Crear Snapshots Automáticos

zfs snapshot raidz/data@autosnap

### Backup a Servidor Remoto o Dispositivo Externo

#### Enviar Snapshot Incremental:

zfs send -R raidz/data@snapshot\_name | ssh user@backupserver "zfs receive backup\_pool/data"

#### Backup a Dispositivo Externo:

zfs send -R raidz/data@snapshot\_name | zfs receive backup\_pool/data

### Verificar Backup

zfs list -t snapshot backup\_pool/data

zpool status backup\_pool

## Optimizaciones Finales

### Habilitar Servicios Clave

**Habilita NetworkManager para la conexión de red:**

systemctl enable NetworkManager

### Configura el borrado de claves y cachés al apagar el sistema:

**Crea un servicio en /etc/systemd/system/clear-cache.service:**

[Unit]

Description=Clear Cache at Shutdown

DefaultDependencies=no

Before=shutdown.target

[Service]

Type=oneshot

ExecStart=/bin/sync

ExecStart=/bin/sh -c "echo 3 > /proc/sys/vm/drop\_caches"

[Install]

WantedBy=shutdown.target

**Activa el servicio:**

systemctl enable clear-cache.service

### Pruebas Finales:

**Probar acceso a particiones cifradas**:

Asegúrate de que puedes acceder a las particiones cifradas de LUKS sin problemas:

cryptsetup luksOpen /dev/sda2 crypt-root

**Verificar snapshots ZFS**:

zfs list -t snapshot

## USERS

### Root password

passwd

### Añadir usuario del sistema

useradd -m -G wheel -s /bin/bash LaraCanBurn

passwd LaraCanBurn

su LaraCanBurn

sudo mkdir -p /etc/systemd/system/getty@tty1.service.d/ && echo -e "[Service]\nUser=LaraCanBurn " | sudo tee /etc/systemd/system/getty@tty1.service.d/override.conf && sudo systemctl daemon-reload && sudo systemctl restart getty@tty1 🡪 para iniciar el equipo con el usuario que hemos creado por defecto, en lugar de con root

### Configuración Sudo

EDITOR=nano visudo

## Uncomment to allow members of group wheel to execute any command

%wheel ALL=(ALL) ALL

### Habilitación de servicios básicos

### Configuración de Pacman.conf

Dentro de /etc/pacman.conf

* Descomentar la línea de Color
* Descomentar ParallelDownloads = 5
* Añadir ILoveCandy

Habilitar repositorios extra

* Descomentar [multilib]

sed -i -e '/^#Color$/s/^#//' \

-e '/^#ParallelDownloads = 5/s/^#//' \

-e '/^ParallelDownloads/a ILoveCandy' \

-e '/^\[multilib\]/,/Include/ s/^#//' \

-e '/^\[multilib-testing\]/,/Include/ s/^Include/#Include/' /etc/pacman.conf

sudo pacman -Sy

### Generar carpetas principales [root]

ls

sudo pacman -S xdg-user-dirs

xdg-user-dirs-update

ls root

### Generar carpetas principales LaraCanBurn

sudo -u LaraCanBurn xdg-user-dirs-update

ls /home/LaraCanBurn

### Eliminar beeps del sistema

**Desactivar el pitido en la BIOS de VMware**

VMware tiene una opción para desactivar el beep de la BIOS al iniciar la máquina virtual. Sigue estos pasos:

**🔹 Método gráfico (Fácil)**

1. **Apaga la VM**.
2. **Abre VMware Workstation/Player** y selecciona la VM.
3. **Haz clic en "Editar configuración de la máquina virtual"**.
4. Ve a la pestaña **"Opciones" → "Opciones avanzadas"**.
5. Busca la opción **"Deshabilitar pitidos"** y actívala.
6. Guarda los cambios y enciende la VM.

**🔹 Método manual (Avanzado)**

Si no ves la opción en la GUI, puedes hacerlo manualmente editando el archivo de configuración de la VM (.vmx):

1. **Apaga la VM**.
2. Abre la carpeta donde está almacenada tu máquina virtual.
3. Busca el archivo de configuración de la VM (NombreDeTuVM.vmx).
4. Ábrelo con un editor de texto (por ejemplo, nano en Linux o Notepad++ en Windows).
5. **Añade esta línea al final del archivo:**

mks.noBeep = "TRUE"

1. Guarda los cambios y **reinicia la VM**.

echo "blacklist pcspkr" | sudo tee /etc/modprobe.d/blacklist-pcspkr.conf 🡪 elimina el beep del arranque, pero las alertas del hardware seguirán activas.

sudo modprobe -r pcspkr

echo "set bell-style none" >> ~/.inputrc 🡪 Desactivar Beeps en Terminales (Bash, Zsh)

echo "setopt NO\_BEEP" >> ~/.zshrc 🡪 Para Zsh

mkdir -p ~/.config/kitty && echo "enable\_audio\_bell no" >> ~/.config/kitty/kitty.conf && kitty @ set-enable-audio-bell no 🡪 Para Kitty

echo "set visualbell" >> ~/.vimrc 🡪 Desactivar Beeps en Vim

echo "set beep off" >> ~/.nanorc 🡪 Desactivar Beeps en Nano

echo "xset -b" >> ~/.xinitrc 🡪 Desactivar Beeps en el Entorno Gráfico (X11: KDE, GNOME, XFCE)

xset -b

Mantener las Alertas Críticas del Hardware:

lsmod | grep snd 🡪 Verificar qué módulos de sonido están cargados

Esto mostrará los módulos relacionados con el sonido. Algunos de los más comunes en Linux son:

* snd\_hda\_intel (para la mayoría de las tarjetas de sonido integradas).
* snd\_pcm (para la gestión del sonido en el sistema).
* snd\_seq (para MIDI y secuencias de audio).
* snd\_usb\_audio (si usas una interfaz de audio USB).

