Tema 4

Administración de software

Administración de Sistemas Operativos

Índice

[Esquema 3](#_Toc170129757)

[Material de estudio 4](#_Toc170129758)

[4.1. Introducción y objetivos. 4](#_Toc170129759)

[4.2. Instalación de paquetes de software. 5](#_Toc170129760)

[4.3. Acceso a directorios y gestión de repositorios. 12](#_Toc170129761)

[4.4. Referencias Bibliográficas. 18](#_Toc170129762)

[A fondo 18](#_Toc170129763)

[Entrenamientos 19](#_Toc170129764)

[Test 24](#_Toc170129765)

Esquema

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Material de estudio

4.1. Introducción y objetivos.

En el mundo de Linux, un programa se divide en unidades más pequeñas denominadas paquetes, cada uno con su función específica, y los paquetes a su vez son un conjunto de archivos necesarios para instalar y ejecutar programas.

La razón para el uso de los paquetes, en lugar del tradicional modelo de software de Windows, radica en la gran diversidad de distribuciones Linux que existen, cada una con sus diferencias, lo que hace imposible garantizar que una misma pieza de software pueda ejecutarse correctamente en todos los equipos. Linux soluciona este inconveniente con las denominadas dependencias o paquetes complementarios que se adaptan a cada sistema específico.

La descarga de los paquetes de software se lleva a cabo desde unos servidores especiales denominados “repositorios” que almacenan los archivos necesarios para la correcta instalación y ejecución del software.

Los objetivos que se pretenden conseguir al final de esta unidad didáctica son:

* **Comprender que son los paquetes de software**: Interiorizar la composición del software en sistemas operativos Linux y el por qué de su naturaleza.
* **Conocer que son los repositorios de Linux**: Entender el concepto de repositorio y como este se relaciona íntimamente con la administración de software en las diferentes distribuciones Linux
* **Aprender como agregar, actualizar y eliminar paquetes de software**: Familiarizase con los comandos utilizados para la gestión de software en Linux tales como **apt update**, **apt upgrade**, **apt install** o **apt remove**.
* **Familiarizarse con el fichero de configuración de repositorios**: Aprender como acceder al fichero y conocer su estructura interna así como la función de cada uno de sus elementos.
* **Conocer cómo se agregan y anular repositorios del sistema**: Estudiar las diferentes formas de gestionar los repositorios del fichero de configuración.

4.2. Instalación de paquetes de software.

En los sistemas operativos Linux, un programa o software se divide en paquetes que deben ser descargados e instalados antes de comenzar a utilizarse.

Conocemos la gestión de paquetes como el proceso de descarga, instalación, actualización y eliminación de los paquetes necesarios para el funcionamiento del software.

Para entender cómo gestionar el software en Linux, primero debemos hablar sobre el concepto de repositorio.

Los repositorios son servidores en internet encargados de guardar los ficheros (paquetes) necesarios para que el software funcione en nuestra máquina. Estos servidores son públicos y pueden ser consultados a través de nuestra consola de comandos en Linux para descargar los paquetes necesarios e instalar el software en cuestión. Mas adelante profundizaremos en la gestión de los repositorios.

Cabe destacar que existen numerosas distribuciones de Linux, algunas de ellas muy orientadas a una labor en concreto, lo que quiere decir que nuestro sistema operativo traerá ya instalados unos paquetes de software u otros en función de la distribución que estemos utilizando, por ejemplo, Ubuntu Linux trae por defecto instaladas algunas herramientas de edición de texto y de imágenes así como un reproductor de música y video, en cambio Kali Linux es una distribución enfocada en la rama de ciberseguridad y por tanto tiene preinstaladas una gran cantidad de software relacionado con el análisis y explotación de vulnerabilidades.

El comando utilizado para la gestión de paquetes en distribuciones basadas en Debian es [ apt ] seguido de la instrucción que queramos llevar a cabo. Para distribuciones basadas en Fedora usamos [dnf] y para distribuciones basadas en Arch se usa [pacman].

En los ejemplos usaremos el comando [apt], pero debemos tener en cuenta que necesitamos contar con permisos de super usuario para ejecutarlo por lo que será necesario incluir la sentencia “sudo” al principio del comando

A continuación se muestran las variantes más utilizadas del comando [ apt ] en una tabla con una definición resumida de sus funciones:

|  |  |
| --- | --- |
| **Comando** | **Función** |
| **apt update** | Actualiza la lista de paquetes disponibles. |
| **apt upgrade** | Descarga las últimas versiones de los paquetes |
| **apt full-upgrade** | Descarga las últimas versiones de los paquetes eliminando las versiones antiguas. Debemos tener especial precaución al utilizar este comando ya que puede modificar paquetes internos del sistema e incluso llegar a dejarlo inutilizable. |
| **apt install “paquete”** | Instala el paquete indicado y lo almacena en la ruta /usr/localbin o /usr/bin si son paquetes comunes a todos los usuarios |
| **apt reinstall** | Reinstala un paquete que ya está instalado. |
| **apt remove “paquete”** | Desinstala el paquete indicado |
| **apt purge “paquete”** | Desinstala y elimina todos los archivos del paquete indicado |
| **apt autoremove** | Elimina los paquetes dependientes de otros que ya no se encuentren instalados. |
| **apt list** | Devuelve un listado con el nombre, versión y arquitectura de los paquetes instados en el sistema ordenados alfabéticamente |
| **apt search “cadena”** | Busca y muestra los paquetes que tengan una cadena de caracteres coincidente con la que indicamos. |

Tabla 1. Resumen de comandos para gestión de paquetes. Fuente: elaboración propia.

