Davante

Actualización, mantenimiento, y recuperación de sistema.

Actividad 7

Título del índice

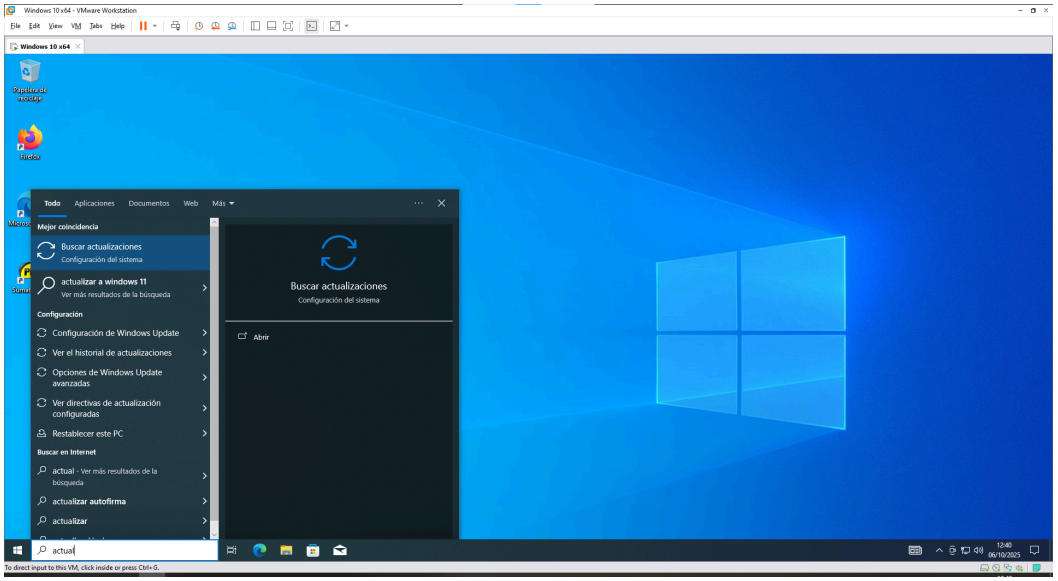
Introducción

La siguiente documentación detalla el proceso de mantenimiento básico del sistema operativo de Windows y Linux, junto con una comparativa de ambos procesos. Este mantenimiento incluye la actualización del sistema, la recuperación y restauración del sistema, el mantenimiento de los drivers, gestión de ficheros de inicio, y gestión de registros en Windows o ficheros de configuración en Linux.

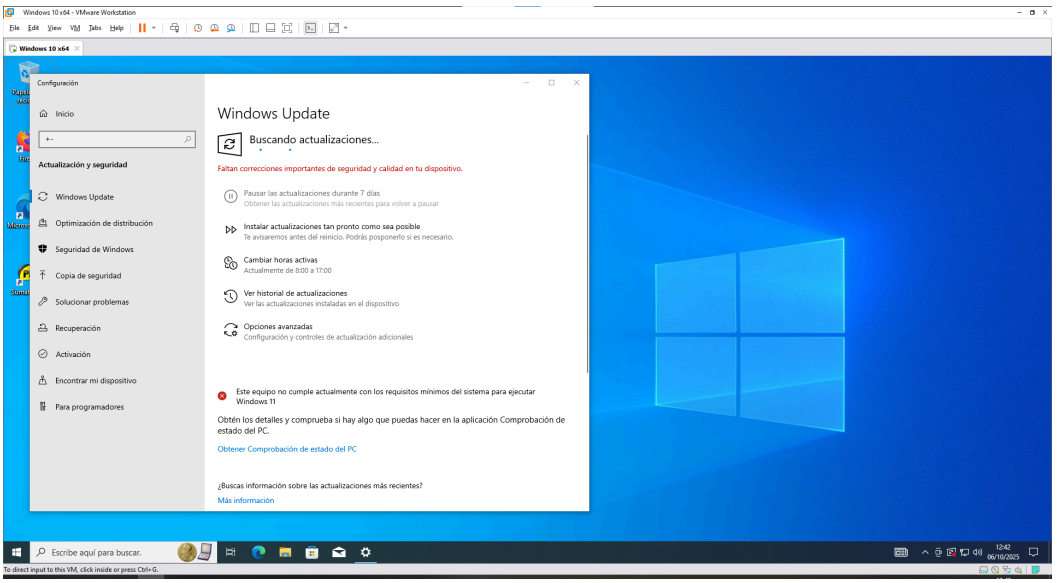
Windows

1. Actualización del sistema

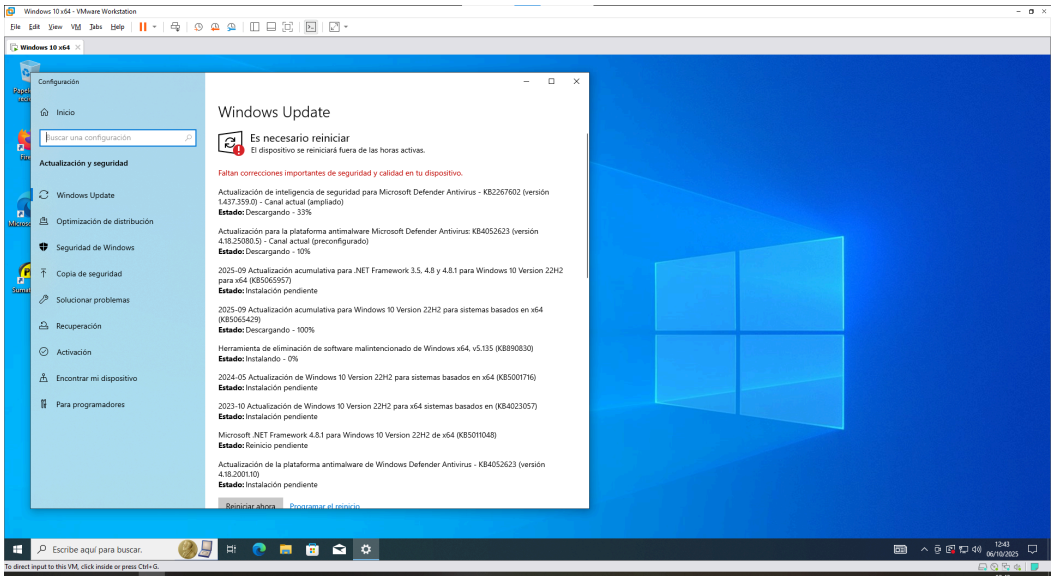
a. Actualizar Windows es sencillo, la mayoría del proceso es automático. Primero, busca “Buscar Actualizaciones” y entra en el menú.