Si accidentalmente desactivaste un módulo de sonido, puedes volver a cargarlo con:

sudo modprobe snd\_hda\_intel

(Sustituye snd\_hda\_intel por el nombre del módulo de audio que necesites).

**Verificar logs de errores críticos del sistema**:

journalctl -p 3 -xb

* Esto mostrará los errores **importantes del sistema** (por ejemplo, si un disco falla o hay problemas en el hardware virtual).

Si quieres verificar que los errores críticos se registran correctamente puedes **forzar una alerta del sistema** (sin riesgos, solo para prueba):

sudo dmesg --level=err

**Forzar una alerta del sistema** (sin riesgos, solo para prueba):

sudo dmesg --level=err

**Si tienes smartmontools instalado**, puedes verificar si el disco de la VM tiene errores (aunque sea virtual):

sudo smartctl -H /dev/sda

* Si responde PASSED, significa que no hay errores críticos en el disco.
* Si responde FAILED, entonces el sistema intentará generar una alerta si el hardware lo permite.

## Desmontar unidades

**Desmontar ZFS y otras particiones al salir de chroot:**

zfs unmount -a

umount -R /mnt

# INSTALL GRAPHICAL ENVIRONMENT

sudo pacman -S xfce4 xorg xorg-server lightdm lightdm-gtk-greeter

sudo pacman -S light-locker 🡪 permite configurar el tiempo de espera del bloqueo de pantalla

sudo pacman -S xfce4-power-manager  Para monitorear la temperatura y el estado de tu hardware.

sudo pacman -S network-manager-applet 🡪 complemento que permite gestionar conexiones de red

sudo pacman -S lightdm-gtk-greeter-settings 🡪 (revisar si es necesario)

sudo pacman -S lm\_sensors && sudo sensors-detect && sudo systemctl enable lm\_sensors && sudo systemctl start lm\_sensors  Gestor de energía para XFCE que te ayudará a gestionar la energía y ahorrar batería si estás en un portátil.

## Instalar emulador de la terminal

sudo pacman -S kitty

## Iniciar Linux Desktop

sudo systemctl enable --now lightdm

 **Appendices**

* Cómo cifrar un USB con LUKS
* Consideraciones de seguridad para claves y backups

# INSTALL ESSENTIAL SYSTEM PACKAGES

## Instalar paquetes básicos

sudo pacman -S neofetch lsb-release brave-bin libreoffice-still wget

## Instalación de VMWare Tools

sudo pacman -S open-vm-tools xf86-video-vmware xf86-input-vmmouse && sudo systemctl enable vmtoolsd && sudo systemctl start vmtoolsd

## Instalación de servidor de sonido y drivers de sonido

### Drivers

sudo pacman -S alsa-base alsa-firmware alsa-utils alsa-plugins alsa-oss alsa-lib alsa-tools alsamixergui alsa-firmware-loaders vol-umeicon-alsa 🡪drivers

sudo pacman -S pulseaudio pulseaudio-alsa pavucontrol paprefs 🡪servidor (pulseaudio||jack)

(revisar)

### Servidor

sudo pacman -S pipewire pipewire-pulse pipewire-jack pipewire-alsa

### Utilidades del sistema

sudo pacman -S htop ncdu tree

### Desarrollo y programación

sudo pacman -S git vim docker kubernetes-cli python python-pip nodejs npm

### Networking

sudo pacman -S openssh net-tools iftop

### Seguridad y privacidad

sudo pacman -S ufw gufw iptables fail2ban

sudo pacman -S ufw && sudo systemctl enable ufw && sudo systemctl start ufw && sudo ufw enable  Instalación y configuración de UFW

### Comodidades y personalización

sudo pacman -S realtime-privileges 🡪 Otorga privilegios de tiempo real, pudiendo mejorar el rendimiento y la precisión para tareas de audio o ingeniería.

sudo pacman -S timeshift 🡪 Permite realizar snapshots

### Multimedia

sudo pacman -S vlc

### Compresión y gestión de archivos

sudo pacman -S p7zip zip unzip tar

(añadir editor de texto, mirar si es major opcion notepad ++ con snapd o sublime text (comprobar si snapd es una buena opción y segura para mantener todo actualizado) )

## Instalar asistentes de AUR (Arch User Repository)

### Paru

sudo pacman -S paru

sudo pacman -S --needed base-devel  dependencias necesarias

git clone <https://aur.archlinux.org/paru.git>  clonar repositorio de Paru

cd paru  navega al repositorio clonado

makepkg -si  compila e instala Paru

### Yay

sudo pacman -S yay

git clone <https://aur.archlinux.org/yay.git>

cd yay

makepkg -si

## Instalar editor de Código Visual Studio Code

sudo yay -S visual-studio-code-bin

## Instalar minikube

sudo yay -S minikube

# INSTALL ESSENTIAL ENVIRONMENT PACKAGES

PRE-INSTALLING AND CONFIGURING

Configuración previa:

Clic derecho 🡪 Desktop Settings

* Icon type: None

Clic derecho 🡪 Window Manager Tweaks

* Desmarcar en Cycling: Draw frame around selected windows while cycling
* Marcar en Placement: At the center of the screen
* Desmarcar en Compositor: Show shadows under dock windows

Descargar “update-xfce-bigsur” y descomprimir en descargas

Seleccionar la carpeta de Wallpapers, la copiamos y la pegamos en Pictures

Clic derecho 🡪 Settings

* Background: folder > Other… e importamos la carpeta de Pictures > wallpapers

Seleccionamos el tema y lo aplicamos.

Vamos a la carpeta de /home/LaraCanBurn y seleccionamos Documents, Music, Pictures, Public, Templates y Videos y hacemos sobre ellos clic derecho Send to side panel.