Tanto el comando [ apt ] (Advanced Package Tool) como el comando [ apt-get ] son prácticamente equivalentes aunque este último funciona a un nivel más bajo.

Veamos más en detalle las instrucciones básicas para la instalación y desinstalación de software con ejemplos:

* **[ sudo apt update ]**

Este comando se encarga de consultar el estado actual de los repositorios para actualizar la lista de paquetes disponibles y sus versiones, pero no instala o actualiza ningún paquete. Esta lista recoge solo los resultados de los repositorios que tenemos definidos en el fichero de configuración sources.list que estudiaremos más adelante.

A continuación el resultado de la ejecución del comando:

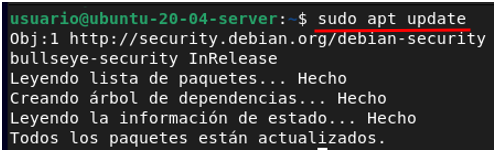


Figura 1. Resultado de ejecución del comando sudo apt update. Fuente: elaboración propia.

* **[ sudo apt upgrade ]**

Una vez el comando anterior ha descargado la lista de software disponible con sus últimas versiones, utilizamos el comando [ apt upgrade ] para instalar las nuevas versiones detectadas respetando la configuración del software cuando sea posible.

A continuación el resultado de la ejecución del comando:

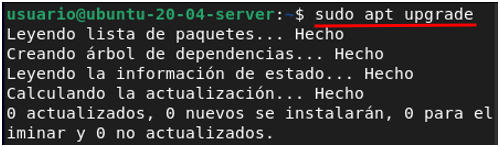


Figura 2. Resultado de ejecución del comando sudo apt upgrade. Fuente: elaboración propia.

Podemos resaltar que en ocasiones reutilizamos una imagen.iso antigua del sistema operativo para montar una máquina virtual y debemos tener en cuenta que el comando apt upgrade se pondrá al día con las versiones de todos los paquetes instalados por lo que el proceso puede alargarse más de lo esperado.

* **[ sudo apt install “nombre\_paquete” ]**

Este comando procederá con la instalación del paquete indicado siempre y cuando dicho paquete exista en uno de los repositorios configurados en nuestro sistema.

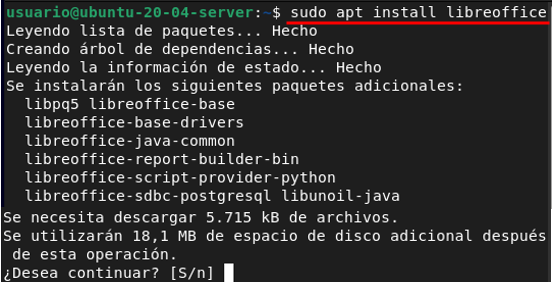


Figura 3. Resultado de ejecución del comando sudo apt install. Fuente: elaboración propia.

Podemos observar que el sistema nos solicita confirmación para continuar con la instalación después de informarnos de los paquetes y el espacio total necesario en disco, si pulsamos la tecla “S” y enter la instalación dará comienzo y se nos mostrará una barra de progreso como se aprecia en la siguiente imagen:



Figura 4. Barra de progreso de instalación en Debian. Fuente: elaboración propia.

Si conocemos el estado del disco y queremos omitir la petición de confirmación, podemos ejecutar el comando con la opción ( -y ) de la siguiente manera:

* **sudo apt install -y “nombre\_paquete”**

El sistema comenzará a ejecutará los pasos de la instalación sin solicitar confirmación en el orden habitual:

1. Leerá la lista de paquetes.
2. Creará la lista de dependencias.
3. Confirmará la información de estado.
4. Informará de los paquetes sugeridos, recomendados y su tamaño.
5. Comenzará la descarga.
6. Finaliza la descarga y desempaqueta el software.
7. Realiza la configuración inicial del software para dejarlo listo.

* **[ sudo apt remove “nombre del paquete” ]**

Este comando se encarga de eliminar los ficheros binarios correspondientes al paquete en cuestión pero no elimina los ficheros de configuración, los de datos ni los demás paquetes dependientes de este denominados “dependencias”.

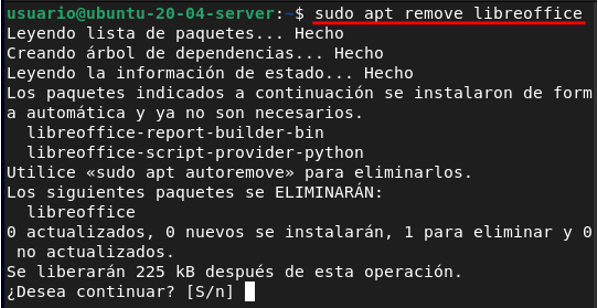


Figura 5. Resultado de ejecución del comando sudo apt remove. Fuente: elaboración propia.