b. Espera a que Windows busque y descargue todas las actualizaciones pendientes.



c. Cuando todas las actualizaciones estén descargadas, te pedirá reiniciar el equipo. Puedes programar el reinicio para una hora más cómoda si hace falta o reiniciarlo ahora.

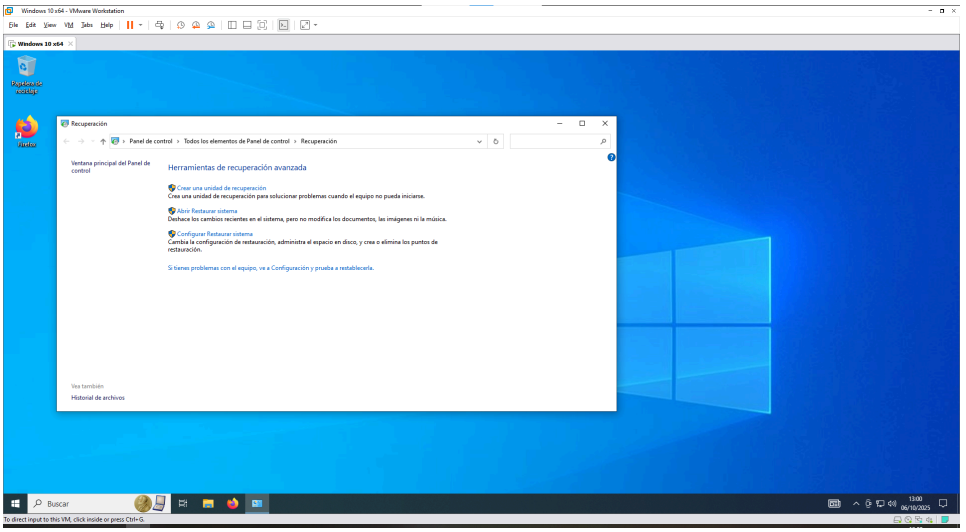


d. Verás varias pantallas azules y porcentajes. Espera a que se vuelva a iniciar en la pantalla de inicio de sesión. Asegúrate de que el equipo no pierda la corriente durante el proceso.

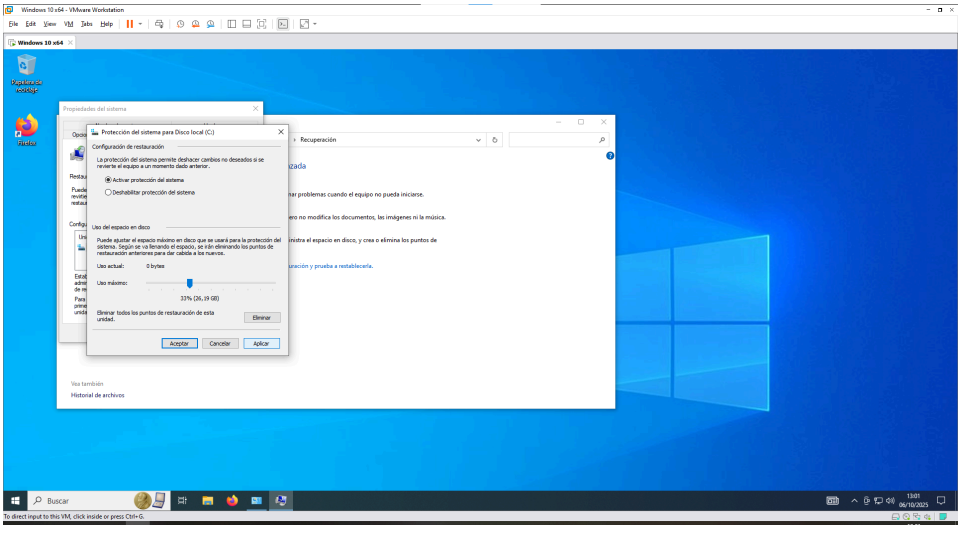
2. Recuperación y restauración

2.1. Creación de un punto de restauración:

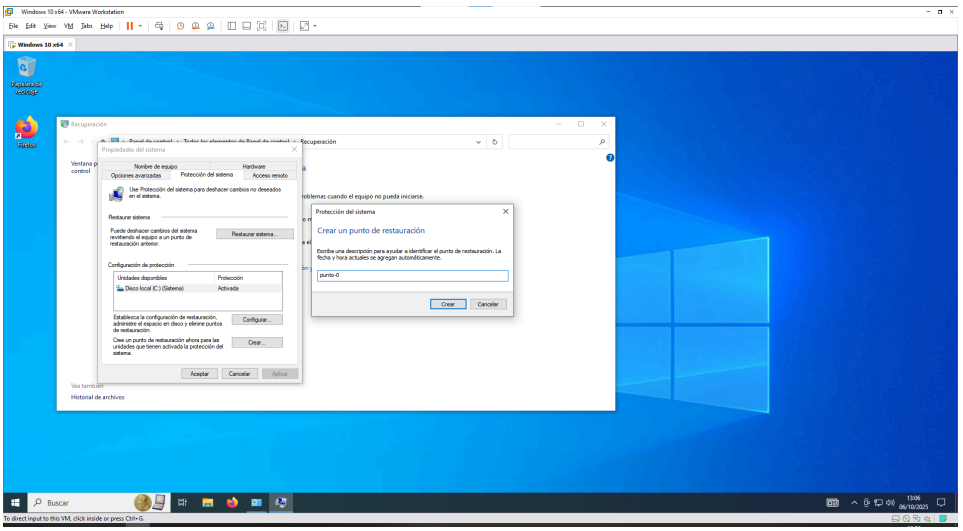
i. Para acceder a las opciones de recuperación de Windows, busca “restauración” y haz clic en “Configurar restaurar sistema”.



ii. Después, selecciona el disco al que quieres hacerle copias de seguridad y entra en “Configurar”. En la configuración, activa la protección del sistema y selecciona la cantidad máxima de espacio que puede usar el sistema para copias de seguridad y aplica los cambios.