INSTALLING AND CONFIGURING THEME – ICONS – CURSORS – FONTS

THEME

sudo pacman -S gtk-engine-murrine sassc

cd Downloads/

git clone <https://github.com/vinceliuice/WhiteSur-gtk-theme>

Comprobamos que se haya descargado con ls

Instalamos el script con 🡪 ./install.sh -c dark -c light

ICONS

cd

git clone <https://github.com/vinceliuice/WhiteSur-icon-theme>

cd WhiteSur-icon-theme/

./install.sh

CURSORS

git clone https://github.com/vinceliuice/WhiteSur-cursors

ls

cd WhiteSur-cursors/

ls

./install.sh

Nos abrimos dos directorios gráficamente, uno con el /home y otro con /Downloads

Dentro de home le damos a “Show hidden files” para que muestre los archivos ocultos

Por otro lado abrimos el directorio de Downloads y update-xfce-bigsur/fonts

Dentro de /home creamos el directorio .fonts y copiamos los archivos de Downloads

Una vez hecho, hacemos clic derecho en el escritorio y vamos a:

* Settings > Appearance > Style, y seleccionamos el tema WhiteSur dark
* Icons > WhiteSur dark
* Fonts > Default fonts > San Francisco Display Regular
* Default Monospace Font > Iosevka Term Regular

Desde la barra de tareas, haremos clic derecho en Applications, e iremos a:

Settings > Window Manager

Seleccionamos en Style > WhiteSur dark

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

En Tittle font > San Francisco Display Medium

INSTALLING GLOBAL MENU

Install Vala-Appmenu manjaro/arch :

pamac build vala-panel-appmenu-common-git vala-panel-appmenu-registrar-git vala-panel-appmenu-xfce-git

sudo pacman -S appmenu-gtk-module

Execute this command after enable vala-appmenu:

xfconf-query -c xsettings -p /Gtk/ShellShowsMenubar -n -t bool -s true

xfconf-query -c xsettings -p /Gtk/ShellShowsAppmenu -n -t bool -s true

CONFIGURING XFCE PANEL

Descarga el File de este link: <https://www.pling.com/p/1529470/>

Vamos a la carpeta de Downloads y extraemos el archivo

Copiamos la carpeta que nos ha dejado en /home/LaraCanBurn/.local/share/

Creamos una carpeta llamada applications y copiamos el contenido del extraíble

Acto seguido iremos a la ruta /home/LaraCanBurn/.config/ y creamos la carpeta de menu y copiamos el archivo xpple.menu

En la barra de tareas hacemos clic derecho y seleccionamos Panel > Panel Preferences y selecciona en Items todas las opciones de Separator, eligiendo la última y añadiendola.

Luego Separator, lo ponemos arriba de la lista, en este caso sería internal name separator-13 y eliminamos de la lista Window Buttons, Workspace Switcher.

Añadimos Keyboard Layouts y lo situamos después de separator. Añadimos también Weather Update mediante el buscador, y AppMenu Plugin

Interfaz de usuario gráfica, Aplicación, Chat o mensaje de texto, Teams

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.Pantalla de celular con imagen de la pantalla de un celular

El contenido generado por IA puede ser incorrecto.

En Applications Menu seleccionamos este apartado para ir a configurarlo

Interfaz de usuario gráfica, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Aquí, una vez le demos a la carpeta, vamos a /home y a Show Hidden Files y vamos a la carpeta .config > menu y seleccionamos el archivo xpple.menu .Seleccionamos AppMenu Plugin y se abrirá un desplegable en la parte superior de la pantalla, marcamos la casilla Use bold application name.

Seleccionamos la casilla Weather Update y volvemos a darle al cuadro inferior, se nos abrirá una pestaña donde cambiamos el Location name.

Después en Keyboard Layouts seleccionamos Show layout as text y en Layout name seleccionamos language, cambiamos el tamaño del Widget al 50, y en Manage Layout seleccionamos la opción globally.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

En la parte de Scrollbox, cambiamos el tema.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Seleccionamos Action Buttons, y desmarcamos las casillas como se ve en la imagen.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Seleccionamos PulseAudio Plugin y volvemos a seleccionar el cuadrado, aquí en Media Players, comprobamos que se vea el VLC que hemos instalado previamente.

Seleccionamos Applications Menu y editamos las opciones con el cuadrado, seleccionamos el Icono.

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Seleccionamos la opción de “Select icon from” > All Icons y en Search icon ponemos “start” y seleccionamos el icono de Apple.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Por último desmarcamos la casilla de Show Button Title

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Haremos clic derecho sobre la barra de tareas y le daremos a propiedades

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Aquí en la parte de Expand, desmarcamos la casilla

Imagen de la pantalla de un video juego

El contenido generado por IA puede ser incorrecto.

Volvemos a hacer clic sobre la manzana de Apple, aquí vamos a realizar el mismo proceso, esta vez marcando el estilo como Transparente.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Una pantalla de telefono

El contenido generado por IA puede ser incorrecto.

En la parte superior derecha, vamos a hacer de nuevo clic derecho, sobre el icono que aparece a la derecha de la hora del sistema.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Una pantalla de un celular

El contenido generado por IA puede ser incorrecto.

