



UNIVERSIDAD DE SONSONATE

USO

SCIENTIAE ET BONIS ARTIBUS

# IMPLANTACIÓN DE SISTEMAS

# AGENDA

1. Explicación de ejercicios
2. ¿Qué es WSL?
3. Fundamentos del análisis de conectividad
4. Análisis de transferencia de datos
5. Diagnóstico avanzado

# WSL (WINDOWS SUBSYSTEM FOR LINUX)

## CARACTERÍSTICAS

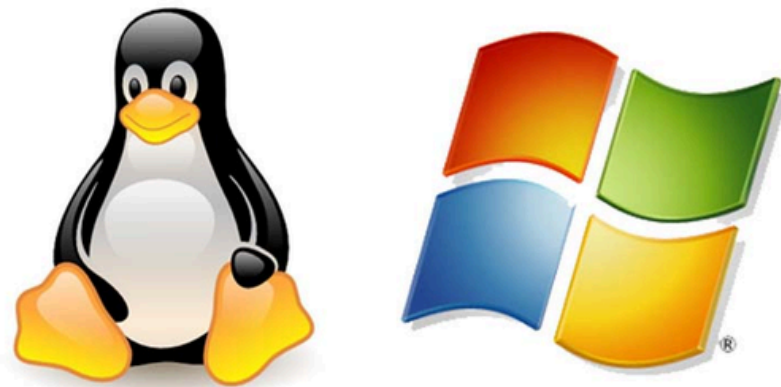
- Linux nativo en Windows
- Uso de comandos de Linux
- Integración con Windows
- Desarrollo multiplataforma

## VERSIONES

- WSL 1
- WSL 2

## BENEFICIOS

- No se necesita maquina virtual.
- Consume menos recursos de hardware.
- Permite facil uso de Linux sin salir de Windows.
- Es gratis.





# ANÁLISIS DE CONECTIVIDAD Y TRANSFERENCIA DE DATOS



# FUNDAMENTOS DEL ANÁLISIS DE CONECTIVIDAD

# METODOLOGÍA DE DIAGNÓSTICO DE RED

## ¿Qué es el análisis de conectividad?

El análisis de conectividad es el proceso de verificar, medir y diagnosticar cómo se comunican dos o más puntos de una red (dispositivos, servidores, servicios en la nube, contenedores, etc.).

Busca responder:

¿Hay alcance? (reachability)

¿Por qué ruta viajan los paquetes?

¿Con qué calidad? (latencia, pérdida, ancho de banda).

Componentes clave de la conectividad:

- Conectividad física: Cables, puertos, interfaces
- Conectividad lógica: Configuración IP, rutas, protocolos
- Conectividad de aplicación: Servicios, puertos, protocolos de capa superior

# INTRODUCCIÓN AL ANÁLISIS DE CONECTIVIDAD

## Enfoque Bottom-Up (De abajo hacia arriba)

1. Capa física (cables, conectores, LEDs)
2. Capa de enlace (ARP, switching)
3. Capa de red (IP, routing)
4. Capa de transporte (TCP/UDP)
5. Capa de aplicación (HTTP, FTP, etc.)

## Enfoque Top-Down (De arriba hacia abajo)

1. Aplicación (¿funciona el servicio?)
2. Transporte (¿están los puertos abiertos?)
3. Red (¿hay conectividad IP?)
4. Enlace (¿está la interfaz activa?)
5. Física (¿hay señal?)

## Cuándo usar cada enfoque:

- Bottom-Up: Problemas intermitentes o de rendimiento
- Top-Down: Problemas evidentes de aplicación

# HERRAMIENTAS FUNDAMENTALES DE DIAGNÓSTICO

## Ping avanzado

ping -c 4 8.8.8.8	# Linux/Mac (4 paquetes)
ping -n 4 8.8.8.8	# Windows (4 paquetes)
ping -s 1400 8.8.8.8	# Paquetes grandes para probar MTU
ping -i 0.2 8.8.8.8	# Intervalo de 0.2 segundos

## Traceroute/Tracert

traceroute google.com	# Linux/Mac
tracert google.com	# Windows
traceroute -n google.com	# Solo IPs, no resolución DNS

## ARP (Address Resolution Protocol)

arp -a	# Ver tabla ARP
arp -d 192.168.1.1	# Eliminar entrada específica
arp -s 192.168.1.1 aa:bb:cc:dd:ee:ff	# Agregar entrada estática