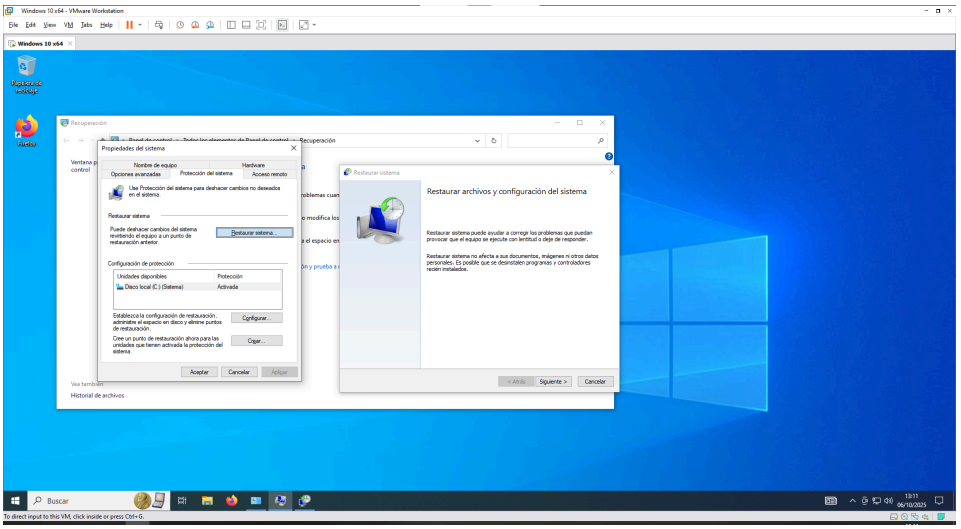
iii. En la pantalla de propiedades del sistema, entra en “Crear…” y dale un nombre al punto de restauración. Luego, dale a “Crear”.



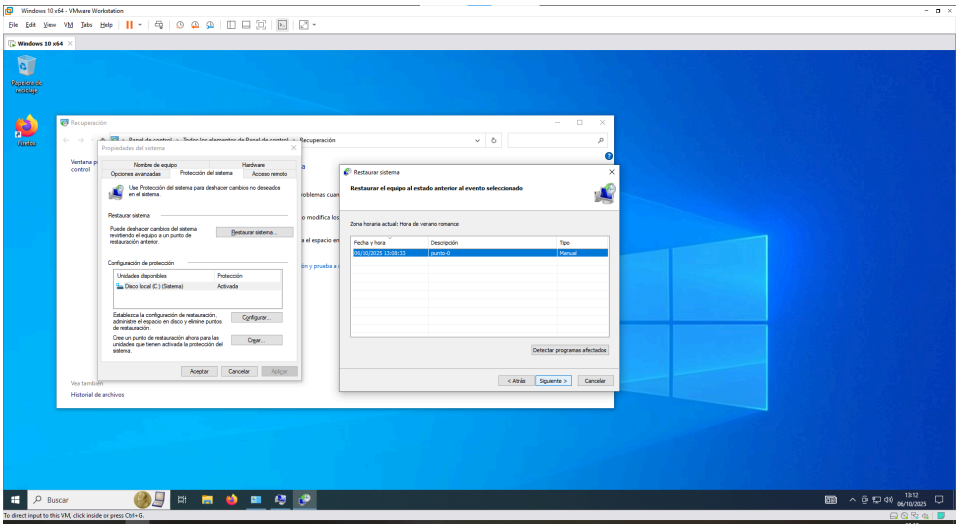
iv. Espera a que finalice el proceso.

2.2. Recuperación desde un punto de restauración:

v. En la pantalla de Propiedades del sistema, entra en “Restaurar sistema”.

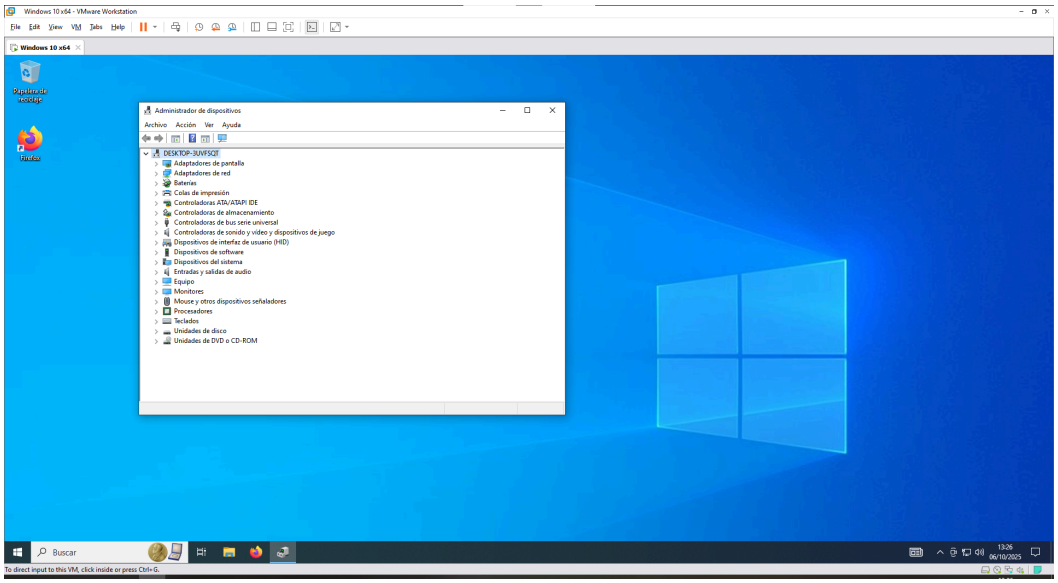


vi. En la siguiente pantalla, selecciona el punto de restauración deseado y continúa hasta finalizar el proceso. El sistema se reiniciará en el estado en el que estaba cuando se hizo el punto de restauración.