En la configuración del Panel, vamos a seleccionar Panel 2 y vamos a darle al icono del – para eliminar la barra de tareas inferior.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

INSTALLING AND CONFIGURE ROFI LAUNCHER

Instalamos rofi 🡪 sudo pacman -S rofi

Copiamos la carpeta que se encuentra en /home/LaraCanBurn/Donwloads/update-xfce-bigsur llamada rofi a /home/LaraCanBurn/.config/

INSTALLING AND CONFIGURE CONKY

pamac build conky-lua-nv

sudo pacman -S jq curl

INSTRUCCIONES VIDEO CONFIGURACIÓN DE ENTORNO

Download Setup file : [https://www.opencode.net/lsteam/xfce-...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbFN5VnEwbURJQl9mY296YW9BZDdCdF9fbGFXQXxBQ3Jtc0tuMTJBcUstaG5HV0dwblM4eWhBUzA0M3BGLUdKTGdsZ2ozdUk4ZFNTLXE0RUQ1dlE5bTZqcUlsbUx4LUU3SkQyb196VjBQUExJUHRubmhDOGltUGVqOFhMdldmdi1UM0hBTHZzTHl5U3BMRG41ZXJMbw&q=https%3A%2F%2Fwww.opencode.net%2Flsteam%2Fxfce-big-sur-setup-file&v=uvvoJU69uNo" \t "_blank) ==================================

CHAPTERS OF THIS VIDEO

==================================

[00:00](https://www.youtube.com/watch?v=uvvoJU69uNo&t=0s) - Intro

[00:10](https://www.youtube.com/watch?v=uvvoJU69uNo&t=10s) - Final Result

[02:24](https://www.youtube.com/watch?v=uvvoJU69uNo&t=144s) - Initial Setup

[04:03](https://www.youtube.com/watch?v=uvvoJU69uNo&t=243s) - Installing and Configuring themes, icons, cursors and fonts

<04:05> - Installing WhiteSur-gtk-theme - [https://github.com/vinceliuice/WhiteS...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbENSRVhxMm9GTlFCZ0FXcmZaOHZSQ1BDYV9aZ3xBQ3Jtc0trdmhGdHVxTGhfUUNMSUpjYzF2aDJrOGZtUHNZQlZPaGNoam1fYlpSYkZwVnJBcm5jQXE3NDBlRXltQjdud3FnZ1FKZG5zNGltcmhNeEhLaDRURmc5d0dPWGNhRVRwalk5MW9XWG5WVWVFUGhCUEx4TQ&q=https%3A%2F%2Fgithub.com%2Fvinceliuice%2FWhiteSur-gtk-theme&v=uvvoJU69uNo" \t "_blank)

<05:49> - Installing WhiteSur-icon-theme - [https://github.com/vinceliuice/WhiteS...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbDdMeUlHd2dYeF9oM3NnTC0tdF9sZTZ4R3pnQXxBQ3Jtc0ttVUR6aFo3WERKNm5JRXM2U2pHWGMwaEVpaGNoNUFRN2NpYjRaLS1PRUM5Q1h5dVF0UFpVZHQxdXp1ZjBfNkU0bHBhWi0xRnFmZVI3cEtlOVVjRVJ6bVZWam5rMm9rU1NsUUpUNEdIa01pZHVROGRrNA&q=https%3A%2F%2Fgithub.com%2Fvinceliuice%2FWhiteSur-icon-theme&v=uvvoJU69uNo" \t "_blank)

<06:27> - Installing WhiteSur-cursors - [https://github.com/vinceliuice/WhiteS...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0FuWEZwbDZCMC1zdEh5QmFjMzlGMHAxTDZkZ3xBQ3Jtc0tuV3hER0lIT2RuMUxiMU1TUzhVaXIySkJJNXFDNXhHcnRrNDFXdXl6MFpsNlVzME1MOERzazdxQXYwNkJlVGVsZG5Jczd3dWZ6aWVaRnlLVGN5VDI0Yk1YTHd1UWxnWHNLUFJEc2JRSFhoSW1BZkNqUQ&q=https%3A%2F%2Fgithub.com%2Fvinceliuice%2FWhiteSur-cursors&v=uvvoJU69uNo" \t "_blank)

[07:02](https://www.youtube.com/watch?v=uvvoJU69uNo&t=422s) - Installing Fonts

[08:52](https://www.youtube.com/watch?v=uvvoJU69uNo&t=532s) - Installing Global Menu Install Vala-Appmenu manjaro/arch :

pamac build vala-panel-appmenu-common-gitvala-panel-appmenu-registrar-git vala-panel-appmenu-xfce-git

sudo pacman -S appmenu-gtk-module

Execute this command after enable vala-appmenu:

xfconf-query -c xsettings -p /Gtk/ShellShowsMenubar -n -t bool -s true

xfconf-query -c xsettings -p /Gtk/ShellShowsAppmenu -n -t bool -s true

[10:41](https://www.youtube.com/watch?v=uvvoJU69uNo&t=641s) - Configuring Xfce Panel | Download Xpple Menu : [https://www.pling.com/p/1529470/](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbE53YjVSaTE4UmEwNXBfaWNLcTR5MzJUTzZxQXxBQ3Jtc0ttYVQ4ZDhYNEd6ekFZYURGcnRtYUFaZU94OFFSaHZNTWgxUzNOeG1yZTVHcXZUTFAzUHJCMGhKckhkVHIyZm5LTm8zOVFXcmhWejVwZWRwTUFUMUZybzNkQ2lwME9kWXV0Z0gwbkpsQXVyVVV3R09DOA&q=https%3A%2F%2Fwww.pling.com%2Fp%2F1529470%2F&v=uvvoJU69uNo" \t "_blank)

[16:35](https://www.youtube.com/watch?v=uvvoJU69uNo&t=995s) - Installing and Configuring Plank Dock sudo pacman -S plank

