

Actualizado a 2026

Introducción a Linux y el Software Libre

Edición 2026 - Curso Linux Essentials (LPI 010-160)



Lic. Clever Flores | Aula Útil | clever@aulautil.com

Objetivo del curso

Entender Linux como plataforma real de trabajo: escritorio, servidor, nube, seguridad, contenedores e IA.

- Reconocer qué es Linux, GNU/Linux y el software libre.
- Identificar distribuciones vigentes y cuándo usar cada una.
- Aprender fundamentos de terminal, archivos, permisos, paquetes y red.
- Conectar Linux con cloud, DevOps, ciberseguridad e inteligencia artificial.
- Alinear el temario con Linux Essentials 010-160, objetivos versión 1.6.



Qué se actualizó frente a la versión 2020

La presentación original tenía foco en historia, distribuciones, cloud y contenedores. Esta edición agrega el contexto 2026.

Kernel 7.x

rama estable actual según kernel.org

Ubuntu 26.04 LTS

nueva LTS publicada en abril 2026

Debian 13

Trixie como estable, soporte extendido

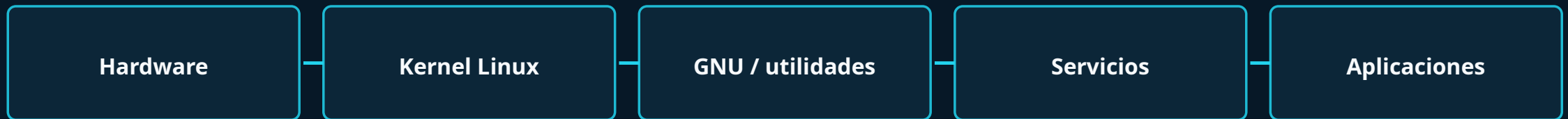
RHEL 10

base empresarial para nube híbrida e IA

- Se reemplazan referencias desfasadas: CentOS 8, Ubuntu 19.10, OpenStack 9, tablas salariales 2019.
- Se incorporan Rocky/AlmaLinux, Podman, Wayland, cloud-init, Kubernetes moderno, observabilidad y Linux para IA.

¿Qué es Linux?

Linux es el kernel: el núcleo que comunica hardware, procesos, memoria, red y almacenamiento.



- Una distribución Linux integra kernel, herramientas, instalador, gestor de paquetes, repositorios y soporte.
- GNU/Linux suele referirse al sistema completo basado en el kernel Linux y herramientas GNU.
- En el trabajo diario decimos “Linux” para referirnos a todo el sistema operativo.

¿Dónde vive Linux hoy?

No solo está en laptops: es la capa invisible de la infraestructura digital.

Nube pública

AWS, Azure, Google Cloud

Nube privada

OpenStack, Proxmox, Kubernetes

Móviles

Android usa kernel Linux

Supercomputación

HPC, simulación, ciencia

Seguridad

firewalls, SIEM, SOC, hardening

IA

GPU servers, MLOps, LLMs



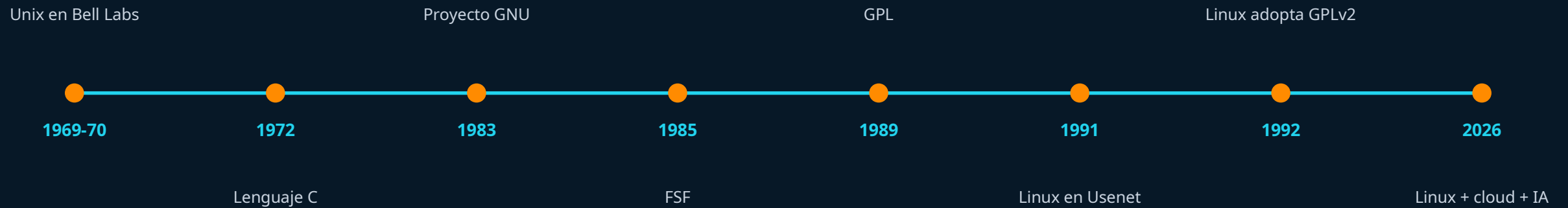
Historia y filosofía

De Unix y GNU al ecosistema abierto que sostiene Internet.



Historia resumida

Linux nace de una cadena de ideas: Unix, C, GNU, GPL y colaboración global.



Software libre: cuatro libertades

El enfoque central no es “gratis”, sino control, aprendizaje, colaboración y redistribución.