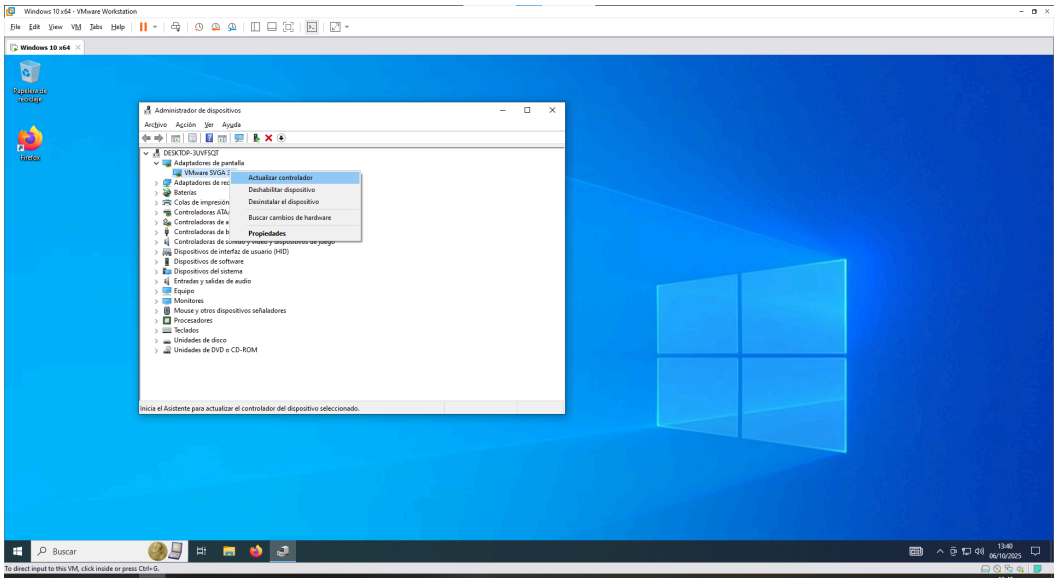


3. Mantenimiento de los controladores

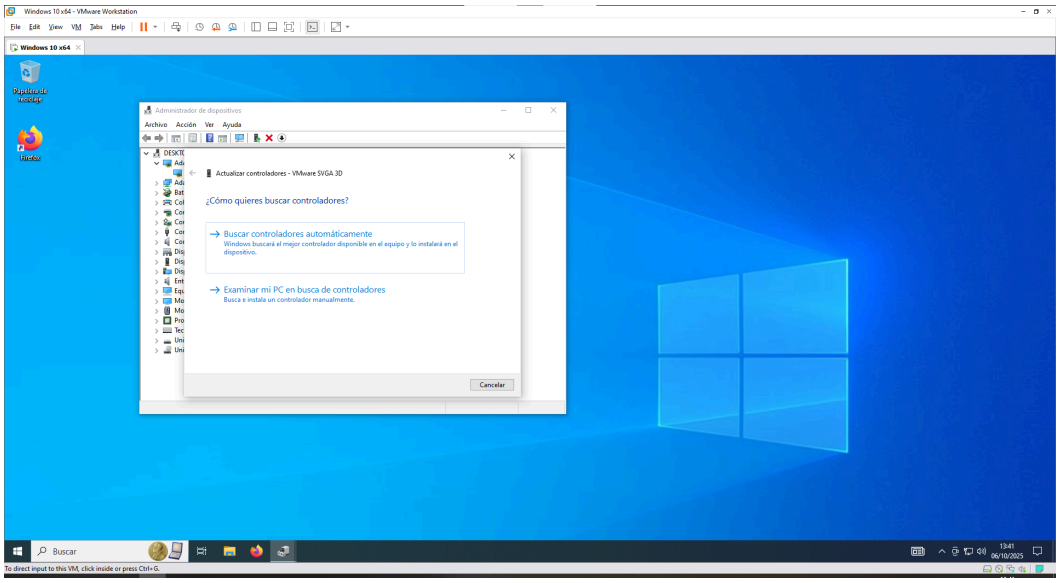
a. Para empezar, abre el administrador de dispositivos. Desde aquí gestionaremos todo lo relacionado con el hardware y los drivers.



b. Ahora, despliega la categoría deseada y selecciona el dispositivo al que le quieres actualizar los controladores. Hay que ir uno por uno, y el proceso debería de ser el mismo en todos los controladores. Haz clic derecho en el dispositivo y pulsa “Actualizar controlador”



c. Ahora selecciona “Buscar actualizaciones automáticamente”. En algunos casos, será necesario buscar los drivers en la página oficial del fabricante y seleccionarlos a través de la segunda opción.



d. Windows instalará los drivers si encuentra una actualización disponible.