Esta opción se utiliza cuando queremos desinstalar un programa pero conservar su fichero de configuración o los archivos de datos relacionados que hayamos creado en nuestra carpeta de trabajo personal.

El comando apt remove también es compatible con la opción ( - y ) que omite la petición de permiso para continuar con la instalación: sudo apt remove -y “nombre\_paquete”

* **[ sudo apt purge “nombre\_paquete” ]**

Este comando eliminará todos los ficheros relacionados con el paquete en cuestión, paquete y ficheros de configuración, pero mantiene las dependencias que se instalaron cuando descargamos el paquete.

Los ficheros de configuración y de datos que se encuentran en las carpetas de trabajo de los usuarios también se mantienen intactos.

Esta opción se utiliza cuando queremos empezar de nuevo y configurar desde cero un programa.

* **[ sudo apt autoremove “nombre\_paquete” ]**

Este comando eliminará todos aquellos paquetes “huérfanos” que se encuentren en el sistema.

En algunas ocasiones se instalan paquetes “accesorios” al paquete principal del software denominados “dependencias” que no son eliminadas con el comando apt remove ni con el comando apt purge.

El comando autoremove nos permite eliminar todos los paquetes instalados de forma automática para satisfacer dependencias de otros paquetes que actualmente no existen o no son necesarios.

Cabe destacar que no es recomendable utilizar el comando autoremove en todas las ocasiones ya que puede desembocar en el borrado de paquetes importantes que comprometan la estabilidad del sistema operativo.

Ahora que conocemos como gestionar el software desde la línea de comandos, vamos a nombrar algunas herramientas con interfaz gráfica que realizan la misma función de gestión pero de una forma más amigable con el usuario no técnico.

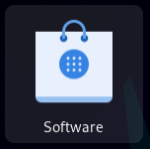
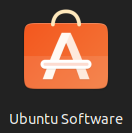
1. Debian utiliza la aplicación “Software” para instalar programas desde paquetes.



Figura 6. Interfaz herramienta “software” en Debian. Fuente: elaboración propia

1. Ubuntu utiliza una herramienta prácticamente idéntica llamada “Ubuntu Software”

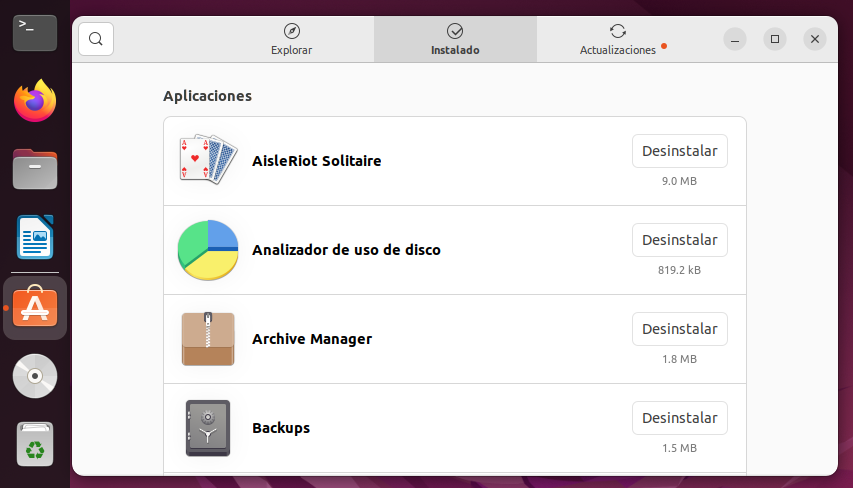


Figura 7. Interfaz herramienta “Ubuntu Software” en Ubuntu. Fuente: elaboración propia.

Podemos encontrar estos programas gestores de software desde el menú principal de cada distribución y en ambas opciones encontramos una aplicación sencilla pero efectiva, con una interfaz clara dividida en tres pestañas; Explorar, Instalado y actualizaciones.

* En la pestaña explorar podrás navegar por la “tienda” de aplicaciones y consultar algunos datos sobre ellas como por ejemplo el tamaño que ocupan.
* Desde la pestaña [Instalado] podremos gestionar el software que ya tenemos instalado pudiendo eliminar los paquetes con un sencillo botón
* Por último, en la pestaña [Actualizaciones] podemos consultar las actualizaciones disponibles para los programas que se encuentran actualmente instalados.

4.3. Acceso a directorios y gestión de repositorios.

Dado que debemos navegar por la estructura de directorios de Linux para gestionar los repositorios del sistema, conviene hacer un pequeño recordatorio de algunos conceptos básicos.