[19:35](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1175s) - Installing and Configuring Rofi Launcher sudo pacman -S rofi Source : [https://github.com/adi1090x/rofi](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa2s0SXlhVThOZE1MRC10NjZSZzdldzUwYWZsZ3xBQ3Jtc0tuNGVNYWdIdUNvOXg2NUg1MXQ1YXkwb2JrRXN1dUFQbDdYY19kelRUSWZBQ1pXUXJtSkxwemdVR1RjRWF5MFZaZjFrTDZKRENXOExKZjlVVU81ZEFVR0pWY1lTeE0tM21iSTdhLUV5NV9QZ0FjUTRCaw&q=https%3A%2F%2Fgithub.com%2Fadi1090x%2Frofi&v=uvvoJU69uNo" \t "_blank)

[20:29](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1229s) - Installing and Configuring Ulauncher

pamac build ulauncher

[23:03](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1383s) - Installing and Configuring Conky

pamac build conky-lua-nv sudo pacman -S jq curl

[24:46](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1486s) - Installing and Configuring LightDM-Webkit2-Greeter

sudo pacman -S lightdm-webkit2-greeter Theme : [https://github.com/manilarome/lightdm...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbHZyamNmYXB4T0dKRHllV0plbHJpZ09SX3l0UXxBQ3Jtc0tudGxaUzluOTZzcWZuRTVrdVN4V096WEw5aVUwYnZkMlJjcXZZVHRMTXMxUjZIZzhMYml5aXNlcTdHSGxVbjhEQ29lWk9oMGZOZnc3MTFCX3N3blJSUjRQQ19oYmJNVU1RTU1lLUJPUnFQZ3FjNnVLOA&q=https%3A%2F%2Fgithub.com%2Fmanilarome%2Flightdm-webkit2-theme-glorious&v=uvvoJU69uNo" \t "_blank)

[28:29](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1709s) - Installing and Configuring Picom Compositor

pamac build picom-ibhagwan-git

[30:17](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1817s) - Customize Firefox Web Browser

[32:01](https://www.youtube.com/watch?v=uvvoJU69uNo&t=1921s) - Customize Xfce terminal

[35:09](https://www.youtube.com/watch?v=uvvoJU69uNo&t=2109s) - Installing Nautilus File Manager

sudo pacman -S nautilus

<36:12> - Installing Comice Control center

git clone [https://github.com/libredeb/comice-co...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbE15UDFiOUtyc21sMzZmTUlsUHBGQ3psMjZEQXxBQ3Jtc0ttM19vckIwLTVTdGdndm1UWUl4Q2M5NUlEa2lqTmJkeUduSktXWjdqRVBycXlJa0h3cE1uRzN5VkpMNmlpVlMzX3d0NEZ4d2g2enk5cmVZNWJRRE91MmtYXzB2REp3b0VCRWxCV0U4TGQ0ekNiZTN1bw&q=https%3A%2F%2Fgithub.com%2Flibredeb%2Fcomice-control-center.git&v=uvvoJU69uNo" \t "_blank)

sudo pacman -S util-linux gsettings-desktop-schemas wireless\_tools iproute alsa-utils dbus-python

sudo pacman -S python-pip

sudo pip3 install -r requirements.txt

./comice-control-center

[40:09](https://www.youtube.com/watch?v=uvvoJU69uNo&t=2409s) – RAM usage before/after setup

==================================

https://github.com/vinceliuice/WhiteSur-firefox-theme

# BIBLIOGRAPHY

[https://www.youtube.co](https://www.youtube.com/watch?v=68z11VAYMS8)[m/watch?v=68z11VAYMS8](https://www.youtube.com/watch?v=68z11VAYMS8)  **Arch Linux: A ℂ𝕠𝕞𝕗𝕪 Install Guide - DenshiVideo**

[https://www.youtube.com/watch?v=kXqk91R4Rw](https://www.youtube.com/watch?v=kXqk91R4RwU)[U](https://www.youtube.com/watch?v=kXqk91R4RwU)  **Arch Linux: An 𝔼𝕟𝕔𝕣𝕪𝕡𝕥𝕖𝕕 Guide - DenshiVideo**

<https://www.youtube.com/watch?v=jmVcOSWVf1Y>**Instalación de Arch con cifrado de disco (LUKS) - jlxip**

<https://www.youtube.com/watch?v=H0s6aLMlJ88> **Curso GNU/Linux SHL: Instalación de ArchLinux UEFI + Cifrado de disco - Security Hack Labs**

<https://www.youtube.com/watch?v=g-DR0n-hILc> **instalacion de Arch Linux UEFI paso a paso !! - Daniel Hernandez**

[https://www.youtube.com/watch?v=uv](https://www.youtube.com/watch?v=uvvoJU69uNo)[voJU69uNo](https://www.youtube.com/watch?v=uvvoJU69uNo)  **UPDATE : How to Customize Your Xfce Desktop Look Like macOS Big Sur | Version 2.0 - LinuxScoop**

<https://www.youtube.com/watch?v=Zy4U0zk2J6U>  **BlackArch, ¿difícil de instalar? ¿muy pesada? ¿DESACTUALIZADA? - Desmintiendo MITOS - Security Hack Labs**

## Resultado esperado con lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 512G 0 disk

├─sda1 8:1 0 512M 0 part /mnt/boot/efi

└─sda2 8:2 0 511.5G 0 part

└─crypt-root 254:0 0 511.5G 0 crypt

├─vol-root 254:1 0 503.5G 0 lvm /mnt

└─vol-swap 254:2 0 8G 0 lvm [SWAP]

sdb 8:16 0 100G 0 disk

└─crypt-zfs1 254:3 0 100G 0 crypt

sdc 8:32 0 100G 0 disk

└─crypt-zfs2 254:4 0 100G 0 crypt

### Explicación de Cada Componente

1. **Disco /dev/sda (512G)**:
   * **/dev/sda1**: Partición EFI (512M) montada en /mnt/boot/efi.
   * **/dev/sda2**: Partición principal cifrada con LUKS (crypt-root).
     + **vol-root**: Volumen lógico LVM para la raíz, montado en /mnt.
     + **vol-swap**: Volumen lógico LVM para swap.
2. **Discos /dev/sdb y /dev/sdc (100G cada uno)**:
   * Ambos están cifrados con LUKS (crypt-zfs1 y crypt-zfs2).
   * Estos dispositivos son utilizados por ZFS para crear un pool RAID-Z (raidz).