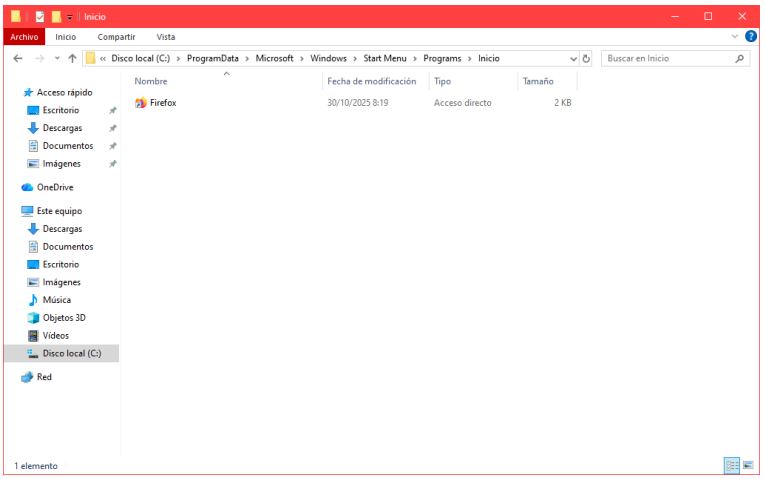
4. Ficheros de inicio

Los ficheros de inicio de Windows son los archivos que carga el sistema al arranque. Este sistema es relativamente sencillo, ya que se limita a cargar todo lo que encuentra en varios directorios específicos. El directorio `%ProgramData%\Microsoft\Windows\Start Menu\Programs\Startup` afecta a todos los usuarios, mientras que el directorio `%userprofile%\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup` afecta únicamente al usuario al que se le aplique.

Para modificar estos ficheros, hay varios métodos. El más directo es el siguiente:

e. Entra en el explorador de archivos e ingresar la dirección de antes

f. Arrastrar dentro los archivos de programa que queramos que se ejecuten al inicio, ya sean accesos directos o archivos en formato .exe. Así es como se



vería si quisiéramos que se abra Firefox al encender el equipo:

5. Registro del sistema

El registro del sistema es una base de datos jerárquica que almacena las configuraciones y preferencias del sistema completo. Este es un componente de gran relevancia para el sistema, ya que es, como su nombre indica, el encargado de registrar todo lo que concierne al sistema, desde los registros del kernel y los drivers hasta las aplicaciones de terceros e interfaces gráficas.

La estructura básica del registro funciona así:

- HKEY_CLASSES_ROOT (HKCR)
  - Combina las definiciones de tipo de archivo de `HKEY_LOCAL_MACHINE\Software\Classes` y la vista personalizada del usuario en `HKEY_CURRENT_USER\Software\Classes`.
  - Gestiona cómo Windows maneja archivos, extensiones y protocolos.
- HKEY_CURRENT_USER (HKCU)
  - Almacena configuraciones específicas del usuario actual que ha iniciado sesión.
  - Es dinámico y se fusiona con los datos de HKCR para personalizar el entorno.
- HKEY_LOCAL_MACHINE (HKLM)
  - Contiene configuraciones globales del sistema, hardware instalado, y servicios. No depende de un usuario específico.
- HKEY_USERS (HKU)
  - Agrupa los perfiles de usuarios en el equipo, permitiendo acceso a configuraciones pasadas o alternativas.
  - Incluye directorios como `Documents and Settings` y perfiles modernos en `NTUSER.DAT`
- HKEY_CURRENT_CONFIG (HKCC)
  - Específico de sistemas basados en Windows NT. Almacena configuraciones del hardware activo durante el inicio.

Linux

En sistemas operativos basados en Linux, la mayor parte del mantenimiento básico del sistema se realiza desde una terminal de comandos, similares entre distintos sistemas operativos. Esta práctica se realizará en Ubuntu 25.04, basado en Debian, por lo que todos los comandos utilizados serán idénticos en otros sistemas como Debian o Linux Mint. Los comandos no relacionados con gestionar paquetes también deberían ser compatibles con cualquier sistema basado en Linux.

1. Actualización del sistema

La actualización de sistema, en Ubuntu, utiliza la misma secuencia de comandos que actualizar programas por separado, pero sin especificar ningún programa, por lo que actualiza todos a la vez. Los comandos a realizar serían:

```
sudo apt update -y
```

a. Actualiza los repositorios locales. Esto significa que se descarga una lista de programas y sus versiones actuales de los servidores de Ubuntu, lo que permitirá al siguiente comando comprobar si hay versiones nuevas de paquetes para actualizar. Este comando, desglosado, sería:
   i. sudo: Super User DO. Ejecuta el siguiente comando como administrador. Nos pedirá la contraseña al ejecutarlo.
   ii. apt: El gestor de paquetes de Ubuntu, una interfaz sencilla para dpkg.
   iii. update: Actualiza los repositorios locales.
   iv. -y: Confirma directamente para que no nos pregunte más adelante.

```
sudo apt upgrade
```

b. Busca actualizaciones disponibles, las descarga, y las aplica. Este es el comando principal para actualizar software en Ubuntu. Tras lanzarlo nos va a mostrar los cambios a realizar. Puede parecer mucho lío, pero es recomendable leer detenidamente estos cambios, ya que en algunas ocasiones desinstala o modifica programas o dependencias que podrían romper partes del sistema. Leerlo antes de tiempo puede permitirnos resolver los conflictos antes de que se rompa todo y sea mucho más complicado de arreglar. Este comando, desglosado, sería:
   i. sudo: Super User DO. Igual que antes.
   ii. apt: Gestor de paquetes de Ubuntu. Igual que antes.
   iii. upgrade: Busca actualizaciones y las aplica.

1.1. Cambio de versión de sistema en Ubuntu

Debido a cómo se organizan las versiones de sistema de Ubuntu, hay dos actualizaciones "principales" al año, conocidas como *releases*. Como se dan en abril y octubre, las versiones de Ubuntu se nombran según el número del año y el mes. Como tenemos la versión 25.04 del sistema y ya está disponible Ubuntu 25.10; vamos a actualizar al nuevo sistema. Esta parte es exclusiva de Ubuntu y tiende a diferir mucho entre distintas distribuciones de Linux.

