
MANUAL DE PREPARACION DEL SISTEMA UBUNTU SERVER

Administración y Optimización de Servidores Ubuntu

Versión del sistema: Ubuntu Server 24.04 LTS o superior.

Proyecto autodidacta de aprendizaje técnico en sistemas GNU/Linux

Fecha de edición: 2025

Enfoque: Optimización, instalación de servicios esenciales y buenas prácticas en servidores Ubuntu

Autor: L. Antonio Baeza

Alias técnico: <B4eza>

Estudiante de Ingeniería en Sistemas Computacionales – Tecnológico Nacional de México (TecNM)

Redes profesionales y contacto:

GitHub: <https://github.com/Anton-Bazh>

LinkedIn: www.linkedin.com/in/antoniobaezat

Instagram: <https://www.instagram.com/baeza.doc/>

Prólogo

Este manual ha sido elaborado como una guía técnica personal para la instalación, configuración y administración de servidores Ubuntu, enfocándose en prácticas comunes y útiles en entornos de trabajo reales. Su propósito es reunir de forma clara y estructurada una serie de comandos, recomendaciones, herramientas y enfoques técnicos que pueden facilitar la administración de sistemas GNU/Linux, especialmente para quienes desean consolidar su experiencia práctica.

Es importante subrayar que **este documento no representa una norma oficial ni una guía universalmente aplicable**. Las recomendaciones aquí contenidas son el resultado de observación, práctica y estudio individual, y si bien pueden ser útiles en múltiples escenarios, **deben ser aplicadas con criterio técnico y responsabilidad**, considerando siempre las particularidades del entorno y las necesidades del proyecto en curso.

Del mismo modo, debido a la naturaleza dinámica de los sistemas GNU/Linux y del software en general, se recomienda validar cada comando o herramienta con la documentación oficial más reciente. Algunas utilidades pueden haber cambiado su comportamiento, parámetros o estado de mantenimiento. **La investigación constante y el aprendizaje activo son claves para un ejercicio profesional seguro y competente.**

Aviso de responsabilidad

Este manual es de **uso estrictamente personal y educativo**. No se ofrece ninguna garantía de funcionamiento correcto, ni se asume responsabilidad alguna por daños, pérdidas de información, fallas de servicio o consecuencias derivadas del uso inapropiado o malinterpretación de la información contenida en este documento.

Toda acción tomada como resultado del contenido de este manual es bajo responsabilidad del usuario.

Este documento fue creado como **proyecto personal** por interés propio y como parte de un esfuerzo autodidacta para reforzar habilidades prácticas en administración de servidores Linux. Forma parte de mi proceso de aprendizaje y desarrollo profesional como estudiante de **Ingeniería en Sistemas Computacionales** en el **Tecnológico Nacional de México (TecNM)**. No fue desarrollado por, ni para, dicha institución. Su contenido refleja únicamente el criterio y experiencia del autor.

Si este material resulta útil para otros, es bienvenido, pero **su propósito principal es apoyar mi propio crecimiento técnico y profesional** dentro del campo de la administración de sistemas.

Autor:

<B4eza>— L. Antonio Baeza

Estudiante de Ingeniería en Sistemas Computacionales

Contacto y redes profesionales:

- GitHub: <https://github.com/Anton-Bazh>
- LinkedIn: www.linkedin.com/in/antoniobaezat
- Instagram: <https://www.instagram.com/baeza.doc/>

Contenido

Prólogo	2
Aviso de responsabilidad	2
1. Optimización y preparación inicial del sistema Ubuntu Server.....	6
Introducción	6
1.1 Acceso remoto mediante SSH con MobaXterm	6
1.2 Actualización del sistema y verificación de paquetes	7
1.2.1 Actualizar índices de paquetes.....	7
1.2.2 Actualizar todos los paquetes disponibles	7
1.3 Limpieza del sistema.....	8
1.3.1 Eliminar paquetes obsoletos	8
1.3.2 Limpiar archivos temporales de paquetes descargados.....	8
1.4 Reparación del sistema	8
1.5 Detección de paquetes huérfanos	9
1.6 Verificación de servicios.....	9
1.7 Revisión del sistema de archivos	9
Resultado esperado	10
2: Instalación de Servicios Esenciales	10
1. Instalación de herramientas básicas de red y diagnóstico	10
2. Instalación del servidor OpenSSH.....	10
3. Instalación de herramientas de supervisión del sistema	11
4. Instalación de compresores y gestores de archivos.....	11
5. Habilitar actualizaciones automáticas de seguridad.....	12
3: Herramientas Esenciales para el Usuario y el Entorno de Trabajo	12
1. Editores de texto en consola.....	13
2. Herramientas para compresión y empaquetado (revisión)	13
3. Herramientas de automatización y scripting	13
4. Herramientas de gestión de usuarios y archivos	14
5. Herramientas de conexión remota y portabilidad	14
6. Utilidades adicionales útiles para el usuario	15

