



Fuente: Bing Chat AI

MOBAXTERM: MANUAL DE REFERENCIA RÁPIDA

DEPARTAMENTO DE INFORMÁTICA. UNIVERSIDAD DE OVIEDO

Seguridad de Sistemas Informáticos | 2023 – 2024 (v4.0 "S-81 Isaac Peral")



Contenido

Introducción y puesta en marcha	2
Descripción general de la herramienta	2
Descarga.....	2
Uso de la herramienta	3
Preparación y conexión al servidor.....	3
Ejecución de aplicaciones gráficas en el servidor sin arrancar un GUI	5

Introducción y puesta en marcha

Descripción general de la herramienta

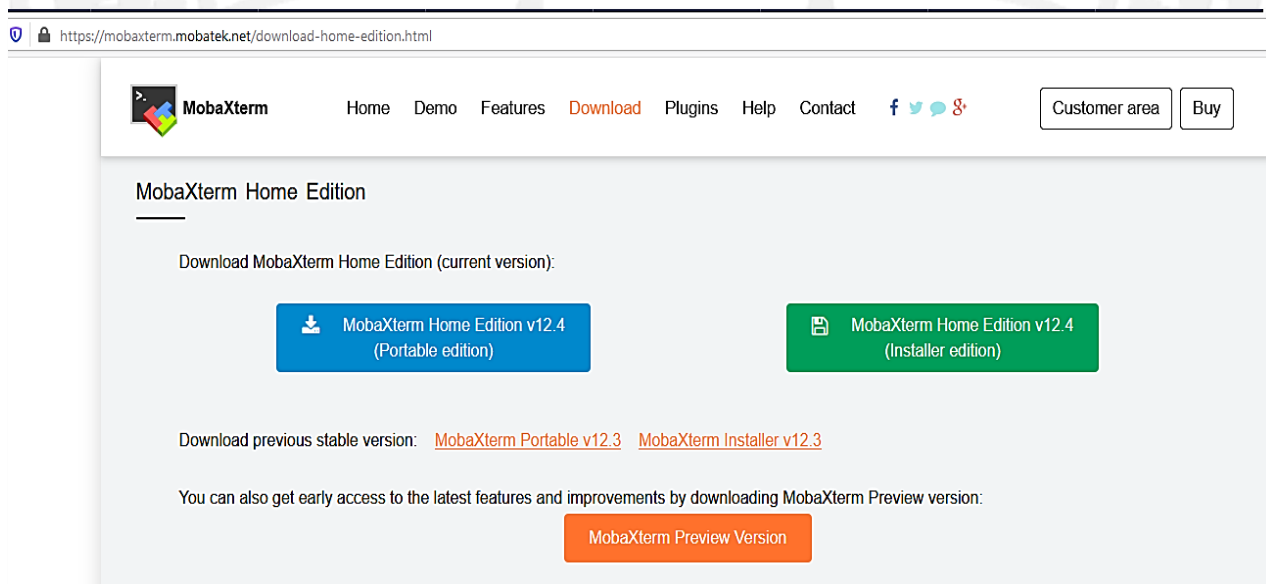
MobaXTerm (<https://mobaxterm.mobatek.net/>) es un cliente que soporta muchos protocolos de acceso remoto a máquinas (SSH, X11, RDP, VNC, FTP, MOSH, ...) y comandos de Unix (`bash`, `ls`, `cat`, `sed`, `grep`, `awk`, `rsync`, ...) en sistemas Windows. Entre sus capacidades más interesantes están:

- **Exportar automáticamente el subsistema gráfico** de un Linux remoto al Windows cliente, de forma que cualquier aplicación gráfica que se arranque en el servidor se vea en el cliente, aunque el servidor esté en modo consola.
- Cliente SSH, admitiendo **varios tabs** e implementando capacidades completas de **copiar y pegar textos**.

Se recomienda usar MobaXTerm como cliente SSH alternativo al conocido Putty o bien para arrancar aplicaciones gráficas en el servidor cuyo interfaz se vea en el cliente sin tener que cambiar ninguna configuración ni hacer operaciones para ello, lo que resulta extremadamente sencillo.