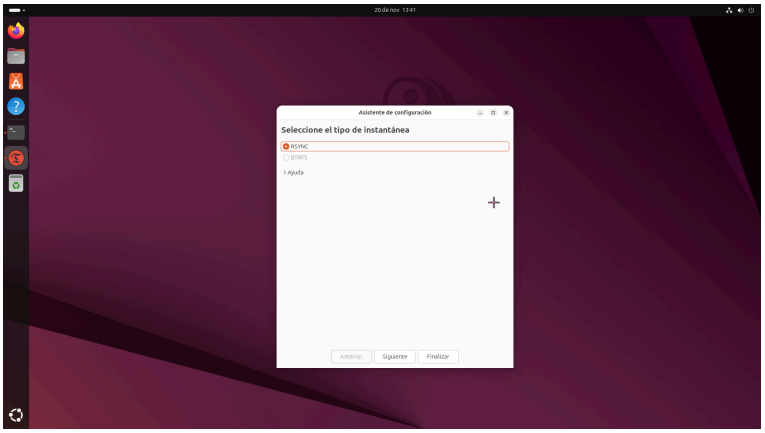
Para actualizar a la nueva *release* de Ubuntu tan solo hay que abrir una terminal y ejecutar el comando: `sudo do-release-upgrade` y confirmar escribiendo `y` cuando nos pregunte.

2. Recuperación y restauración

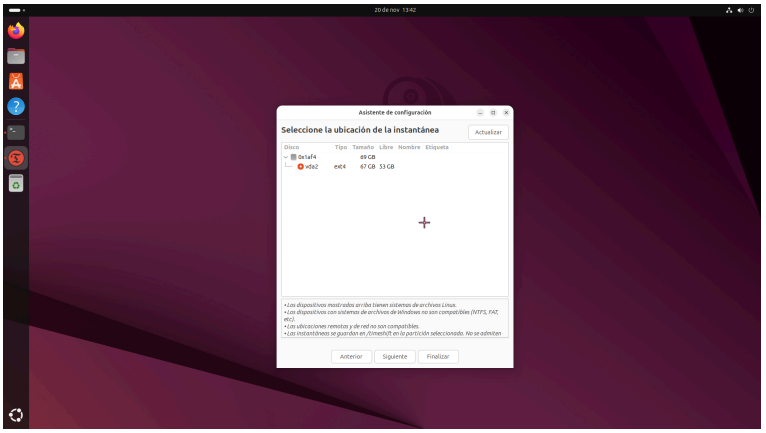
Ubuntu no trae preinstalada ninguna herramienta de restauración del sistema por defecto, pero podemos instalarla de forma sencilla. Para poder continuar, instalaremos el programa Timeshift utilizando apt: `sudo apt install timeshift`

Usaremos timeshift desde la interfaz gráfica, para mayor facilidad.

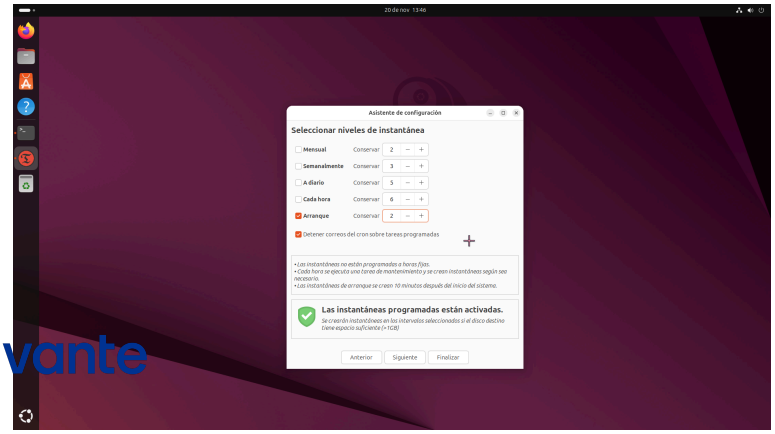
a. Lo primero que vemos es esta pantalla. En la mayoría de casos, será mejor seleccionar “RSYNC” y continuar.



b. La siguiente pantalla, en nuestro caso con un único disco, la vamos a pasar.

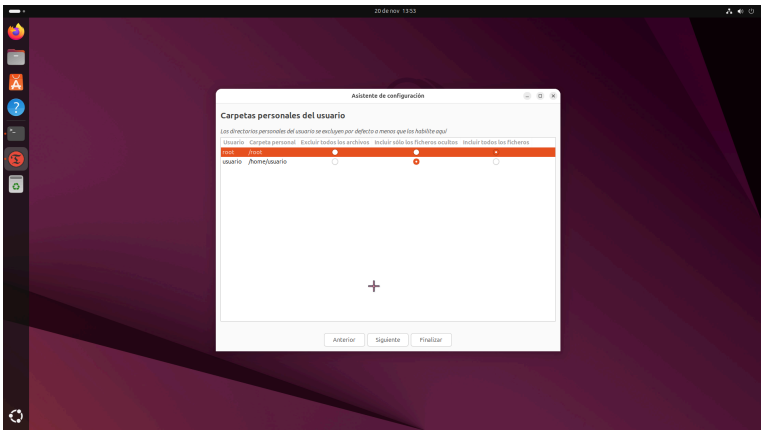


c. Esta pantalla es, probablemente, la más importante. Aquí seleccionaremos la frecuencia con la que timeshift realiza copias de seguridad y cuántas de cada tipo va a almacenar antes de eliminar las copias más antiguas. En mi caso, al ser una máquina virtual en la

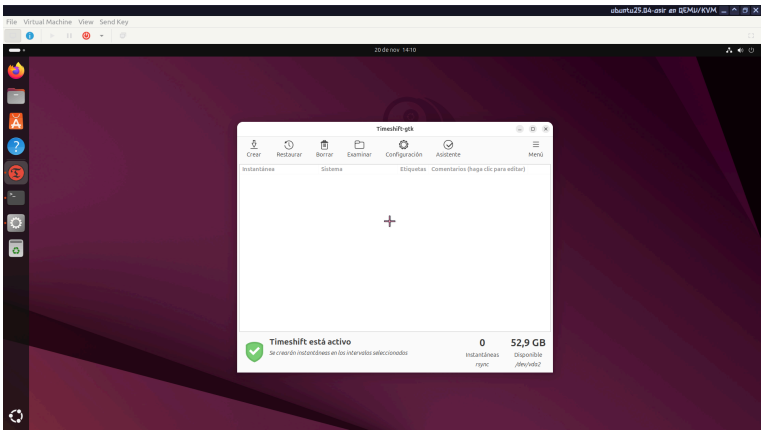


que realizo distintos trabajos cada vez, prefiero que se cree una copia cada vez que abro la máquina y no demasiadas, para no saturar el almacenamiento que le he dedicado.

d. En la siguiente pantalla seleccionamos los archivos que queremos almacenar de cada usuario. Por defecto, se excluyen todos los archivos, por lo que no se copiaría nada. Podemos elegir entre excluir todo (no copiar nada), incluir ficheros ocultos (los *dotfiles*, que veremos más adelante), o incluir todos los archivos. Como quiero copiar principalmente programas y configuraciones, he decidido almacenar todos los datos de root y sólo los archivos de configuración del usuario.