1. Optimización y preparación inicial del sistema Ubuntu Server

Autor: <B4eza>

Introducción

Una instalación reciente del sistema operativo Ubuntu Server no siempre está completamente optimizada o actualizada. Por ello, es fundamental realizar una serie de acciones iniciales que aseguren estabilidad, rendimiento, seguridad y una base limpia para futuras configuraciones.

Esta sección describe el procedimiento técnico recomendado para preparar y optimizar un servidor Ubuntu 24.04 LTS, inmediatamente después de su instalación, haciendo uso de una terminal remota a través de una conexión SSH.

1.1 Acceso remoto mediante SSH con MobaXterm

Recomendación profesional:

Se recomienda utilizar un cliente SSH robusto y con soporte visual como [MobaXterm](#) en sistemas Windows. Este cliente facilita la administración remota, al integrar funcionalidades como:

- Soporte SSH directo y comprimido.
- Explorador de archivos remoto (SCP).
- Encaminamiento X11 para interfaces gráficas remotas.
- Terminal Bash integrada con múltiples herramientas Unix.

Procedimiento:

1. Instala MobaXterm en tu sistema cliente (Windows).
2. Obtén la dirección IP del servidor usando el siguiente comando en el mismo:
3. `ip a`

Localiza la IP asociada a la interfaz de red conectada (por ejemplo, `enp6s0`).

4. Inicia una nueva sesión SSH en MobaXterm:
 - Tipo de sesión: SSH.
 - Dirección IP: la obtenida previamente.
 - Usuario: el nombre de usuario creado durante la instalación de Ubuntu Server (por ejemplo, `void`).
5. Conéctate y autentícate.

1.2 Actualización del sistema y verificación de paquetes

1.2.1 Actualizar índices de paquetes

Comando:

```
sudo apt update
```

Descripción técnica:

Este comando sincroniza el índice local de paquetes con los repositorios configurados. No instala ni actualiza software, pero prepara al sistema para saber qué versiones de paquetes están disponibles.

Importante:

Si el sistema se conecta por primera vez, es altamente recomendable realizar esta actualización para evitar la instalación de paquetes obsoletos.

1.2.2 Actualizar todos los paquetes disponibles

Comando:

```
sudo apt full-upgrade
```

Descripción técnica:

Este comando realiza una actualización completa del sistema. A diferencia de `apt upgrade`, que evita eliminar paquetes existentes, `full-upgrade` permite resolver conflictos de dependencias removiendo o reemplazando paquetes si es necesario.

Recomendación:

Usar siempre `full-upgrade` en entornos controlados o inmediatamente después de la instalación base, ya que garantiza coherencia en todas las dependencias.

1.3 Limpieza del sistema

Para mantener un sistema óptimo, se deben eliminar paquetes, archivos temporales o bibliotecas que ya no se usan.

1.3.1 Eliminar paquetes obsoletos

Comando:

```
sudo apt autoremove --purge
```

Este comando elimina automáticamente paquetes instalados como dependencias que ya no son requeridos, usando `--purge` para también eliminar sus archivos de configuración.

1.3.2 Limpiar archivos temporales de paquetes descargados

```
sudo apt clean  
sudo apt autoclean
```

- `apt clean`: elimina todo el contenido del directorio de paquetes (`/var/cache/apt/archives`).
 - `apt autoclean`: elimina solo los paquetes `.deb` que ya no pueden descargarse (versiones antiguas).
-

1.4 Reparación del sistema

Para asegurar que el sistema no tiene configuraciones incompletas o rotas:

```
sudo dpkg --configure -a  
sudo apt install -f
```

- `dpkg --configure -a`: reconfigura cualquier paquete parcialmente instalado.
 - `apt install -f`: corrige dependencias rotas, forzando la instalación de los paquetes necesarios.
-

1.5 Detección de paquetes huérfanos

Instala y usa `deborphan` para detectar bibliotecas instaladas que ya no son requeridas por ningún paquete:

```
sudo apt install deborphan
deborphan
```

Revisa con atención antes de eliminar cualquier biblioteca, ya que algunas pueden ser utilizadas por aplicaciones fuera del sistema de paquetes.