1. El comando [ cd /ruta\_destino ] seguido de la ruta de destino se utiliza para movernos por la estructura de directorios de Linux.
2. Las rutas pueden ser de dos tipos, rutas relativas al punto donde nos encontramos situados y rutas absolutas que parten siempre desde la carpeta raíz [ / ].
3. El comando [ cd ] sin ruta nos ubica en nuestro directorio de trabajo personal.
4. El comando [ cd .. ] nos ubica un directorio por encima del actual.
5. El comando [ cd - ] nos devuelve a la última posición del árbol visitada.
6. Por norma general un usuario estándar en Linux no tiene permisos para crear ficheros dentro de cualquier directorio del árbol que no sea el suyo propio.
7. Para manipular ficheros sobre los cuales no tenemos permiso de escritura deberemos utilizar la sentencia “sudo”.
8. Podemos editar un fichero de texto con herramientas como Nano o Vi, en este caso utilizaremos el comando [ sudo nano /etc/apt/sources.list ]
9. La estructura de directorios de Linux con sus enlaces por defecto es la siguiente:

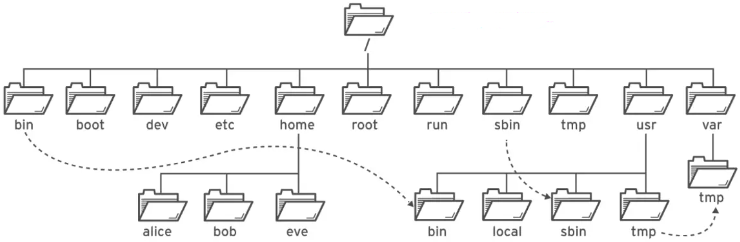


Figura 8. Estructura de directorios en Linux. Fuente: computingforgeeks.com

* **Gestión de repositorios**

Cada repositorio destinado a la administración de software en Linux cuenta con una url o dirección publica en internet a la que nuestro sistema operativo accede cuando realizamos las operaciones estudiadas en el punto anterior.

Existen dos tipos de repositorios, los repositorios oficiales que son proporcionados por cada distribución de Linux y los repositorios PPA (Personal Package Archives) que son repositorios no oficiales creados y mantenidos por la comunidad.

Para que nuestro sistema reconozca estas direcciones a través de los comandos de gestión de software, existe un fichero de configuración ubicado dentro en nuestra máquina de forma local en la ruta /etc/apt/sources.list. Recordemos que el directorio /etc guarda todos los ficheros de configuración del sistema.

Las distribuciones de Linux tienen por defecto configurados unos determinados repositorios pero no todos los repositorios albergan todos los paquetes que podemos necesitar en un momento determinado, por eso es importante comprender la gestión de repositorios mediante la edición del fichero sources.list

Veamos en primer lugar como se estructura el contenido del fichero accediendo a él mediante el comando [ sudo nano /etc/apt/sources.list ]:

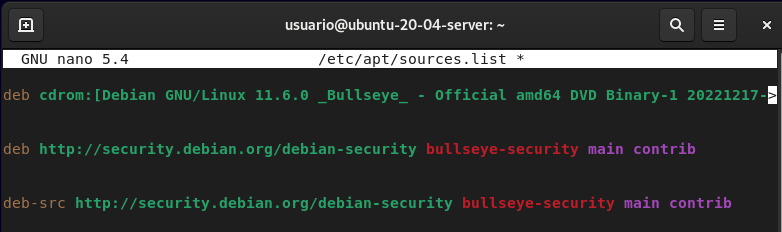


Figura 9. Estructura del fichero de configuración sources.list. Fuente: elaboración propia.

Por defecto, observamos tres líneas en el fichero sources.list de un Debian 11 recién instalado, a continuación se van a desgranar cada una de ellas:

* Primera línea
* 

La primera de las líneas hace referencia a la fuente de instalación primaria, el CD-ROM de instalación del sistema operativo. Esta fuente es utilizada durante la primera configuración del sistema para obtener los paquetes del software que trae instalado por defecto.

En ocasiones el sistema puede encontrar un conflicto cuando intentamos ejecutar el comando [ sudo apt update ] por primera vez en nuestra máquina virtual ya que como primera opción intentará consultar la unidad de disco óptica buscando el repositorio del CD-ROM y si no lo encuentra, se emitirá un fallo por consola impidiendo que se siga consultando el resto de repositorios del fichero sources.list.

En la siguiente imagen puedes observar dicho error:

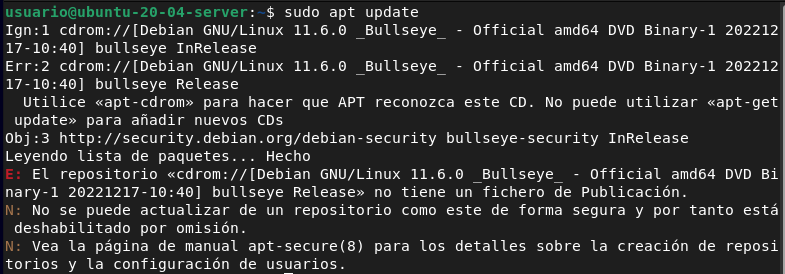


Figura 10. Resultado de la ejecución del comando sudo apt update. Fuente: elaboración propia.