# ANÁLISIS DE TRANSFERENCIA DE DATOS

# CONCEPTOS DE RENDIMIENTO DE RED

## MÉTRICAS FUNDAMENTALES

### ANCHO DE BANDA (BANDWIDTH)

- CAPACIDAD MÁXIMA TEÓRICA DEL ENLACE
- SE MIDE EN BPS, KBPS, MBPS, GBPS
- DIFERENCIA ENTRE ANCHO DE BANDA Y THROUGHPUT

### THROUGHPUT (RENDIMIENTO EFECTIVO)

- VELOCIDAD REAL DE TRANSFERENCIA DE DATOS ÚTILES
- SIEMPRE MENOR QUE EL ANCHO DE BANDA DEBIDO A OVERHEAD
- AFECTADO POR PROTOCOLOS, CONGESTIÓN, ERRORES

### LATENCIA (DELAY)

- TIEMPO QUE TARDA UN PAQUETE EN VIAJAR DEL ORIGEN AL DESTINO
- COMPONENTES: PROPAGACIÓN, TRANSMISIÓN, PROCESAMIENTO, COLA
- SE MIDE EN MILLISEGUNDOS (MS)

### PÉRDIDA DE PAQUETES (PACKET LOSS)

- PORCENTAJE DE PAQUETES QUE NO LLEGAN AL DESTINO
- CAUSAS: CONGESTIÓN, ERRORES, FALLAS DE HARDWARE
- IMPACTO EN TCP VS UDP

# HERRAMIENTAS DE MEDICIÓN DE RENDIMIENTO



## Ping estadístico

```
ping -c 100 8.8.8.8 | tail -1 # Linux: estadísticas finales
ping -n 100 -l 1024 8.8.8.8 # Windows: paquetes de 1024 bytes
```

## Speedtest-CLI

```
speedtest-cli # Prueba básica
speedtest-cli --server 12345 # Servidor específico
speedtest-cli --json # Salida en JSON
```

## hping3 (Ping avanzado)

```
hping3 -S -p 80 google.com # SYN a puerto 80
hping3 -1 -i u100 google.com # ICMP cada 100 microsegundos
```

## iPerf/iPerf3 (Medición de ancho de banda)

```
# Servidor
iperf3 -s
```

```
# Cliente
iperf3 -c servidor_ip -t 30 # Prueba de 30 segundos
iperf3 -c servidor_ip -P 4 # 4 conexiones paralelas
iperf3 -c servidor_ip -u # Prueba UDP
```

# ANÁLISIS DE CONECTIVIDAD POR CAPAS

## Capa Física

# Linux

ethtool eth0

# Estado de interfaz

ip link show

# Interfaces disponibles

dmesg | grep eth

# Mensajes del kernel

# Windows

ipconfig /all

# Configuración completa

## Capa de aplicación

# Telnet para probar puertos

telnet google.com 80

telnet smtp.gmail.com 587

# nmap para escaneo de puertos

nmap -p 80,443 google.com

nmap -sT -p 1-1000 192.168.1.1

## Capa de Red

# Tabla de ruteo

route -n

# Linux

route print

# Windows

ip route show

# Linux moderno

# Configuración IP

ifconfig

# Linux clásico

ip addr show

# Linux moderno

ipconfig

# Windows

# HERRAMIENTAS DE MONITOREO CONTINUO

## MTR (My TraceRoute)

```
mtr google.com          # Traceroute continuo
mtr -r -c 100 google.com # Reporte de 100
paquetes
mtr --no-dns google.com  # Sin resolución DNS
```

## tcpdump (línea de comandos)

```
tcpdump -i any icmp      # Capturar ICMP en
todas las interfaces
tcpdump -i eth0 port 80  # Tráfico HTTP
tcpdump -w captura.pcap  # Guardar captura
```

## Wireshark básico

- Captura de paquetes en tiempo real
- Filtros básicos: icmp, tcp.port == 80, ip.addr == 8.8.8.8
- Análisis de handshake TCP
- Identificación de retransmisiones

The image features a central text block flanked by four decorative geometric patterns. The top-left and bottom-right patterns are composed of dark blue lines, while the top-right and bottom-left patterns are composed of teal lines. Each pattern consists of multiple horizontal and vertical lines of varying lengths, creating a stylized, architectural look.

MUCHAS  
GRACIAS