1.6 Verificación de servicios

Para detectar servicios del sistema que hayan fallado:

```
systemctl --failed
```

Un sistema correctamente configurado y funcional no debería mostrar unidades fallidas en esta consulta.

1.7 Revisión del sistema de archivos

Para ejecutar una verificación de integridad del sistema de archivos en el próximo reinicio:

```
sudo touch /forcefsck
sudo reboot
```

El archivo `/forcefsck` instruye al sistema a realizar una comprobación de todos los sistemas de archivos durante el próximo arranque. Es útil para detectar sectores dañados, inodos corruptos o estructuras inconsistentes.

Resultado esperado

Al completar esta fase inicial, el sistema se encontrará:

- Totalmente actualizado.
 - Libre de paquetes obsoletos o no utilizados.
 - Reparado en cuanto a dependencias o configuraciones pendientes.
 - Listo para uso en producción o instalación de servicios adicionales.
-

2: Instalación de Servicios Esenciales

Una vez optimizado el sistema base, el siguiente paso crítico en la administración de un servidor Ubuntu es la instalación y configuración de los servicios esenciales que habilitan su operatividad dentro de una infraestructura.

1. Instalación de herramientas básicas de red y diagnóstico

Estas utilidades permiten la verificación del estado de la red, resolución de dominios, y conectividad entre nodos.

```
sudo apt install -y net-tools iproute2 dnsutils curl wget traceroute  
telnet nmap
```

Descripción técnica:

- `net-tools` y `iproute2`: Herramientas para inspección de interfaces, rutas y configuración IP.
- `dnsutils`: Incluye `dig` y `nslookup`, para pruebas de resolución de nombres DNS.
- `curl`, `wget`: Clientes para transferencias HTTP/HTTPS y otros protocolos.
- `traceroute`, `telnet`: Diagnóstico de conectividad y latencia.
- `nmap`: Escáner de red para descubrimiento de puertos abiertos y servicios.

Recomendación: Verifica la conectividad básica con `ping`, `curl` o `dig` tras la configuración inicial de red.

2. Instalación del servidor OpenSSH

Para permitir la administración remota segura del servidor.

```
sudo apt install -y openssh-server  
sudo systemctl enable ssh  
sudo systemctl start ssh
```

Verificación:

```
systemctl status ssh  
ss -tulpn | grep :22
```

Consideraciones técnicas:

- El puerto 22 debe estar abierto en el firewall si se usa `ufw` o `iptables`.
 - Se recomienda cambiar el puerto por uno no estándar para seguridad adicional y configurar el acceso por clave pública.
-

3. Instalación de herramientas de supervisión del sistema

Estas utilidades son clave para monitorear el rendimiento, detectar cuellos de botella, y realizar tareas de mantenimiento preventivo.

```
sudo apt install -y htop iftop iotop sysstat duf bmon
```

Descripción técnica:

- `htop`: Monitor interactivo de procesos y carga.
 - `iftop`: Visualización en tiempo real del uso de ancho de banda por conexión.
 - `iotop`: Muestra el uso de disco por proceso.
 - `sysstat`: Incluye `iostat`, `mpstat`, `pidstat` para métricas de CPU, disco y procesos.
 - `duf`: Visualización de uso de disco por sistema de archivos.
 - `bmon`: Monitor de ancho de banda con interfaz TUI.
-

4. Instalación de compresores y gestores de archivos

Para trabajar con archivos comprimidos, logs y backups.

```
sudo apt install -y zip unzip p7zip-full tar gzip bzip2 xz-utils rsync
```

Utilidad técnica:

- Soporte para los formatos `.zip`, `.tar.gz`, `.xz`, `.7z`, etc.
 - `rsync`: Herramienta versátil de copia incremental eficiente, ideal para respaldos remotos o sincronización local.
-

5. Habilitar actualizaciones automáticas de seguridad

Aunque ya se realizó la optimización, es buena práctica asegurar la continuidad mediante actualizaciones de seguridad automatizadas.

```
sudo apt install -y unattended-upgrades  
sudo dpkg-reconfigure --priority=low unattended-upgrades
```

Explicación:

Esto garantiza que vulnerabilidades críticas sean parchadas automáticamente, reduciendo la superficie de ataque en servidores productivos.

Nota profesional: Todos estos servicios forman la base operativa mínima sobre la cual se pueden desplegar funciones más complejas como servidores web, bases de datos, contenedores o entornos virtualizados. Documentar cada instalación y configurar los servicios con archivos de control bajo `/etc/` es esencial para mantener trazabilidad y facilidad de administración.