Puedes evitar este conflicto y solucionar el error simplemente comentando esta línea mediante la adición del símbolo “ # ” al principio de la misma

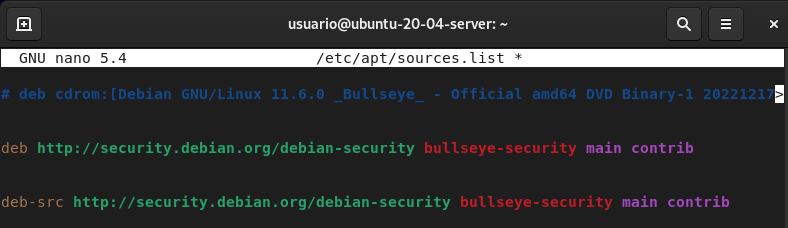


Figura 11. Estructura del fichero de configuración sources.list. Fuente: elaboración propia.

Comentando cualquier línea de un fichero de configuración, conseguimos que el sistema simplemente la ignore, en este caso cuando ejecutemos el comando [ apt ] y se acceda al fichero sources.list esta línea no será interpretada.

* Segunda línea



Esta segunda línea hace referencia a un repositorio real y se divide en varias partes que se pueden diferenciar en diferentes colores:

* deb: Esta primera palabra le indica al sistema que, en ese repositorio debe buscar los paquetes con extensión .deb, comúnmente conocidos como instaladores o binarios.
* <http://security.debian.org/debian-security> : En esta segunda parte de la línea se detalla la dirección del servidor o repositorio en cuestión
* Bullseye-security: Sirve para especificar la versión actual de la distribución de Linux que estemos utilizando
* Main y contrib: Hacen referencia a las secciones dentro del servidor donde el sistema puede descargarse paquetes.

Las secciones se suelen organizar en función del soporte que le ofrece de forma oficial la distribución utilizada y también en función de su licencia de uso y distribución.

* Tercera línea



En la tercera línea encontramos exactamente los mismos datos que en la segunda excepto por la palabra inicial que detallamos a continuación:

* deb-src: Esta primera palabra le indica al sistema que, en ese repositorio debe buscar los paquetes deb-src, comúnmente conocidos como ficheros fuente o código fuente, recordemos que una de las características más importantes del sistema operativo Linux es En los sistemas operativos es que se trata de software libre y por tanto podemos revisar su código fuente para ver cómo están construidos.

En el caso de tratarse de un usuario al que no le interese revisar el código fuente de los programas, podríamos comentar también esta línea para que no se descarguen los ficheros del código fuente y ahorrar así ancho de banda durante la descarga y memoria en el sistema.

Esta es la estructura del fichero sources.list pero ¿Qué sucede si necesitamos descargar un paquete que no se encuentra alojado en ninguno de estos repositorios?

En ese caso tenemos varias opciones.

El primer paso es buscar si la aplicación tiene página oficial con versión de Linux para descargar, en caso negativo deberemos buscar cual es el nombre concreto del paquete para descargar esta aplicación y en que repositorio podemos encontrarlo, (si existe).

Una de las ventajas de Linux es que goza de un gran soporte por parte de la comunidad online y en la mayoría de los casos se encuentra la información que buscamos sin mayores complicaciones.

Cabe destacar que para instalar un paquete que hemos descargado en un fichero .deb, debemos ejecutar el comando [ sudo apt install ] y a continuación especificar la ruta donde se encuentra el fichero.

* **Agregar un repositorio al fichero de configuración.**

Una vez que sabemos cual es la dirección del repositorio que queremos añadir tenemos dos formas de hacerlo:

1. Añadimos de forma manual la línea al fichero con la estructura conocida
2. Utilizamos el comando de sistema destinado a añadir una línea al fichero

En caso de elegir la opción dos los pasos a seguir son los siguientes:

1. Ejecutamos el comando indicado para añadir un repositorio al fichero sources.list es: **sudo add-apt-repository “repositorio\_a\_añadir”**
2. Ejecutamos el comando necesario para actualizar la lista de paquetes disponibles: **sudo apt update**

4.4. Referencias Bibliográficas.

Podemos encontrar más información en la documentación oficial de la distribución que estemos utilizando, a continuación algunos ejemplos:

<https://help.ubuntu.com/stable/ubuntu-help/addremove-sources.html.es>

<https://www.debian.org/doc/index.es.html>

Los comandos de ayuda siempre son una buena opción para aprender más:

apt - -help o man apt

A fondo

Configurar repositorios en Debian.

<https://www.youtube.com/watch?v=RDXgsvCdhlg>

Este video nos proporciona información acerca del fichero sources.list y de los repositorios que en el podemos guardar.

Instalación de software en Linux: Paquetes y Repositorios.

<https://www.youtube.com/watch?v=NTpSA9k5fV4&t=263s>

Este recurso en forma de video de YouTube nos muestra la estructura del fichero sources.list y los comandos básicos de gestión de software mediante consola de comandos.

Repositorios de Linux | Qué son, y cómo instalar y actualizar aplicaciones.

<https://www.youtube.com/watch?v=NTpSA9k5fV4&t=263s>

Este enlace conduce a un video de YouTube donde se nos muestra como configurar un nuevo repositorio para distribuciones Ubuntu

Entrenamientos

Entrenamiento 1

* Enunciado

Agrega la dirección del siguiente repositorio al fichero de configuración mediante el comando diseñado para este propósito.

<http://deb-multimedia.org>

A continuación explica una forma alternativa de añadir el repositorio al sistema.

