LEASIDE BUSINESS SCHOOL

DESPLIEGUE Y ADMINISTRACIÓN DE INFRAESTRUCTURA CON DOCKER

PLANIFICACIÓN CURRICULAR Y ESTRUCTURA MODULAR FLEX (12 HORAS CRONOLÓGICAS - 100% ASÍNCRONO)

I. DATOS DEL CURSO LIBRE

CURSO:	**Despliegue y Administración de Infraestructura con Docker**	**CÓDIGO:**	**LBS-FLEX-DCK2026**
MODALIDAD:	100% A distancia (Online)	**DESARROLLO:**	100% Asíncrono (Autogestionado)
DURACIÓN:	12 Horas cronológicas (720 min)	**MÓDULOS:**	4 Módulos de Aprendizaje FLEX
PÚBLICO:	Estudiantes, desarrolladores y profesionales TI	**REQUISITOS:**	Ninguno (Nivel Inicial / Intermedio)

CERTIFICACIÓN: Certificado de Aprobación Oficial / Constancia de Participación expedido por **Leaside Business School** al completar las 12 horas cronológicas y aprobar las evaluaciones.

II. PRESENTACIÓN Y FINALIDAD DEL CURSO

El curso **Despliegue y Administración de Infraestructura con Docker** es un programa de capacitación práctica diseñado para dotar a los participantes de las competencias técnicas y operativas necesarias para aprovisionar servidores Linux seguros, empaquetar aplicaciones en contenedores estandarizados (OCI) y administrar infraestructuras resilientes en producción. A través de la metodología FLEX, el participante aprenderá a asegurar nodos cloud con llaves criptográficas Ed25519 y cortafuegos UFW, construir imágenes ultraligeras mediante *Dockerfiles multi-stage*, orquestar arquitecturas multiservicio aisladas con Docker Compose v2 y desplegar proxies inversos Nginx con certificados SSL/TLS Grado A+ integrados a estrategias de respaldo SRE 3-2-1 automatizadas.

JUSTIFICACIÓN TÉCNICA DEL AJUSTE HORARIO (REESTRUCTURACIÓN FLEX A 12.0 HRS FIJAS): Los tiempos de cada etapa FLEX han sido recalibrados a 150 minutos exactos por módulo (2.5 hrs), logrando una sincronización matemática del 100% con la duración total de 12.0 Horas Cronológicas (720 min).

III. INTRODUCCIÓN Y ORIENTACIÓN AL ESTUDIANTE

- **Video de Bienvenida:** Inducción ejecutiva de 2-3 minutos explicando la dinámica de trabajo, configuración de terminal y metas.
- **Estructura Curricular:** 4 módulos de aprendizaje práctico con secuencia pedagógica estandarizada FLEX.
- **Navegación en Plataforma:** Acceso 100% asíncrono las 24 horas con seguimiento de progreso automático.
- **Metodología FLEX:** Implementación de 8 procesos de aprendizaje guiado por cada módulo (150 min exactos por módulo).

- **Criterio de Aprobación:** Calificación mínima acumulativa de **14 / 20** para la emisión de certificación.
- **Certificado Digital:** Descarga directa de constancia institucional por 12 horas al concluir los 4 módulos.

IV. DISTRIBUCIÓN HORARIA DE CONTENIDOS (12 HORAS CRONOLÓGICAS)

Fase / Módulo del Curso	Horas	Minutos	Enfoque Principal / Objetivo Clave
Bienvenida, Orientación y Evaluación Diagnóstica Inicial	**0.5 Hrs.**	30 min	Inducción al entorno FLEX, configuración de terminal y test inicial
Módulo 1: Fundamentos de Linux, Redes y Aprovisionamiento Seguro	**2.5 Hrs.**	150 min	Arquitectura cloud, SSH criptográfico Ed25519 y cortafuegos UFW
Módulo 2: Motor de Contenedores Docker y Construcción de Imágenes OCI	**2.5 Hrs.**	150 min	Namespaces, Cgroups v2, ciclo de vida y Dockerfile multi-stage
Módulo 3: Orquestación Multiservicio con Docker Compose y Redes Aisladas	**2.5 Hrs.**	150 min	Manifiestos compose.yaml, volúmenes, redes bridge y DOCKER-USER
Módulo 4: Proxy Inverso Nginx, SSL/TLS Grado A+ y Operaciones SRE 3-2-1	**2.5 Hrs.**	150 min	Enrutamiento Ingress, certificados Let's Encrypt, ab y backups 3-2-1
Repaso Integrador y Evaluación Final de Certificación	**1.5 Hrs.**	90 min	Despliegue de proyecto integrador en entorno real y examen final
DURACIÓN TOTAL DEL CURSO FLEX	**12.0 Hrs.**	**720 min**	**Certificación Oficial Leaside Business School**