### Validación

Puedes verificar los pools ZFS y los datasets montados con los siguientes comandos:

1. **Verificar el estado del pool ZFS**:

zpool status

Salida esperada:

pool: raidz

state: ONLINE

scan: none requested

config:

NAME STATE READ WRITE CKSUM

raidz ONLINE 0 0 0

crypt-zfs1 ONLINE 0 0 0

crypt-zfs2 ONLINE 0 0 0

1. **Verificar datasets montados**:

zfs list

Salida esperada:

NAME USED AVAIL REFER MOUNTPOINT

raidz 100G 80G 500M /mnt/data

raidz/root 10G 70G 1G /mnt

raidz/data 20G 60G 10G /mnt/data

Esto confirmará que las particiones y configuraciones están correctamente montadas y funcionales.

zfs send -R raidz/data@snapshot\_name | ssh user@backupserver "zfs receive backup\_pool/data"

zfs list -t snapshot backup\_pool/data

zpool status backup\_pool

zpool scrub backup\_pool

1. **Asegurarte de que el swap está habilitado:**

swapon –show

1. **Verificación de Cifrado en ZFS**

Para verificar que un dataset de ZFS está cifrado, puedes usar el siguiente comando:

zfs get encryption raidz/securedata

Este comando te mostrará el estado del cifrado del dataset. Si el dataset está cifrado, verás algo como esto:

NAME PROPERTY VALUE SOURCE

raidz/securedata encryption aes-256-gcm local

* **VALUE** debería mostrar el algoritmo de cifrado que configuraste, como aes-256-gcm.
* **SOURCE** indicará si es una propiedad local (en este caso, configurada por ti).

Sí, con la configuración optimizada que hemos planteado, se minimiza la posibilidad de que información sensible quede accesible una vez el sistema se apague. Aquí te detallo los puntos clave y mejoras específicas para garantizar esto:

**Medidas para Minimizar Información Persistente**

**1. Swap Cifrado**

* El swap está completamente cifrado dentro de un volumen LUKS independiente. Esto asegura que cualquier dato temporal que el sistema pueda escribir en la memoria de intercambio sea inaccesible tras un apagado.
* **Mejora adicional**: Configura LUKS para que borre automáticamente la clave del swap al apagar el sistema:
* swapoff -a
* cryptsetup close cryptswap

Puedes automatizarlo añadiendo scripts en /etc/rc.local o usando un servicio de systemd.

**2. Root Cifrado**

* La partición root está cifrada con LUKS. Esto significa que los datos del sistema y cualquier archivo no volátil que el sistema pueda guardar estarán protegidos tras el apagado, siempre que la clave no esté en memoria activa.

**3. ZFS con Cifrado**

* El RAID ZFS tiene cifrado nativo (AES-256-GCM) para garantizar que cualquier dato almacenado en los discos RAID sea inaccesible sin la clave.
* Además, los datos en tránsito hacia respaldos externos también pueden ser cifrados utilizando ZFS send/receive con claves.
* **Comando adicional** para descartar claves en RAM tras desmontar:
* zfs unload-key raidz/securedata

**4. Datos Temporales en RAM**

* Configura /tmp y otros directorios temporales para que usen un sistema de archivos basado en RAM (tmpfs), lo cual garantiza que los datos temporales desaparezcan al apagar el sistema:
  + Edita /etc/fstab:
  + tmpfs /tmp tmpfs defaults,noatime,mode=1777 0 0

Esto almacena temporalmente archivos en memoria en lugar de en disco.

**5. Limpiar Cachés al Apagar**

* Puedes configurar el sistema para que limpie la caché de disco y datos en memoria justo antes de apagar:
  + Crea un archivo de servicio en /etc/systemd/system/clear-cache.service:
  + [Unit]
  + Description=Clear Cache at Shutdown
  + DefaultDependencies=no
  + Before=shutdown.target
  + [Service]
  + Type=oneshot
  + ExecStart=/bin/sync
  + ExecStart=/bin/sh -c "echo 3 > /proc/sys/vm/drop\_caches"
  + [Install]
  + WantedBy=shutdown.target
  + Activa el servicio:
  + systemctl enable clear-cache.service

**6. Desmontar y Cerrar Particiones Automáticamente**

* Configura el sistema para desmontar automáticamente los volúmenes cifrados y cerrar los dispositivos LUKS al apagar:
  + Edita /etc/crypttab:
  + cryptroot /dev/sda2 none luks,discard
  + cryptswap /dev/sda3 none luks,discard
  + cryptzfs1 /dev/sdb1 none luks,discard
  + cryptzfs2 /dev/sdc1 none luks,discard
  + Asegúrate de que los dispositivos se cierren al apagar con un servicio de systemd.

**Confirmación de Seguridad**

* Tras apagar el sistema, intenta acceder manualmente a los discos (/dev/sda2, /dev/sdb1, etc.). Sin las claves LUKS o la clave del pool ZFS, los datos deberían ser inaccesibles.
* Además, puedes verificar que no quede información residual al reiniciar con comandos como hexdump para inspeccionar sectores del disco directamente:
* sudo hexdump -C /dev/sda2 | head

**Resumen**

Con las medidas propuestas:

1. **Root y swap cifrados**: Cualquier dato persistente está inaccesible sin la clave LUKS.
2. **RAID cifrado y optimizado con ZFS**: Los datos del RAID están protegidos con cifrado avanzado.
3. **Borrado de datos temporales y cachés**: Los datos en RAM y directorios temporales desaparecen al apagar.
4. **Automatización del cierre de claves y desmontaje**: Nada queda accesible una vez el sistema se apaga.