0

Usar

ejecutar el programa para cualquier propósito

1

Estudiar

entender y modificar el código fuente

2

Compartir

copiar y distribuir el programa

3

Mejorar

publicar versiones modificadas

Software libre y Open Source

Comparten código abierto, pero priorizan énfasis distintos.

| Enfoque | Software libre | Open Source |
|--------------------|--------------------------------------|---|
| Pregunta central | ¿Qué libertades conserva el usuario? | ¿El código fuente está disponible y colaborativo? |
| Licencias típicas | GPL, AGPL, LGPL | MIT, BSD, Apache, MPL |
| Uso empresarial | Soporte, consultoría, capacitación | Integración, SaaS, nube, comunidad |
| Riesgo a gestionar | Cumplimiento de copyleft | Fragmentación, dependencia de proveedores |

Licencias: copyleft y permisivas

Elegir licencia define cómo se puede usar, modificar, integrar y distribuir el software.

- GPL / AGPL: exige preservar libertades al distribuir derivados.
- LGPL: permite uso en bibliotecas con menos restricciones.
- MIT / BSD / Apache: permisivas; facilitan adopción comercial.
- Apache 2.0: incluye cláusulas de patente, útil en proyectos empresariales.
- Modelo dual: una versión libre y otra comercial con soporte o funciones extra.

Regla práctica: licencia no es detalle legal menor. Afecta arquitectura, negocio, distribución y soporte.

¿Cómo se sostiene el software libre?

La economía open source se mueve por confianza, comunidad y servicios profesionales.



- Suscripciones empresariales: parches, soporte, certificación y ciclo de vida.
- Servicios profesionales: instalación, migración, hardening, monitoreo, automatización.
- Productos cloud: el proveedor opera el software y cobra por disponibilidad, escala y soporte.
- Capacitación y certificaciones: transfieren conocimiento y reducen riesgo operativo.

02

Distribuciones y uso diario

Elegir una distribución depende del caso de uso, soporte y ciclo de vida.

Distribuciones Linux vigentes en 2026

La distribución es la “edición” completa del sistema.

| Familia | Uso típico | Ejemplos 2026 |
|---------|---|---|
| Debian | estabilidad, servidores, base comunitaria | Debian 13, Ubuntu 26.04 LTS, Linux Mint |
| Red Hat | empresa, certificación, soporte extendido | RHEL 10, Rocky 10, AlmaLinux 10, Oracle Linux |
| Fedora | innovación, escritorio, desarrollo | Fedora 44, Fedora Server, Silverblue |
| SUSE | empresa europea, SAP, edge | SLES, openSUSE Leap/Tumbleweed |
| Arch | rolling release, aprendizaje avanzado | Arch, Manjaro, EndeavourOS |

Linux en laptops y PCs

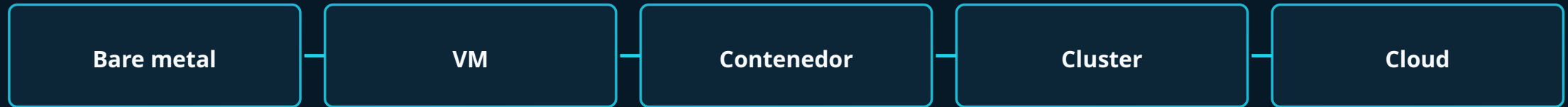
En 2026 el escritorio Linux es más maduro: Wayland, GNOME, KDE Plasma, drivers y apps modernas.



- Ubuntu, Linux Mint, Fedora, Debian, openSUSE, Arch y derivados.
- GNOME y KDE Plasma dominan los entornos modernos; Xfce sigue siendo ideal para equipos modestos.
- Wayland es la ruta principal para gráficos modernos, escalado y seguridad.
- Flatpak, Snap y AppImage simplifican la instalación de aplicaciones.

Linux en servidores

El servidor Linux prioriza estabilidad, seguridad, automatización y soporte.



- Debian/Ubuntu Server: repositorios amplios, comunidad y cloud images.
- RHEL/Rocky/Alma/Oracle: ecosistema empresarial, SELinux y compatibilidad corporativa.
- SLES/openSUSE: entornos enterprise, SAP y escenarios regulados.
- Alpine: imágenes ligeras para contenedores y appliances.

Prioridad 2026

- ciclo de vida LTS
- parches automatizados
- hardening
- observabilidad
- backup probado

CentOS cambió: qué usar ahora

CentOS Linux clásico dejó de ser la base gratuita equivalente a RHEL. En 2026 el mapa es distinto.

| Necesidad | Opción recomendada |
|--------------------------------|--|
| Producción compatible con RHEL | Rocky Linux / AlmaLinux / Oracle Linux |
| Upstream de RHEL | CentOS Stream |
| Soporte comercial | Red Hat Enterprise Linux |
| Migración de CentOS 7/8 | evaluar Rocky/Alma/RHEL, validar repositorios y aplicaciones |
| Desarrollo cloud-native | Fedora, Ubuntu LTS, Debian, contenedores |

Seguridad Linux en 2026

La seguridad ya no es opcional: debe estar integrada en sistema, red, identidad y automatización.