* Solución

El comando para agregar el repositorio del ejercicio al fichero de configuración es el siguiente:

* **Sudo add-apt-repository** [**http://deb-multimedia.org**](http://deb-multimedia.org)

Una vez añadido al fichero debemos sincronizarlo y actualizar los paquetes disponibles a su última versión.

Podemos agregar manualmente la siguiente línea al fichero de configuración:

**deb** [**http://deb-multimedia.org**](http://deb-multimedia.org) **bullseye main**

La palabra “deb” hace referencia a los ficheros con extensión .deb o binarios que se deben buscar en el repositorio en cuestión.

La palabra “bullseye” corresponde con la versión de la distribución Debian que usamos como ejemplo pero esta debe ser sustituida por la versión que corresponda en cada caso.

Por último, la palabra main referencia la sección concreta dentro del repositorio donde el sistema irá a buscar los paquetes.

Entrenamiento 2

* Enunciado

Sincroniza tu sistema con los repositorios disponibles y actualiza todos los paquetes instalados a su última versión, después responde a las siguientes cuestiones:

¿Cuántos paquetes se han actualizado?

¿Cuántos nuevos paquetes se han instalado?

* Solución

Primero debemos actualizar la lista de paquetes disponibles dentro de los repositorios que tenemos configurados, esta tarea la llevamos a cabo mediante la ejecución del siguiente comando:

* **sudo apt update**

A continuación procedemos con la descarga de los nuevos paquetes disponibles si los hubiese así como la actualización a las ultimas versiones de los paquetes que ya tenemos instalados. Esta tarea la llevamos a cabo mediante la ejecución del siguiente comando:

* **sudo apt upgrade**

Una vez finalizada la actualización observamos los datos mostrados por consola para responder a las cuestiones planteadas, veamos un ejemplo:

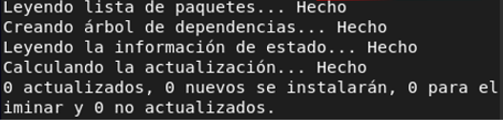


Figura 12. Parte de la ejecución del comando sudo apt upgrade. Fuente: elaboración propia.

Entrenamiento 3

* Enunciado

Descarga e instala el reproductor multimedia VLC desde la consola haciendo uso del comando indicado y responde a las siguientes cuestiones:

¿Cuántos nuevos paquetes se instalarán?

¿Cuántos megabytes es necesario descargar y cuanto espacio adicional se utilizará en disco?

* Solución

El comando que debemos ejecutar para instalar el paquete VLC es el siguiente:

* **sudo apt install vlc**

Una vez que finaliza el proceso de instalación, extracción y configuración observamos los datos mostrados por consola para responder a las cuestiones planteadas, veamos un ejemplo:

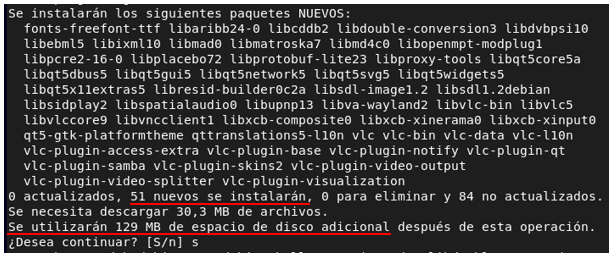


Figura 13. Parte de la ejecución del comando sudo apt install. Fuente: elaboración propia.

Entrenamiento 4

* Enunciado

Elimina del sistema el reproductor multimedia VLC incluyendo el paquete de software y su fichero de configuración. A continuación elimina los paquetes que se instalaron de forma automática y ya no son necesarios.

¿Cuántos espacio se libera tras eliminar el paquete y los ficheros de configuración?

¿Cuántos megabytes se liberan después de eliminar las dependencias?

* Solución

Eliminamos el paquete VLC y sus ficheros de configuración con el siguiente comando:

**sudo apt purge vlc**

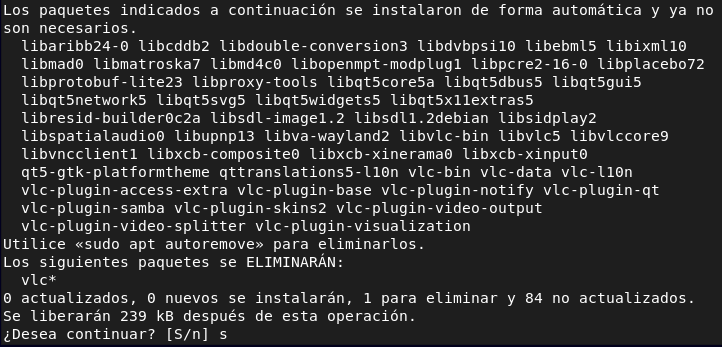


Figura 14. Parte de la ejecución del comando sudo apt purge Fuente: elaboración propia.