Esto garantiza que, incluso en caso de intrusión física, los datos no puedan ser recuperados sin las claves correctas.

Cuando tienes un sistema encriptado, tanto para los volúmenes del sistema como para los volúmenes de almacenamiento (por ejemplo, RAID 1 o RAID-Z), el proceso de hacer un **backup** y recuperarlos debe tomar en cuenta tanto la seguridad como la integridad de los datos. Aquí te proporciono dos enfoques para realizar **backups** en sistemas encriptados, destacando cuál es el más **óptimo** y **seguro**.

**1. Backup con Encriptación Desmontada (Método más seguro y simple)**

El **método más seguro** es realizar los backups **desmontando la encriptación temporalmente** y luego copiando los datos no cifrados a un medio de respaldo. Esto reduce la complejidad de los respaldos y asegura que los datos no estén cifrados cuando realices la copia.

**Pasos para hacer un backup seguro con encriptación desmontada:**

1. **Desmontar la encriptación (si se aplica a volúmenes)**:
   * Si tu sistema está encriptado con **LUKS**, abre el volumen de la partición cifrada (por ejemplo, el sistema root, o las particiones de datos RAID).
2. cryptsetup open /dev/sda2 crypt # Para abrir el volumen cifrado

**Nota**: Si tienes múltiples volúmenes cifrados (como el volumen RAID y particiones adicionales), debes abrir cada uno.

1. **Montar las particiones desencriptadas**:
   * Una vez desencriptadas las particiones, puedes montarlas de la manera usual:
2. mount /dev/mapper/crypt-root /mnt
3. **Realizar el backup**:
   * Puedes usar herramientas como **rsync**, **tar** o incluso hacer una imagen completa de las particiones para asegurarte de que los datos se copien correctamente. Un comando típico con **rsync** sería:
4. rsync -aAXv /mnt/ /path/to/backup/ # Respaldar el sistema montado a un destino de backup

O usar **tar** para crear un archivo comprimido:

tar -czvf /path/to/backup/system-backup.tar.gz /mnt/

Si estás usando un RAID o un sistema de archivos como **ZFS**, puedes también utilizar herramientas específicas de ese sistema para hacer un respaldo de volúmenes o datasets.

1. **Verificación del backup**:
   * Verifica que los archivos respaldados estén completos y sean legibles. Puedes intentar montar el backup en una máquina diferente para asegurarte de que los datos no están corruptos.
2. **Desmontar y cerrar la encriptación**:
   * Una vez que hayas realizado el respaldo, desmonta y cierra los volúmenes de manera segura:
3. umount /mnt
4. cryptsetup close crypt

**2. Backup de Datos con Encriptación Activa (Método más complejo pero más seguro a largo plazo)**

Si prefieres mantener los datos cifrados en todo momento (incluyendo los backups), puedes hacer un **backup cifrado**. Esto implica que el backup se realiza de manera que los datos cifrados no se desencriptan en ningún momento. Este método es más **complejo**, pero puede ser útil en entornos donde la seguridad es crucial.

**Pasos para hacer un backup de datos cifrados (sin desencriptar):**

1. **Copia de los volúmenes cifrados**:
   * Para realizar un backup sin desencriptar, puedes copiar directamente las particiones cifradas o los volúmenes cifrados a un medio de almacenamiento. Utilizar herramientas como dd o partimage es útil aquí, ya que hacen una copia exacta del volumen en su estado cifrado.

Ejemplo con **dd**:

dd if=/dev/sda2 of=/path/to/backup/sda2.img bs=64K status=progress

Esto crea una copia exacta de la partición /dev/sda2 en el archivo sda2.img. El archivo de backup será **cifrado** y podrá restaurarse como un volumen cifrado más tarde.

1. **Usar herramientas de backup cifrado**:
   * Si deseas realizar un backup de directorios sin desencriptar, una opción sería usar herramientas de **backup cifrado**. Una opción común es **Duplicity**, que realiza copias incrementales cifradas, manteniendo los datos cifrados durante todo el proceso.
2. duplicity /mnt file:///path/to/backup/ --encrypt-key=YOUR\_ENCRYPTION\_KEY

Aquí **duplicity** realiza una copia cifrada de los archivos dentro de /mnt a un destino de respaldo. Puedes utilizar tu clave de encriptación para cifrar y restaurar los datos cuando sea necesario.

**3. Método Óptimo y Seguro**

**La forma más segura y óptima** de hacer un backup de datos en un sistema encriptado es **desmontar la encriptación temporalmente** (siguiendo los pasos de la primera opción), realizar el backup de los archivos desencriptados y luego cerrar la encriptación. Esto permite que el backup se haga de forma confiable, fácil de verificar y sin comprometer la seguridad del sistema cifrado.

* **Pros**:
  + Los datos son más fáciles de recuperar y verificar.
  + No es necesario usar herramientas especializadas para cifrar y desencriptar los backups.
  + La copia de seguridad será legible en un sistema diferente sin necesidad de descifrado previo.
* **Contras**:
  + Requiere acceso al sistema encriptado (necesitarás abrir y montar la encriptación antes de hacer el backup).
  + Menos conveniente si necesitas backups constantes sin intervención.

**4. Restauración de Backups**

Cuando necesites restaurar los backups, el proceso variará dependiendo de si estás trabajando con un backup cifrado o no:

* **Restauración desde un backup desencriptado**:
  + Simplemente monta el sistema de backup, y usa **rsync**, **tar**, o **cp** para devolver los archivos a las particiones correspondientes.
* **Restauración desde un backup cifrado**:
  + Desencripta la copia del volumen o los archivos antes de restaurarlos en el sistema encriptado.