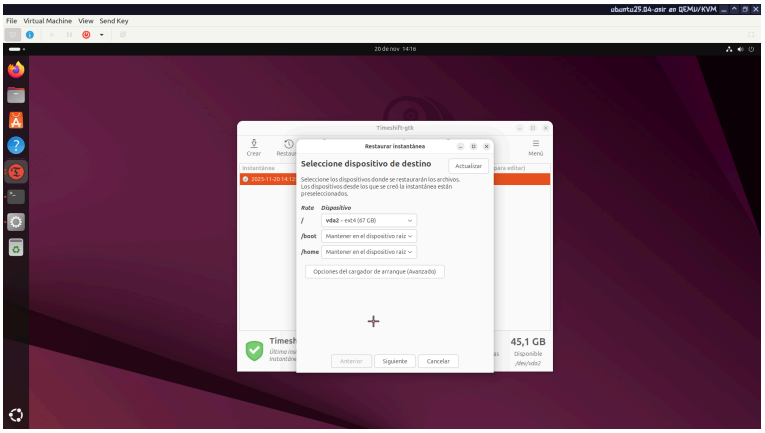


e. Finalizamos el proceso y nos saldrá el menú principal de Timeshift.



f. Para crear manualmente una copia de seguridad, sólo hay que hacer clic sobre “crear” y se guardará una instantánea del sistema en su estado actual. Debido a que de fondo estamos utilizando RSYNC, un software de copias de seguridad incrementales, la primera copia que hagamos será más pesada y más lenta de crearse que las demás. Esto es normal y esperado.

g. Para restaurar una copia de seguridad, la seleccionamos y hacemos clic en “restaurar”. Esto abrirá la siguiente ventana:



h. Seleccionamos el disco que queremos restaurar y continuamos las siguientes pantallas, que contienen información para confirmar la restauración que se va a realizar. Tras finalizar esta parte, comenzará el proceso de restauración y se reiniciará el ordenador con los datos antiguos recuperados.

3. Mantenimiento de los controladores

Debido a cómo se organiza Linux, como proyecto, y a cómo está estructurado el kernel; la inmensa mayoría de drivers de Linux forman parte directamente del kernel o se distribuyen como módulos del mismo, con sólo contadas excepciones de manufacturadores que distribuyen sus drivers privativos se forma independiente.

Este modelo de distribución agiliza muchísimo las tareas de mantenimiento, ya que con actualizar el kernel a la última versión publicada podemos mantener actualizados todos los drivers sin mucha más preocupación. Además, este proceso es llevado a cabo por los desarrolladores de la distribución de Linux que estemos usando, y no suele hacerse de forma manual salvo en distribuciones especializadas como Arch o Gentoo. En nuestro caso, los responsables de mantener el kernel actualizado son los desarrolladores de Canonical, la empresa propietaria de Ubuntu.

En resumidas cuentas, para mantener actualizados los drivers sólo hay que seguir los pasos del punto anterior para actualizar el sistema operativo cada seis meses y seguir las instrucciones del distribuidor de drivers privativos en caso de ser necesario.

4. Ficheros de inicio

En el proceso de arranque de Linux intervienen dos componentes principales: el gestor de arranque o *bootloader* y el gestor de procesos. Estos dos componentes son esenciales para el inicio del sistema operativo. Comenzamos analizando el papel y funcionamiento del *bootloader*.

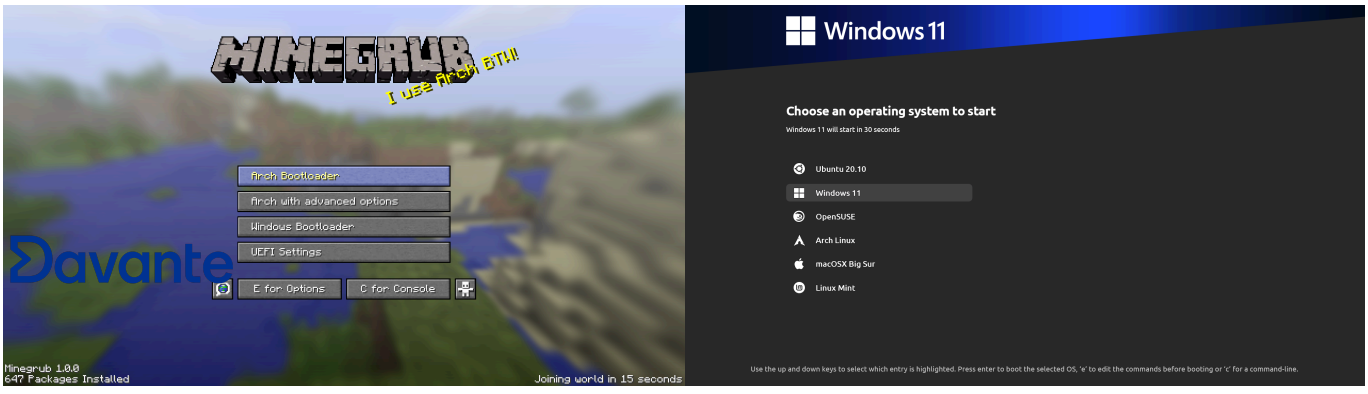
4.1. El gestor de arranque

El *bootloader* es un software cuyo fin es iniciar un sistema operativo. Este software tiene, además, la capacidad de cargar distintos sistemas operativos en memoria, proporcionando al usuario la capacidad de elegir un sistema entre varias opciones instaladas en distintos o incluso en el mismo dispositivo de almacenamiento. Este programa, al contrario que el *bootloader* privativo de Windows, tiene la capacidad de cargar cualquier sistema operativo que se encuentre en los discos conectados al ordenador. No importa si se trata de sistemas comerciales como Windows o sistemas libres o abiertos como Linux o BSD, e incluso herramientas de diagnóstico y recuperación independientes al sistema principal; que pueden ayudar a detectar o arreglar problemas de hardware o errores críticos del sistema operativo.