Eliminamos las dependencias en desuso con el siguiente comando:

* **sudo apt autoremove**

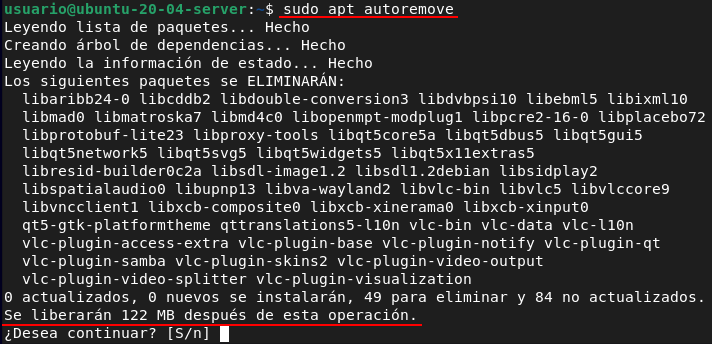


Figura 15. Parte de la ejecución del comando sudo apt autoremove. Fuente: elaboración propia.

Entrenamiento 5

* Enunciado

¿Cuál es el nombre del fichero de configuración de repositorios en Linux?

¿En qué ruta dentro del árbol de directorios podemos encontrarlo?

Ejecuta el comando necesario para editar el fichero de configuración de repositorios y explica como conseguimos que el sistema ignore uno de los repositorios del fichero sin necesidad de eliminarlo.

* Solución

El nombre del fichero de configuración de repositorios es sources.list y podemos encontrarlos en la ruta local /etc/apt/sources.list

Existen varios editores de texto que podemos utilizar para modificar el contenido del fichero de configuración como Nano, Vi o Visual Studio Code. Para el ejemplo vamos a utilizar nano ejecutando el siguiente comando:

* **sudo nano /etc/apt/sources.list**

Una vez entramos al fichero de configuración encontramos varias líneas escritas que serán leídas por el sistema en orden descendente cuando ejecutemos un comando que lo requiera.

Si comentamos una de las líneas añadiendo el símbolo ( # ) al inicio conseguiremos que el sistema operativo las ignore, como si no existieran, cuando necesite recorrer el fichero de nuevo.

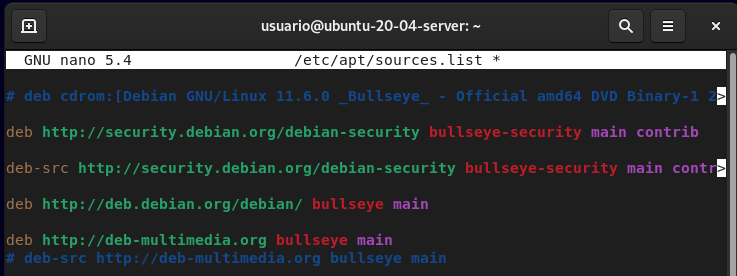


Figura 16. Estructura del fichero de configuración sources.list. Fuente: elaboración propia.

Test

**1.** ¿Qué es un paquete de software en Linux?

A. Un conjunto de archivos necesario para arrancar el sistema.

B. Un fichero de configuración de red.

\_C. Un conjunto de archivos necesarios para instalar y arrancar programas.

D. Un comando especialmente comprimido.

Los paquetes son los conjuntos de archivos que conforman el software en Linux

**2.** ¿Cuáles de los siguientes comandos son utilizados en la gestión de software?

A. tr y cp

B. find y grep.

C. cut y mkdir.

\_D. apt y apt-get.

apt es el comando utilizado en distribuciones basadas en Debian como Ubuntu.

**3.** ¿Qué es un repositorio en Linux? (referido a la administración de software)

A. Un espacio reservado en memoria para actualizaciones

\_B. Servidores en internet que albergan los paquetes de software

C. Una partición de disco destinada a backup

D. Una de los directorios que alberga ficheros de sistema

Los repositorios son los servidores desde los cuales descargamos los paquetes.

**4.** ¿Qué tipos de repositorios de software existen?

A. Primarios y secundarios

B. Repositorios de paquetes y multimedia

\_C. Oficiales y no oficiales (ppa)

D. Repositorios WAN y repositorios MAN

Los repositorios oficiales son ofrecidos y mantenidos por las propias distribuciones mientras que los no oficiales o ppa son mantenidos por la comunidad.

**5.** ¿Qué entendemos por gestión de paquetes en Linux?

\_A. Descarga, instalación, actualización y eliminación de los paquetes.

B. Compartición, modificación y eliminación de los paquetes.

C. Análisis, cifrado y descifrado de los paquetes..

D. Edición y explotación de los paquetes.

La gestión de paquetes nos permite descargar, instalar, actualizar y eliminar paquetes.

**6.** ¿Cuál de las siguientes opciones es correcta?

\_A. apt update actualiza la lista de paquetes disponibles en los repositorios.

B. apt update desinstala los paquetes menos utilizados.

C. apt update actualiza a la última versión los paquetes instalados.

D. apt update instala el primer paquete de cada repositorio.

El comando apt update sincroniza el equipo con el repositorio para actualizar la lista de paquetes disponibles.

**7.** ¿Cómo se denomina el fichero de configuración de repositorios por defecto?

A. repositories.txt

\_B. sources.list

C. sources.xml

D. rep.sys

El fichero de configuración de repositorios por defecto en Linux es sources.list

**8.** ¿Qué significa la palabra deb-src dentro de una línea del fichero source.list?

A. hace referencia a los ficheros binarios del repositorio.

B. hace referencia a la url del repositorio.

C. hace referencia a la distribución del sistema.

\_D. hace referencia a los ficheros de código fuente del repositorio.