Descarga

MobaXTerm tiene una versión gratuita que podemos descargar de aquí: <https://mobaxterm.mobatek.net/download.html>, y que es suficiente para nuestras prácticas. Es importante elegir la opción portable, ya que no requiere instalación y se puede usar en el laboratorio sin problemas:

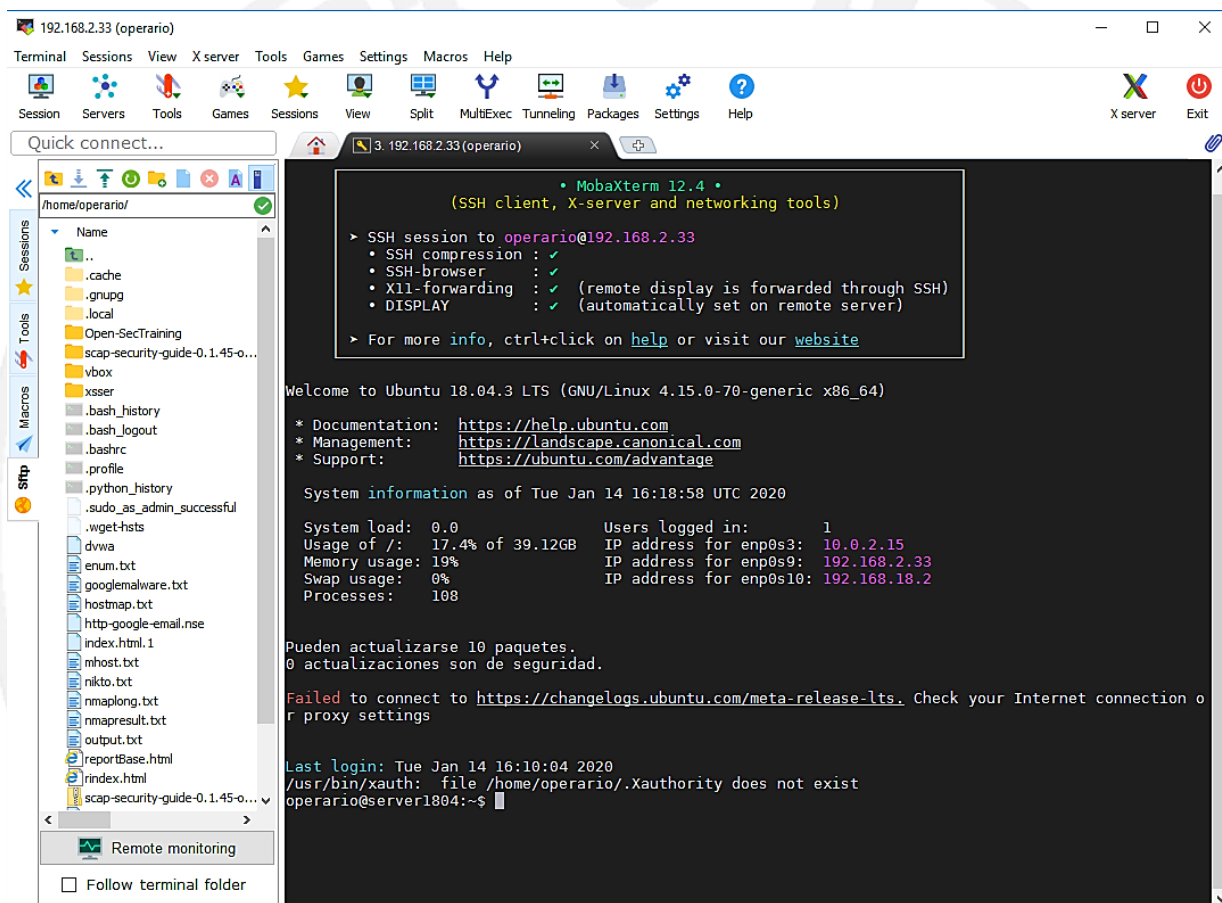


Es compatible con cualquier versión de Windows.