- Actualizaciones frecuentes y kernel LTS cuando se requiera estabilidad.
- SSH con llaves, MFA donde aplique y deshabilitar root remoto.
- SELinux/AppArmor, permisos mínimos y sudo auditado.
- Logs centralizados, monitoreo, EDR/SIEM y alertas.
- Backups probados, snapshots y respuesta ante incidentes.

04

Cloud, virtualización y contenedores

Linux es el lenguaje común de la infraestructura moderna.

Linux y Cloud Computing

Cloud permite consumir infraestructura, plataformas o software bajo demanda.



- IaaS: VMs, redes, volúmenes, balanceadores y firewalls.
- PaaS: bases de datos, colas, runtimes y servicios administrados.
- SaaS: aplicaciones listas para usar como correo, colaboración, CRM o BI.
- Linux está presente en imágenes base, cloud-init, contenedores y appliances de red.

Virtualización sobre Linux

KVM/QEMU convirtió a Linux en una plataforma de virtualización de producción.

| Capa | Tecnologías |
|--------------------------|------------------------------------|
| Hipervisor | KVM, QEMU, Xen, VMware ESXi |
| Gestión | Proxmox VE, oVirt, OpenStack, RHEV |
| Contenedores del sistema | LXC, LXD |
| Almacenamiento | ZFS, Ceph, NFS, iSCSI |
| Red | Linux bridge, Open vSwitch, SDN |

Para el alumno: entender Linux ayuda a administrar Proxmox, cloud privado, DR, backups y automatización de infraestructura.

Proxmox + ZFS + Ceph + PBS

Contenedores: virtualización ligera

Los contenedores empaquetan procesos con dependencias, usando aislamiento del kernel.



- Docker popularizó el flujo de trabajo; Podman es común en entornos empresariales por su integración con systemd y rootless.
- LXC/LXD se usan para contenedores de sistema completos.
- Los contenedores no reemplazan las VMs: se complementan según aislamiento, operación y seguridad.

Kubernetes y OpenShift

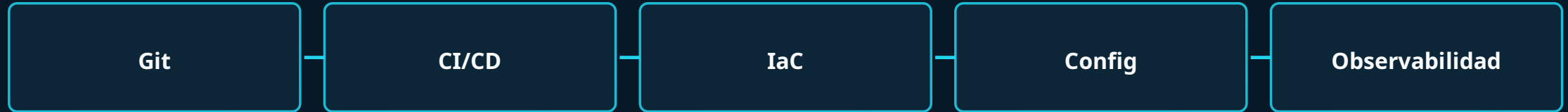
Orquestan contenedores declarando el estado deseado de aplicaciones, red, secretos y almacenamiento.

- Pods, deployments, services, ingress y namespaces son conceptos base.
- Helm y Kustomize ayudan a empaquetar despliegues.
- OpenShift agrega una capa empresarial con seguridad, operadores, CI/CD y consola integrada.
- La operación requiere Linux, red, certificados, almacenamiento y observabilidad.



DevOps e infraestructura como código

Linux deja de administrarse “a mano”: se versiona, prueba y automatiza.



| Necesidad | Herramientas |
|-----------------------------|------------------------------------|
| Provisionar infraestructura | Terraform / OpenTofu |
| Configurar sistemas | Ansible, Salt, Puppet |
| Construir imágenes | Packer, cloud-init, Dockerfile |
| Desplegar aplicaciones | GitHub Actions, GitLab CI, Jenkins |
| Monitorear | Prometheus, Grafana, Zabbix, ELK |

Observabilidad y SOC Open Source

Medir, registrar y alertar es parte de operar Linux profesionalmente.



- Zabbix: monitoreo de infraestructura, hosts, servicios y capacidad.
- Prometheus + Grafana: métricas cloud-native y visualización.
- Wazuh: SIEM/XDR open source, cumplimiento, FIM y respuesta.
- ELK/OpenSearch: logs, búsqueda y análisis.
- BunkerWeb/pfSense/HAProxy: seguridad perimetral y publicación web.

05

Linux con Inteligencia Artificial

La IA moderna se despliega, acelera y opera mayoritariamente sobre Linux.

¿Por qué Linux domina la IA?

Entrenar, servir y automatizar modelos exige control del sistema, drivers, contenedores y red.