La palabra deb-src delante de una línea del fichero de configuración de repositorios nos permite descargar el código fuente del software libre que especifiquemos.

**9.** ¿Con que símbolo podemos comentar una línea en el fichero de configuración?

A. $

\_B. #

C. %

D. !

Utilizamos el símbolo ( # ) al principio de la línea para comentarla.

**10.** ¿Qué función cumple el comando apt autoremove?

\_A. Elimina las dependencias en desuso dentro del sistema.

B. Elimina solo los ficheros de configuración del paquete indicado.

C. Elimina solo el paquete de software indicado.

D. Elimina los ficheros de configuración en desuso dentro del sistema.

Usamos el comando apt autoremove para eliminar los paquetes huérfanos o dependencias sin utilizar que se encuentran ocupando espacio en el disco.

**11** ¿Qué función cumple el comando apt install nombre\_paquete?

1. Busca en el fichero de configuración e instala el repositorio del paquete.

B. hace referencia al paquete más descargado del repositorio.

C. hace referencia a la versión del sistema que estamos utilizando.

\_D. Busca en los repositorios e instala el paquete indicado.

apt install es el comando utilizado para instalar el paquete que le indicamos como parámetro.

**12.** ¿Cuál es la palabra clave usada para ejecutar un comando con permisos de super usuario?

A. superuser

\_B. sudo

C. supu

D. chmod

La palabra clave sudo nos permite ejecutar comandos con permisos de super usuario después de solicitarnos la contraseña pertinente y solo durante unos minutos.

**13.** ¿En qué ruta se encuentra el fichero de configuración de repositorios?

A. /bin/bash/sources.list

\_B. /etc/apt/sources.list

C. /home/user/sources.list

D. /dev/null/sources.list

La ruta por defecto donde el sistema almacena el fichero de configuración para la gestión de repositorios es /etc/apt/sources.list.

**14.** ¿Qué función cumple el comando apt remove nombre\_paquete?

\_A. Elimina solo el paquete indicado.

B. Elimina todos los ficheros relacionados con el paquete indicado.

C. Elimina el paquete indicado y sus dependencias.

D. Elimina solo los ficheros de configuración del paquete indicado.

El comando utilizado cuando queremos desinstalar un paquete de software es apt remove nombre\_paquete, este comando conserva tanto sus dependencias como sus ficheros de configuración.

**15.** ¿Qué significa la palabra deb dentro de una línea del fichero source.list?

\_A. hace referencia a los ficheros instaladores o binarios del repositorio.

B. diferencia un repositorio oficial de uno no oficial.

C. distingue un repositorio actualizado de otro que no lo está.

D. hace referencia a los ficheros ocultos dentro de los repositorios.

La palabra deb indica que en ese repositorio se deben buscar los paquetes con extensión .deb, comúnmente conocidos como instaladores o binarios.

**16.** ¿Qué nombre recibe la herramienta para la gestión de paquetes con interfaz gráfica de las distribuciones Debian?

A. “Software Store”

B. “Debian Software”

\_C. “Software”

D. “Debian Software Store”

La herramienta de gestión de paquetes con interfaz gráfica de las distribuciones Debian se denomina “Software”

**17.** ¿Cómo omitimos la petición de confirmación al instalar o desinstalar paquetes?

A. indicando la palabra omit al final del comando

B. Añadiendo la opción ( -o ) al comando en cuestión

C. Desactivando esta opción en el menú de configuración

\_D. Añadiendo la opción ( -y ) al comando en cuestión

La opción ( -y ) omitirá el mensaje de petición de confirmación cuando instalamos o eliminamos un paquete después de informar de los cambios en el disco

**18.** ¿Cuál es la función del comando apt purge nombre\_paquete?

\_A. Elimina el paquete indicado y sus ficheros de configuración.

B. Depura los bugs o errores del paquete indicado.

C. Elimina por complete todos los ficheros relacionados con el paquete.

D. Descarga la última actualización disponible para el paquete indicado.

Utilizamos el comando purge cuando queremos eliminar un paquete de software y sus ficheros de configuración.

**19.** ¿Cuál es la ruta donde se almacenan los binarios de los paquetes descargados en Debian/Ubuntu?

A. /etc/apt

\_B. /usr/bin o /usr/local/bin

C. /etc/tmp/apt

D. /var/archives

La ruta /usr/bin contiene los ficheros ejecutables de programas instalados para todos los usuarios mientras que /usr/local/bin solo los ejecutables específicos del usuario

**20.** ¿Cuál es la función del comando apt list?

A. Devuelve una lista con los repositorios configurados en sources.list.

B. Lista las dependencias cuyo paquete principal ya no está instalado.

C. Devuelve una lista con los paquetes que están desactualizados.

\_D. Lista los paquetes disponibles en el sistema por nombre.

El comando apt list imprime por pantalla un listado con el nombre versión y arquitectura de los paquetes instados en el sistema ordenados alfabéticamente