Uso de la herramienta

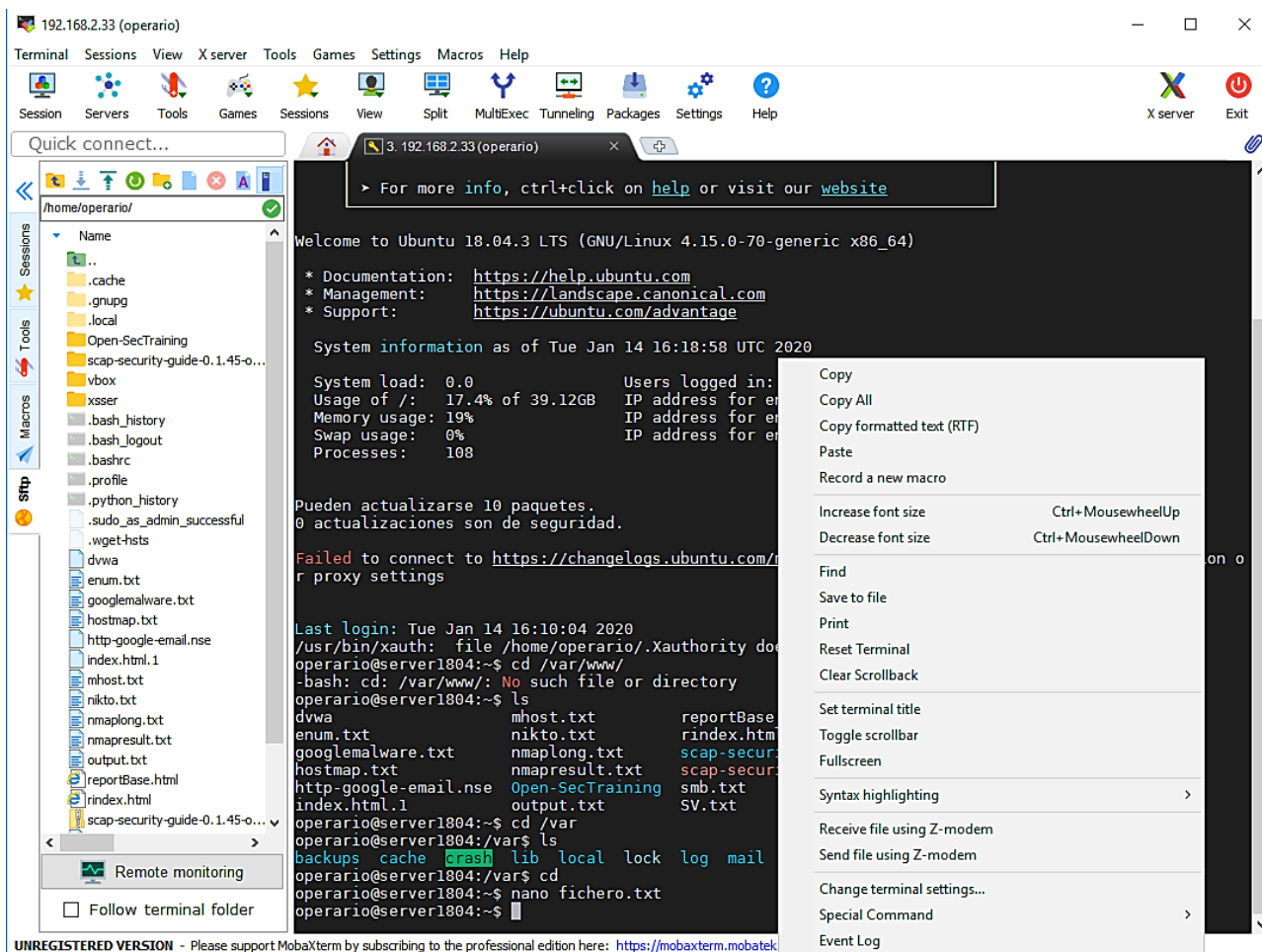
Preparación y conexión al servidor

Para conectarse en modo consola a cualquier servidor remoto es necesario tener instalado el paquete *OpenSSH Server* (si no lo tenemos ya) en dicho servidor con `sudo apt install openssh-server`. En caso de que lo tengamos veremos una consola como esta. Además de la terminal, tenemos acceso a un navegador por los archivos remotos que tengamos permisos de lectura.