Ejemplo de restauración de un backup cifrado con dd:

dd if=/path/to/backup/sda2.img of=/dev/sda2 bs=64K status=progress

Y luego, asegúrate de montar el volumen desencriptado y verificar que la restauración fue exitosa.

**Recomendación**

Si deseas **seguridad a largo plazo** y **facilidad de recuperación** en caso de que los datos sean necesarios, te recomiendo el **método de hacer un backup desencriptado** (paso 1 de la primera opción), y luego utilizar **rsync** o **tar** para la copia de archivos. Este método te permitirá restaurar rápidamente los datos y verificar su integridad.

**Usa cifrado sólo cuando sea necesario** (por ejemplo, con **duplicity** o backups cifrados completos) si necesitas mantener los datos completamente cifrados en los respaldos.

Si tienes más preguntas o necesitas más detalles sobre alguna de las opciones, no dudes en preguntar. ¡Estoy aquí para ayudarte!

La configuración propuesta es bastante robusta y considera varios aspectos clave para mantener la **integridad del sistema**, **seguridad** y **privacidad**, especialmente en caso de intrusiones. Sin embargo, como en cualquier diseño de sistema, siempre hay margen para optimización según las prioridades específicas. Aquí está un análisis de la solución y algunas consideraciones adicionales:

**Fortalezas de la Configuración**

1. **Cifrado Integral (LUKS)**:
   * **Ventaja**: Toda la partición raíz (incluyendo el swap) está cifrada, lo que asegura que ningún dato sensible quede accesible si el sistema está apagado.
   * **Relevancia**: El cifrado también se extiende al RAID ZFS, protegiendo los datos almacenados.
2. **ZFS con RAID-Z**:
   * **Ventaja**: ZFS ofrece integridad de datos mediante sumas de verificación, lo que asegura que cualquier corrupción de datos sea detectada y, si es posible, corregida automáticamente.
   * **Privacidad y Seguridad**: Cifrar el RAID ZFS asegura que incluso si los discos son extraídos físicamente, no puedan ser leídos.
3. **EFI Separada**:
   * **Ventaja**: Tener la partición EFI separada minimiza el riesgo de corrupción o compromisos en el gestor de arranque.
4. **Swap encriptado**:
   * **Ventaja**: Evita la posibilidad de que datos sensibles queden en la memoria de intercambio sin protección.
5. **Compatibilidad y Escalabilidad**:
   * Uso de GPT permite mayor compatibilidad con hardware moderno y flexibilidad para futuras expansiones o configuraciones.

**Áreas de Optimización y Mejora**

1. **Protección del Gestor de Arranque (EFI)**:
   * **Riesgo**: Aunque la partición EFI no está cifrada (porque el gestor de arranque necesita acceder a ella), sigue siendo un vector potencial de ataque.
   * **Mitigación**: Utilizar un gestor de arranque seguro como GRUB con una contraseña configurada. Además, habilitar **Secure Boot** si es compatible con tu hardware.
2. **Integridad del Sistema**:
   * Implementar **verificación de firmas GPG** en tus configuraciones importantes o snapshots de ZFS para detectar cambios no autorizados.
   * Configurar **journaling** o habilitar la función de ZFS para rastrear modificaciones.
3. **Swap en Root vs. Archivo de Swap**:
   * **Alternativa**: Considera usar un archivo de swap dentro del sistema de archivos cifrado en lugar de una partición LVM separada. Esto puede simplificar la gestión del espacio y mantener la privacidad.
4. **RAID-Z vs RAID-Z2 o RAID-Z3**:
   * Si la disponibilidad y recuperación de datos son críticas, considera aumentar la redundancia (RAID-Z2 o RAID-Z3) con discos adicionales. RAID-Z1 es más susceptible a fallos múltiples durante la reconstrucción.
5. **Seguridad Física y Red**:
   * Habilitar un sistema de detección de intrusiones como **fail2ban** o **iptables/nftables** para proteger el acceso remoto.
   * Implementar medidas de seguridad física como discos con **SED (Self-Encrypting Drive)** o módulos TPM para el cifrado de LUKS.

**Privacidad y Seguridad Mejorada en Caso de Intrusión**

* **Ataque en el Sistema Activo**:
  + Utiliza ZFS para crear snapshots regulares con políticas de retención estrictas. Los snapshots no pueden ser modificados, lo que dificulta los ataques como ransomware.
* **Ataque en Sistema Apagado**:
  + Toda la partición de datos y el sistema operativo están cifrados, por lo que un atacante necesitaría las claves para acceder a los datos.
* **Gestión de Claves**:
  + Asegúrate de usar un método seguro para gestionar las claves de LUKS (por ejemplo, un archivo de claves almacenado en un hardware seguro o autenticación multifactor).

**Resumen: Más Optimización y Seguridad**

La configuración actual es **muy segura y funcional**, especialmente si priorizas la privacidad en caso de intrusiones. Sin embargo, podrías considerar estas mejoras adicionales:

1. **Secure Boot y GRUB con contraseña** para proteger el arranque.
2. **Snapshots de ZFS** como medida de recuperación contra ataques activos.
3. Redundancia adicional con **RAID-Z2** si agregas más discos.
4. Monitorización y protección de red con herramientas como **fail2ban**.
5. Revisión periódica de logs y configuraciones.

Si implementas estas mejoras, tendrás un sistema extremadamente seguro, con un balance óptimo entre privacidad, seguridad e integridad de datos.

Comprobar UUID (hacer ahora)

Comprobar que el orden de la instalación es el correcto (revisar)

Comprobar que las optimizaciones finales están en orden correcto (revisar)

Conversación dev sda2 luksaddkey y orden de respaldo de luks (hacer ahora)