Las implementaciones actuales del *bootloader* más comunes son Grub2 y rEFInd. La diferencia principal entre ambos es que rEFInd está desarrollado exclusivamente con sistemas EFI en mente, mientras que Grub2 llevaba ya años soportando sistemas heredados BIOS y actualmente soporta ambos tipos de sistemas.

El *bootloader* se almacena en el directorio `/boot` y además, en sistemas UEFI, también almacena partes necesarias en `/boot/efi`. Estos directorios se suelen montar en particiones separadas al sistema raíz y, aunque no es obligatorio, es recomendado hacerlo siempre que sea posible. Al estar almacenados en un lugar accesible dentro del sistema y ser proyectos abiertos, tienen muchas formas de personalización tanto de configuraciones avanzadas para multi-arranques, como el tiempo de espera, si aparece el menú por defecto o sólo tras pulsar una tecla, y hasta modificaciones estéticas del menú de arranque.

Así es como, modificando la configuración en `/boot/grub/grub.cfg` y añadiendo los archivos correctos en `/boot/grub/themes/`, logré personalizar el aspecto del *bootloader* de mi disco de sistemas. Esto podría ser interesante también para empresas que quieran que sus sistemas que requieran *dual boot* mantengan la imagen de la empresa durante el proceso.



4.2. Gestor de procesos

Una vez se ha seleccionado el sistema que queremos arrancar, uno de los primeros módulos del sistema cargados por el kernel es el gestor de procesos. Este programa es lo que en sistemas basados en Unix se conoce como un *daemon* o demonio del sistema: programas especiales que se ejecutan en segundo plano como servicios y se reinician si se cierran de forma incorrecta. El gestor de procesos es, en resumen, el demonio encargado de gestionar el resto de demonios.

Pese a que hasta hace unos años predominaba el sistema init como gestor de procesos y sigue siendo el principal en sistemas basados en BSD, actualmente la mayoría de distribuciones de Linux utilizan un sistema más moderno llamado systemd.

Systemd es mucho más eficiente y sencillo de configurar, ya que funciona en base a archivos de configuración escritos en un lenguaje de marcas extremadamente sencillo. Estos archivos, una vez colocados en `/etc/systemd/system/` con un nombre terminado en *.service*, se pueden habilitar e iniciar para que comience a realizar el servicio que le hemos indicado. Para más detalles sobre la sintaxis y comandos necesarios, lo más recomendable es leer la documentación oficial de los servicios de systemd.

5. Archivos de configuración

Al igual que Windows, los sistemas operativos basados en Linux almacenan las preferencias globales del sistema en un directorio centralizado para mantener ordenadas las configuraciones a través del sistema. En el caso de Linux, en lugar de tener un registro, cada configuración se guarda en un archivo independiente. La mayoría de configuraciones básicas se almacenan en el directorio `/etc`. A continuación, vamos a presentar algunos de los archivos de configuración con los que más tenemos que estar familiarizados para controlar correctamente nuestro sistema:

`/etc/fstab`: Controla la forma en la que son montados los dispositivos durante el arranque del sistema. Especifica, de forma similar al comando `mount`, qué dispositivos van a ser montados, en qué formato, propietario, permisos, y en qué directorio.

`/etc/passwd`: Almacena la información sobre los usuarios del sistema: el nombre, la UID, la GID, y el directorio “home” del mismo. Antiguamente almacenaba la contraseña, pero no era un método seguro, por lo que ahora se almacenan en otro fichero.

`/etc/shadow`: Almacena las contraseñas de los usuarios encriptadas.

`/etc/group`: Almacena la información sobre los grupos.

`/etc/sudoers`: Almacena los usuarios y grupos con permiso para utilizar el comando sudo, que permite ejecutar comandos como administrador.

`/etc/default/grub`: Contiene las preferencias del gestor de arranque.

Comparativa

Como hemos visto, Windows y Linux siguen paradigmas completamente distintos a la hora de organizar la estructura de configuraciones y procesos del sistema. Las principales diferencias que podemos apreciar son:

- Windows se gestiona principalmente a través de interfaces gráficas, mientras que en Linux se utiliza más a menudo la línea de comandos.
- Linux, debido a que tiene muchas versiones distintas, reparte las configuraciones esenciales en una estructura de archivos estandarizada. Por otro lado Windows, al no tener que lidiar con muchas versiones desarrolladas por distintas organizaciones, puede establecer sus propias reglas y las mantiene centralizadas en el registro del sistema.
- La gestión de controladores e instalación de software también funcionan radicalmente distinto. En Windows se tienen que instalar los programas con ejecutables arbitrarios y la mayoría de drivers se instalan manualmente. Linux, al ser un proyecto colaborativo, ya incluye la mayoría de drivers como parte del kernel y el resto de software se instala desde repositorios verificados.
- Windows tiene un proceso de inicio más sencillo y pensado para funcionar bien para la mayoría de usuarios y necesidades. Linux, aunque es más complejo y a veces requiere configurar minuciosamente el sistema, ofrece muchas más opciones personalizables que pueden ayudar a construir sistemas a medida de nuestras necesidades.
- El proceso de recuperación de sistema es prácticamente idéntico en ambos sistemas.

Visto esto, podemos concluir con que ambos sistemas tienen sus formas de hacer cada cosa y sus complejidades, junto con ventajas y desventajas para distintos entornos.