V. DESARROLLO MODULAR ANALÍTICO (PROCESOS METODOLOGÍA FLEX)

MÓDULO 1: Fundamentos de Linux, Redes y Aprovisionamiento Seguro (2.5 Hrs / 150 min)	
1. INICIA (5 min)	**Objetivo:** Mitigar riesgos perimetrales. **Caso:** 'Un servidor recién desplegado con contraseña simple recibe 1,400 intentos de intrusión en sus primeros 15 minutos. ¿Cómo se mitiga el riesgo mediante llaves criptográficas Ed25519 y cortafuegos perimetral?'
2. APRENDE (45 min)	**Lecciones 1.1 a 1.4:** Arquitectura física de Centros de Datos y virtualización KVM. Direccionamiento IPv4/IPv6 y mapeo de puertos TCP/UDP. Protocolo SSH criptográfico (llaves Ed25519 vs RSA). Jerarquía de archivos FHS 3.0 e inodos. Usuarios sin privilegios y blindaje con UFW.
3. OBSERVA (15 min)	**Screencast Guiado:** Creación de un par de llaves Ed25519 en consola local, transferencia segura hacia el VPS remoto, deshabilitación del acceso a root en `sshd_config` y verificación de conexión en puerto no estándar.
4. EXPLORA (15 min)	**Guía Técnica Descargable:** 'Checklist Maestro de Hardening para Ubuntu 24.04 LTS: Configuración de Sysctl, Fail2ban, límites de descriptores y memoria Swap'.
5. PRACTICA (35 min)	**Simulador Interactivo:** Desafío de comandos Bash para resolver denegaciones de permisos, asignar permisos octales (`chmod 750`), cambiar propietarios (`chown`) e inspeccionar sockets activos con `ss -tulpn`.
6. APLICA (20 min)	**Taller Práctico:** Caso 'Configuración perimetral de un nuevo nodo cloud'. Creación del usuario `deployer`, restricción de sudoers con `visudo` y activación de reglas de bloqueo por fuerza bruta con UFW y Fail2ban.
7. COMPRUEBA (10 min)	**Quiz de Retención (10 preguntas):** Cuestionario sobre puertos seguros, funcionamiento de Diffie-Hellman en SSH, cálculo de inodos y resolución de conflictos de red.
8. RECUERDA (5 min)	**Resumen Visual:** Infografía técnica con el 'Diagrama de Flujo de Conexión Segura SSH': Cliente local → Cifrado asimétrico → Puerto 22 → Autenticación por `authorized_keys` → Shell restringida.

MÓDULO 2: Motor de Contenedores Docker y Construcción de Imágenes OCI (2.5 Hrs / 150 min)	
1. INICIA (5 min)	**Objetivo:** Estandarizar entornos con unidades inmutables. **Caso:** 'En mi máquina sí funcionaba, pero en el servidor falló por discrepancias de librerías. ¿Cómo resuelve Docker este conflicto empaquetando el código junto a sus dependencias exactas?'
2. APRENDE (45 min)	**Lecciones 2.1 a 2.4:** Primitivas del Kernel de Linux (los 8 Namespaces y Cgroups v2). Arquitectura del demonio: `dockerd`, `containerd`, `containerd-shim` y `runc`. Ciclo de vida de contenedores (`create, start, run, exec, logs, inspect`). Construcción de imágenes OCI con Dockerfile multi-stage y usuario no privilegiado.
3. OBSERVA (15 min)	**Video Paso a Paso:** Instalación oficial de Docker CE en Ubuntu 24.04 LTS con repositorio firmado criptográficamente, configuración del grupo docker y compilación de una imagen Node.js optimizada reduciendo el peso de 1 GB a 45 MB.
4. EXPLORA (15 min)	**Documento Técnico:** 'Manual Canónico de Directivas de Dockerfile: FROM, WORKDIR, COPY, RUN, USER, EXPOSE, HEALTHCHECK, ENTRYPOINT y CMD analizados palabra por palabra'.
5. PRACTICA (35 min)	**Laboratorio Interactivo:** Escritura y depuración de un Dockerfile con vulnerabilidades detectadas, corrigiendo la ejecución de root, eliminando archivos innecesarios con `.dockerignore` y fijando variables de entorno.
6. APLICA (20 min)	**Caso Práctico:** 'Empaquetado de una API RESTful'. Construcción de imagen con cacheo eficiente de capas, definición de un healthcheck nativo con `wget` y ejecución en segundo plano con límites estrictos de memoria Cgroups (`--memory=512m`).
7. COMPRUEBA (10 min)	**Quiz de Retención (10 preguntas):** Evaluación sobre capas de solo lectura OverlayFS, diferencias entre CMD y ENTRYPOINT, y propagación de señales SIGTERM/SIGKILL.
8. RECUERDA (5 min)	**Cuadro Comparativo Visual:** Matriz de contraste entre Máquinas Virtuales (Hipervisor KVM con Kernel duplicado) vs. Contenedores Docker (Aislamiento a nivel de Kernel sobre el sistema anfitrión).