- Drivers y runtimes para GPU: NVIDIA CUDA, AMD ROCm y stacks de aceleración.
- Python, compilers, librerías nativas y paquetes científicos.
- Contenedores reproducibles para entrenamiento e inferencia.
- Kubernetes, Slurm o colas de trabajo para clusters.
- Seguridad y observabilidad para modelos en producción.

Stack Linux para IA en 2026

De la GPU al modelo, cada capa requiere administración Linux.



| Capa | Ejemplos |
|---------------|---|
| Frameworks | PyTorch, TensorFlow, JAX, ONNX Runtime |
| Model serving | vLLM, TGI, Ollama, Triton, FastAPI |
| Datos | PostgreSQL, S3/MinIO, vector DB, pipelines |
| MLOps | MLflow, DVC, Kubeflow, Airflow |
| Operación | Prometheus, Grafana, logs, alertas, backups |

Casos prácticos de Linux + IA

El valor no está solo en el modelo: está en integrarlo y operarlo.

LLM local

modelos privados para documentos internos

Voz IP + IA

transcripción, agentes, clasificación de llamadas

Ciberseguridad

detección, correlación y respuesta asistida

Automatización

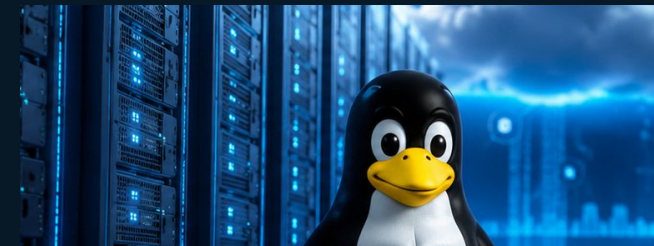
scripts, agentes y runbooks para operaciones

MLOps

versionado, despliegue, monitoreo y retraining

Edge AI

inferencia en cámaras, IoT y gateways



Ruta de aprendizaje

De usuario competente a administrador, cloud engineer o especialista en IA.

Prácticas mínimas para Linux Essentials

El examen valida fundamentos; el mercado exige práctica real.

| Módulo | Práctica recomendada |
|-----------------------|---|
| Comunidad y licencias | comparar GPL, MIT, Apache y modelos de negocio |
| Comandos | navegar, buscar, comprimir, redireccionar, usar pipes |
| Archivos | permisos, enlaces, usuarios y grupos |
| Sistema | procesos, servicios, logs y paquetes |
| Red | SSH, puertos, curl, DNS básico |
| Scripts | Bash simple con variables, condicionales y cron |

Perfiles profesionales 2026

Linux es base para varias rutas de carrera en TI.

Administrador Linux

servidores, seguridad, backups

Cloud Engineer

IaaS, redes, automatización

DevOps / SRE

CI/CD, Kubernetes, observabilidad

Ciberseguridad

hardening, SIEM, respuesta

IA Infra / MLOps

GPU, modelos, APIs, datos

Open Source Integrator

consultoría, soporte, capacitación

Consejo: aprender Linux con proyectos reales vale más que memorizar comandos aislados.

Buenas prácticas para operar Linux

Lo profesional se nota en continuidad, seguridad y documentación.

- Usar versiones LTS o empresariales para producción.
- Separar ambientes: laboratorio, pruebas, producción.
- Automatizar instalación, hardening, parches y respaldos.
- Monitorear desde el primer día: CPU, RAM, disco, red, servicios, logs.
- Documentar cambios, runbooks y procedimientos de recuperación.
- Probar backups y restauración; backup no probado no existe.
- Aplicar mínimo privilegio y rotación de credenciales.

Conclusión

Linux no es solo un sistema operativo.

Es la plataforma común de la nube, la seguridad, los contenedores, la automatización y la inteligencia artificial.

Preguntas

Lic. Clever Flores
clever@aulautil.com
Aula Útil

Linux Essentials 2026

Fuentes y referencias de actualización

Material actualizado a junio de 2026. Las imágenes nuevas fueron generadas para esta versión.

- Presentación original: Introducción a Linux y el Software Libre, Clever Flores, Lima 2020.
- Linux Kernel Archives: kernel.org - ramas stable y longterm.
- Linux Professional Institute: Linux Essentials 010-160, objetivos v1.6.
- Ubuntu 26.04 LTS Release Notes, Canonical, 23 abril 2026.
- Debian 13 “trixie” release information, Debian Project.
- Red Hat Enterprise Linux 10 release dates y anuncio RHEL 10.
- TOP500, lista junio 2025: sistemas exascale y supercomputación.
- Kubernetes v1.34 release blog, Kubernetes Project.