Excelente enfoque. Tener un entorno de trabajo sólido desde el inicio reduce errores operativos y mejora la eficiencia. A continuación te presento la siguiente sección de tu manual:

3: Herramientas Esenciales para el Usuario y el Entorno de Trabajo

En esta sección se detallan herramientas de utilidad general que todo administrador debería tener disponibles en el servidor desde el inicio, tanto para edición de archivos, manejo de datos, automatización de tareas, como para mejorar la interacción remota.

Estas herramientas permiten trabajar de manera eficiente, evitando instalaciones improvisadas durante momentos críticos de administración o depuración.

1. Editores de texto en consola

Los editores permiten modificar archivos de configuración o scripts directamente en el servidor.

```
sudo apt install -y nano micro vim
```

Detalles técnicos:

- `nano`: Editor por defecto en muchos sistemas, de uso simple y sintaxis básica.
- `micro`: Editor moderno con resaltado de sintaxis, soporte para mouse y atajos intuitivos (`Ctrl+S`, `Ctrl+Q`). Ideal para usuarios acostumbrados a entornos gráficos.
- `vim`: Potente y ampliamente usado en entornos Unix. Requiere curva de aprendizaje, pero es altamente eficiente una vez dominado.

Recomendación: Configura `EDITOR=micro` o el editor de preferencia como variable de entorno en tu perfil para uso por defecto con `crontab`, `git`, etc.

2. Herramientas para compresión y empaquetado (revisión)

Aunque ya se instalaron previamente, es recomendable verificar la disponibilidad de:

```
zip unzip p7zip-full tar gzip bzip2 xz-utils
```

Uso común: Manejo de backups, logs comprimidos, o instalación de software desde archivos `.tar.gz` o `.zip`.

3. Herramientas de automatización y scripting

Son esenciales para la administración repetitiva o programada.

```
sudo apt install -y cron at jq moreutils
```

Funciones técnicas:

- `cron`: Daemon de tareas programadas por minuto, hora, día, etc. Persistente.
 - `at`: Permite programar tareas puntuales no recurrentes.
 - `jq`: Procesador de datos JSON desde línea de comandos.
 - `moreutils`: Conjunto de utilidades útiles como `sponge` o `ts`.
-

4. Herramientas de gestión de usuarios y archivos

Para trabajar cómodamente con permisos, estructura y administración del sistema de archivos.

```
sudo apt install -y tree mc acl
```

- `tree`: Visualiza la estructura de directorios de forma jerárquica.
 - `mc` (Midnight Commander): Entorno visual tipo Norton Commander. Navegación de archivos, FTP/SFTP, edición, extracción.
 - `acl`: Permite control de accesos más allá de los permisos POSIX estándar (útil en entornos multiusuario).
-

5. Herramientas de conexión remota y portabilidad

Aunque `ssh` ya está instalado como servicio, estas herramientas mejoran la experiencia remota.

```
sudo apt install -y mosh autossh sshfs
```

Características:

- `mosh` (Mobile Shell): Sustituto de SSH con reconexión automática y tolerancia a cambios de red (ideal en conexiones inestables).
- `autossh`: Reinicia sesiones SSH automáticamente si se caen.
- `sshfs`: Permite montar directorios remotos vía SSH como si fueran locales (`FUSE`), útil para edición directa desde tu equipo de escritorio.

Ejemplo de uso de `sshfs`:

```
sshfs usuario@servidor:/ruta/remota /ruta/local/montaje
```

Requiere tener instalado `fuse` y `sshfs` en el cliente.

6. Utilidades adicionales útiles para el usuario

```
sudo apt install -y lsd bat fzf ripgrep neofetch
```

- `lsd`: Sustituto moderno para `ls`, con colores, íconos y mejor legibilidad.
- `bat`: Sustituto mejorado de `cat`, con resaltado de sintaxis y numeración de líneas.
- `fzf`: Buscador incremental interactivo. Útil en flujos con `git`, `history`, etc.
- `ripgrep` (`rg`): Búsqueda de texto ultrarrápida (más veloz que `grep`).
- `neofetch`: Muestra resumen del sistema (útil para documentación o capturas rápidas).

Recomendación profesional: Estas herramientas deben instalarse antes de iniciar cualquier fase avanzada de configuración o servicio. Tenerlas listas minimiza errores, facilita diagnósticos y permite trabajar con fluidez incluso en sesiones largas o situaciones críticas.