MÓDULO 3: Orquestación Multiservicio con Docker Compose y Redes Aisladas (2.5 Hrs / 150 min)	
1. INICIA (5 min)	**Objetivo:** Aislar redes y proteger bases de datos. **Caso:** 'Una base de datos publicada en el puerto 5432 sufre un ataque de fuerza bruta porque Docker omitió el cortafuegos UFW. ¿Cómo se diseña una red bridge aislada y se mitiga el Docker-UFW Bypass?'
2. APRENDE (45 min)	**Lecciones 3.1 a 3.4:** Orquestación declarativa con Docker Compose v2. Manifiestos `compose.yaml`: sintaxis de servicios, volúmenes, redes y variables `.env`. Persistencia de datos (Named Volumes vs Bind Mounts). Redes bridge personalizadas, DNS interno en `127.0.0.11` y cadena `DOCKER-USER` en iptables.
3. OBSERVA (15 min)	**Screencast Técnico:** Despliegue de un stack completo de 3 niveles (Frontend Nginx + Backend Node.js + Base de Datos PostgreSQL) con orden de inicio condicional mediante `depends_on` y verificación de salud de base de datos.
4. EXPLORA (15 min)	**Plantilla Descargable:** 'Manifiesto compose.yaml de Producción Enterprise': Estructura de carpetas en `/opt/production`, archivo `.env` protegido con `chmod 600` y configuración de volúmenes persistentes.
5. PRACTICA (35 min)	**Simulador de Redes:** Interconexión de contenedores mediante CLI, ejecución de pruebas de conectividad con `getent hosts` y validación de sockets con `netcat` comprobando el aislamiento de la base de datos.
6. APLICA (20 min)	**Taller Integrador:** Caso 'Tienda de Comercio Electrónico'. Creación del archivo `compose.yaml` con redes segmentadas (`frontend-net` y `backend-net`), asignación de límites Cgroups y volúmenes montados con permisos UID/GID.
7. COMPRUEBA (10 min)	**Quiz de Retención (10 preguntas):** Evaluación sobre resolución de nombres en redes bridge, sintaxis YAML de Compose v2, persistencia ante `docker compose down` y mitigación del bypass UFW.
8. RECUERDA (5 min)	**Diagrama de Red:** Esquema visual del 'Par Veth y Red Bridge': Conmutador virtual `docker0` en el anfitrión conectado mediante túnel `veth` a la interfaz `eth0` interna del contenedor.

