

¡Perfecto! Aquí tenés el **Cheatsheet Combinado Windows + Linux + WSL**, con el mismo estilo que tus ejemplos. Todo está estructurado para usarlo como referencia rápida y profesional.

💻 Windows + 🛡️ Linux + 🛡️ WSL – Monitoring & Diagnostics Cheatsheet

Tema: Monitoreo, diagnóstico y herramientas esenciales en Windows, Linux y WSL **Objetivo:** Dominio completo del sistema, uso combinado y entender qué funciona en cada ambiente **Incluye:** Tablas comparativas, comandos clave y explicación técnica

[Introducción]

Este cheatsheet combina lo mejor del diagnóstico de:

- **Windows** (Task Manager, PowerShell, Performance Monitor)
- **Linux** (htop, journalctl, systemd, perf, etc.)
- **WSL** (sub-sistema híbrido con limitaciones)

Sirve como referencia completa para técnicos, sysadmins, devs y power users.

✳️ 1. Tabla Comparativa de Herramientas

Área	Windows	Linux	WSL Compatibilidad
CPU/RAM	Task Manager, Resource Monitor	top, htop, glances	✓ Funciona
Disco	PerfMon, Resource Monitor	iostop, df, lsblk	✗ iostop falla (no hardware real)
Red	TCPView, Get-NetTCPConnection	ss, iftop, nethogs	✓ ss / ✗ iftop
Logs	Event Viewer	journalctl	✓ Parcial (si WSL tiene systemd)
Kernel	PerfMon	dmesg, perf	✗ Sin kernel real
Procesos	Process Explorer	ps, pstree	✓ Total
Boot	Reliability Monitor	systemd-analyze	✗ WSL no tiene boot
S.M.A.R.T. disco	CrystalDiskInfo	smartctl	✗ WSL no accede a discos
Debug profundo	ProcMon	strace, perf	✓ strace, ✗ perf

□ 2. Monitoreo y Diagnóstico en Windows

◆ Herramientas principales

Herramienta	Uso
Task Manager	Diagnóstico rápido de procesos
Resource Monitor	CPU, RAM, disco, red por proceso
Performance Monitor (PerfMon)	Métricas avanzadas y logging
Event Viewer	Logs del sistema
Reliability Monitor	Línea de tiempo de fallos
Sysinternals Suite	Kit profesional: ProcMon, Process Explorer, TCPView, Autoruns

◆ Comandos PowerShell útiles

CPU y procesos

```
Get-Process | Sort CPU -Descending | Select -First 10
```

RAM

```
Get-Process | Sort WorkingSet -Descending | Select -First 10
```

Disco

```
Get-Disk  
Get-PhysicalDisk  
Get-Volume
```

Red

```
Get-NetTCPConnection
```

Logs del sistema

```
Get-EventLog -LogName System -Newest 50
```



3. Monitoreo y Diagnóstico en **Linux**

◆ Herramientas principales

Herramienta	Uso
htop/top	Procesos en tiempo real
iostop	Uso de disco por proceso
iftop/nethogs	Tráfico de red
vmstat	Memoria, swap, procesos
journalctl	Logs del sistema
systemctl	Estado de servicios
lsblk/df/du	Información de disco
perf	Profiling del CPU
strace	Syscalls de procesos

◆ Comandos esenciales

Procesos

```
ps aux  
pstree  
kill -9 PID
```

CPU

```
mpstat 1  
top -o %CPU
```

Memoria

```
free -h  
vmstat 1
```

Disco

```
df -h  
du -sh /*  
lsblk  
iostop
```

Red

```
ss -tulpn  
ip a  
ip route
```

Logs

```
journalctl -xe  
journalctl -u servicio  
dmesg -T
```

4. Diagnóstico en WSL (Compatibilidad explicada)

WSL funciona como una capa Linux sobre Windows, pero **NO** accede al kernel real ni al hardware físico.

✓ Herramientas **que sí funcionan** en WSL

Funcionan porque no requieren acceso al kernel o hardware real:

```
top  
htop  
ps  
free  
df  
du  
lsof  
ss  
ip  
pstree  
journalctl      # solo si hay systemd  
strace
```

- ✓ Diagnóstico de procesos ✓ Métricas básicas ✓ Herramientas de red basadas en sockets ✓ Logs del userland
-

✗ Herramientas que **NO funcionan** en WSL

No funcionan porque requieren acceso directo al hardware o al kernel:

```
iotop
iostat
smartctl
hdparm
iftop
nethogs
perf
systemd-analyze
lspci
lsusb
```

⌚ Por qué no funcionan (explicación corta)

Recurso	¿WSL lo ve?	Motivo
CPU	✓	Kernel virtualizado replica métricas
RAM	✓	Es memoria asignada a WSL
Disco real	✗	WSL usa un VHDX virtual
S.M.A.R.T.	✗	No hay acceso a /dev/sda
Kernel logs	✗	dmesg está limitado
Tráfico real	✗	Interfaz virtual, no ve tráfico de Windows
Servicios systemd	✓ / ✗	Depende de la versión (WSL2 puede activarlo)
Boot del sistema	✗	WSL no bootea como Linux real

⌚ 5. En qué es mejor cada entorno

Tarea	Windows	Linux	WSL
Monitoreo de hardware	★★★★★	★★★★★	✗
Procesos y debugging	★★★	★★★★★	★★★
Análisis de red	★★★	★★★★★	★★
Logs del sistema	★★	★★★★★	★

Tarea	Windows	Linux	WSL
Desarrollo	☆☆	☆☆☆☆☆	☆☆☆☆☆
Automatización	☆☆☆	☆☆☆☆☆	☆☆☆
Seguridad	☆☆	☆☆☆☆☆	☆☆

⌚ Conclusión

✓ **Windows**: perfecto para monitorear **hardware real** y aplicaciones ✓ **Linux**: ideal para **servidores, debugging profundo y administración técnica** ✓ **WSL**: excelente para **desarrollo**, diagnósticos rápidos y herramientas básicas ✗ Pero no reemplaza a Linux real en rendimiento ni acceso al kernel

¿Quieres ahora...?

↗ Versión **PDF** del cheatsheet ↗ Versión **DOCX** ↗ Una hoja de ruta para **convertirte en experto en monitoreo** ↗ Un **script para Windows**, otro para **Linux**, o uno **mixto Windows + WSL**

Decime y te lo preparo.