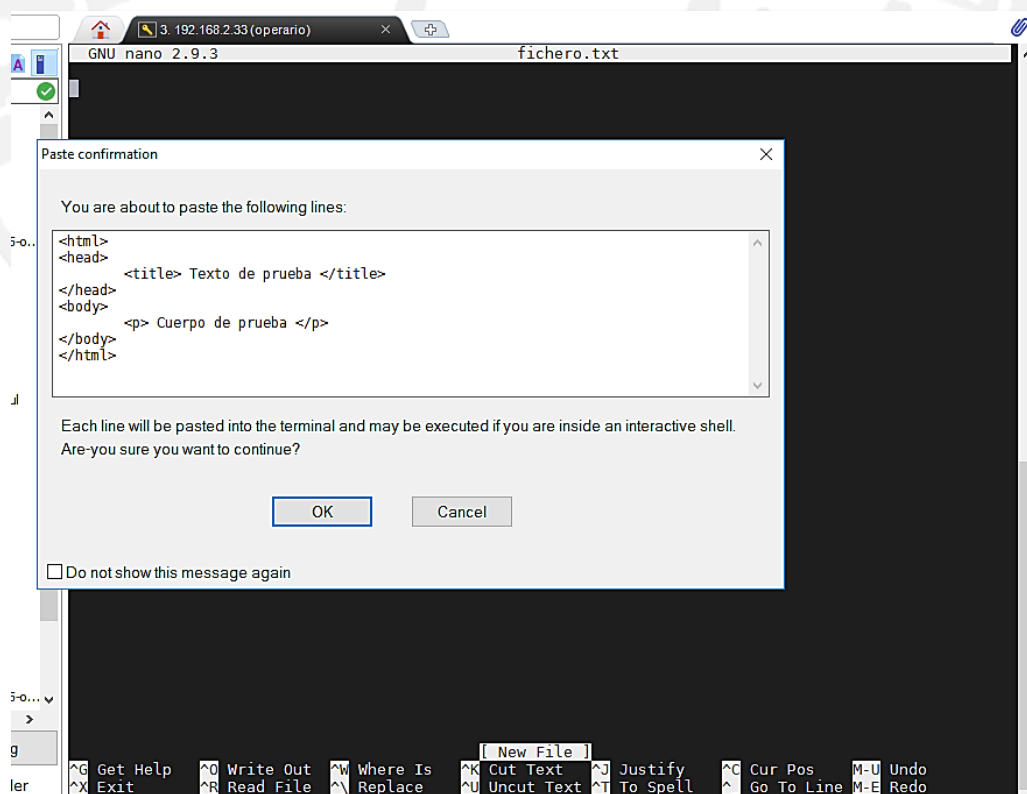


Si hacemos clic derecho en la ventana de la terminal se nos preguntará si queremos asignar este botón al comando **Pegar** o al menú contextual. Asignamos a este último para ver las siguientes opciones (entre las que se encuentra **Pegar** también). La opción de **Pegar** nos permite **copiar cualquier texto de la máquina cliente en el servidor sin problemas**, lo cual es especialmente útil en prácticas que requieran copiar muchos comandos. Además, lo hace sin necesidad de instalar un GUI, por lo que no supone ningún uso de recursos adicional en el servidor. Nótese que esta funcionalidad **también la soporta Putty**.

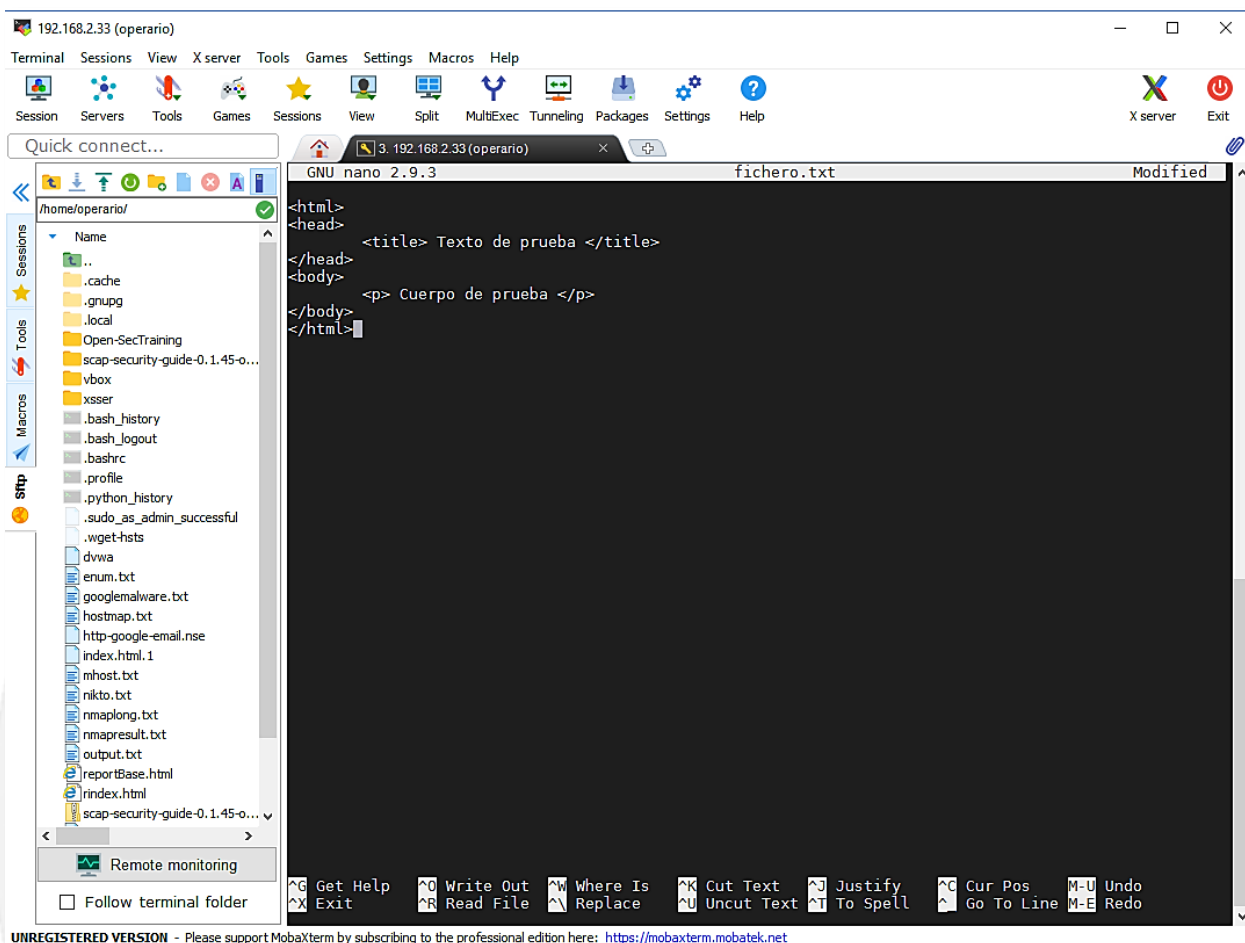
La consola de *MobaXTerm* admite muchos más comandos que nos pueden ser útiles (ver imagen) y que os animamos a probar 😊. Fíjate que también se incluye la opción de **Copiar** el contenido de la ventana.