MÓDULO 4: Proxy Inverso Nginx, SSL/TLS Grado A+ y Operaciones SRE con Respaldo 3-2-1 (2.5 Hrs / 150 min)	
1. INICIA (5 min)	**Objetivo:** Alta disponibilidad y continuidad SRE. **Caso:** 'Un certificado SSL vence a medianoche y el disco del servidor se llena al 100% por logs no rotados de Docker, provocando la caída del negocio. ¿Cómo evitan este desastre un proxy Nginx automatizado y una estrategia SRE 3-2-1?'
2. APRENDE (45 min)	**Lecciones 4.1 a 4.4:** Proxy inverso con Nginx y modelo asíncrono `epoll`. Bloques `server`, `proxy_pass` y cabeceras de seguridad (HSTS, CSP, X-Frame-Options). Automatización SSL/TLS con Certbot y Let's Encrypt (HTTP-01). Rotación de logs en `/etc/docker/daemon.json`. Pruebas con ApacheBench y Respaldo 3-2-1 (Zstandard, GnuPG y Rclone).
3. OBSERVA (15 min)	**Video Demostrativo:** 1) Configuración de Nginx como ingress proxy hacia contenedores y obtención de certificado SSL con calificación A+ en SSL Labs. 2) Simulacro de restauración en frío (Cold Restore Drill) levantando la base de datos desde un archivo cifrado.
4. EXPLORA (15 min)	**Script Descargable:** `backup-vps.sh`: Código Bash robusto con `set -euo pipefail`, volcado en caliente de PostgreSQL con `pg_dump`, compresión `zstd`, cifrado AES-256 y sincronización automática a Cloudflare R2 con Rclone.
5. PRACTICA (35 min)	**Laboratorio SRE:** Ejecución de prueba de estrés con `ab -n 10000 -c 100`, monitoreo en vivo con `docker stats` y `vmstat 1 5`, y diagnóstico forense de procesos sacrificados por el OOM Killer mediante `dmesg -T`.
6. APLICA (20 min)	**Proyecto Integrador:** Despliegue de un proxy inverso Nginx securizado con SSL, configuración de alertas de salud en Uptime Kuma y programación de temporizadores *Systemd Timers* para respaldos diarios desatendidos.
7. COMPRUEBA (10 min)	**Quiz Técnico Final (10 preguntas):** Evaluación sobre parámetros de `proxy_set_header`, renovación de certificados con Certbot, cálculo de percentiles de latencia (p95/p99) y protocolo de respaldo 3-2-1.
8. RECUERDA (5 min)	**Checklist Producción:** Matriz de 10 puntos de control SRE (Límites Cgroups, rotación de logs, cortafuegos DOCKER-USER, SSL A+, respaldos externos y monitoreo de salud activo).

VI. SISTEMA DE EVALUACIÓN Y PONDERACIÓN

Instrumento de Evaluación	Indicador / Competencia Evaluada	Ponderación
Evaluaciones Formativas (Quizzes)	Comprensión conceptual de redes TCP/IP, criptografía SSH, primitivas de Kernel (Namespaces/Cgroups), directivas Dockerfile, sintaxis Compose y seguridad SSL/TLS.	**30%**
Actividades Prácticas (FLEX Práctica/Aplica)	Aprovisionamiento de host seguro con UFW/Fail2ban, construcción de imágenes optimizadas multi-stage, orquestación de stack multi-contenedor y pruebas de carga con ApacheBench.	**40%**
Proyecto Integrador Final	Despliegue integral de una infraestructura productiva en VPS: Nginx Ingress con SSL Grado A+, aplicación contenedorizada, base de datos persistente, rotación de logs y script de respaldo 3-2-1 con Rclone.	**30%**
TOTAL NOTA FINAL DEL CURSO	**Escala Vigesimal (0 a 20). Nota Mínima de Aprobación: 14 / 20**	**100%**

VII. CIERRE DEL CURSO Y RECURSOS

Al concluir y aprobar satisfactoriamente los 4 módulos de aprendizaje, la plataforma virtual otorgará automáticamente:

- Resumen ejecutivo de competencias e indicadores alcanzados en administración de infraestructura con Docker.
- Manuales técnicos descargables en PDF, plantillas `compose.yaml` y script automatizado `backup-vps.sh`.
- Encuesta de Satisfacción de Calidad Académica FLEX.
- **Certificado de Aprobación Oficial por 12 Horas** expedido por **Leaside Business School**.

VIII. BIBLIOGRAFÍA Y FUENTES TÉCNICAS

1. Docker Inc. (2026). *Docker Engine, Compose Specification & OCI Image Format Documentation*. Docker Technical Resources.
2. Linux Foundation (2025). *Filesystem Hierarchy Standard (FHS 3.0) & Kernel Namespaces Architecture*.
3. Nginx Inc. (2026). *Nginx Core & Reverse Proxy Architecture Reference Manual*. F5 Documentation.
4. Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site Reliability Engineering: How Google Runs Production Systems*. O'Reilly Media.
5. CISA (2024). *Data Backup Options and the 3-2-1 Strategy*. US Department of Homeland Security.