Una cosa que debemos tener en cuenta es que antes de pegar textos se nos ofrecerá una **preview** de lo que queremos pegar por si acaso. Podemos desactivar esta **preview** si no queremos usarla.



Y finalmente la terminal copiará el texto sobre el programa que tengamos activo: ya sea un editor como el **nano**, o la propia *shell* del SO. Con ello, crear documentos, modificarlos o ejecutar órdenes a partir de los guiones de prácticas en máquinas sin GUI es muy sencillo. Si además lo combinas con soportar varias terminales al mismo tiempo gracias a **tmux**, tendrás un entorno de trabajo basado en texto muy avanzado.



Ejecución de aplicaciones gráficas en el servidor sin arrancar un GUI

Como dijimos anteriormente, una de las características más interesantes de *MobaXTerm* es la de poder **usar aplicaciones gráficas instaladas en el servidor** desde el cliente *Windows* que se conecta al servidor, pero sin necesidad de que el servidor tenga instalado o arranque un GUI completo. Eso es conveniente porque permite usar programas con GUI en un *Ubuntu* que en principio no tiene GUI instalado, **solo la base del subsistema X11**, que ocupa menos que un GUI (unos 500 Mb): <https://developer.ibm.com/tutorials/l-lpic1-106-1/>

Para ello instalamos los siguientes paquetes que nos dan la base del sistema X11: **`sudo apt install xorg openbox`**. Hecho esto podemos instalar cualquier aplicación gráfica, como por ejemplo *Firefox* con **`sudo apt install firefox`**. Por favor, ten en cuenta que la MV que os damos prefabricada ya tiene un GUI instalado, por lo que este paso no es necesario. Gracias a esto no obstante podrás usar aplicaciones con GUI sin falta de arrancarlo (**startx**), lo que ahorra recursos igualmente. Esto es una opción viable si has usado la opción de crear tu propia MV para la asignatura desde cero y no has instalado el GUI todavía: ahora puedes elegir instalar solo la base del sistema X11.

Si arrancamos *Firefox* en el servidor desde *MobaXTerm*, su interfaz se mostrará automáticamente en nuestro *Windows* cliente sin necesidad de instalar todo un GUI. Por tanto, con *MobaXTerm* cubrimos todos los casos de uso que podemos necesitar en la práctica:

- Copia de bloques de texto
- Pegado de bloques de texto
- Prueba de aplicaciones con GUI instalando el mínimo software posible en la máquina *Ubuntu* de la asignatura.

