Active Directory Lab Setup

Cross-Platform Development Environment Implementation with Windows Subsystem for Linux (WSL2)

Overview

This document details the step-by-step implementation and optimization of a cross-platform development environment using Windows Subsystem for Linux 2 (WSL2). The project replaced dual-boot setups and streamlined development and security testing workflows by integrating Linux (Ubuntu and Kali) directly into a Windows environment.

Goals

- Eliminate the need for dual-boot configurations
- Create a seamless development environment across Windows and Linux
- Enable GUI and CLI-based Linux tool usage inside Windows
- Integrate advanced security testing without extra hardware
- Document and scale implementation for team-wide adoption

Step-by-Step Implementation

Step 1: Install WSL2 and Linux Distributions

1. **Enable WSL and Virtual Machine Platform**

   dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart

   dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart

2. **Install WSL2 Kernel Update** from Microsoft: https://aka.ms/wsl2kernel

3. **Set WSL2 as the default version**: wsl --set-default-version 2

4. **Install Linux Distributions (Ubuntu, Kali)** from Microsoft Store

---

Step 2: Configure Shared File Access

1. Access the Linux file system via Windows at: \\wsl$\Ubuntu

2. Mount Windows drives in Linux via /mnt/c, /mnt/d, etc.

3. Use a shared folder for seamless access and Git version control.

Step 3: Optimize WSL2 Virtualization Settings

1. Create `.wslconfig` in Windows user folder:

```
[wsl2]
memory=4GB
processors=2
swap=2GB
localhostForwarding=true
```

2. Restart WSL: wsl --shutdown

Step 4: Cross-Platform Command Execution

1. From Linux: /mnt/c/Windows/System32/notepad.exe

2. From Windows (PowerShell): wsl ls -la

3. Use scripts and aliases to bridge toolchains.

Step 5: Install GUI Support (WSLg)

- GUI support is available out-of-the-box in Windows 11

- Example: sudo apt install gedit, then run gedit

Step 6: Security Testing via Kali Linux

1. Launch Kali via WSL terminal

2. Install tools:

   sudo apt update && sudo apt install nmap metasploit-framework john hydra

3. Access and scan local network:

   sudo nmap -sP 192.168.1.0/24

Step 7: Team Documentation & Rollout

1. Created setup and troubleshooting documentation

2. Hosted README and guides on internal wiki

3. Delivered onboarding sessions and video walkthroughs

4. Achieved 100% adoption by team within one month

Project Impact

- Reduced context-switching between OSes by 40%
- Cut setup time for new environments from days to hours
- Enabled integrated Linux/Windows workflows
- Improved security posture with built-in tools
- Reduced hardware costs

Repository Structure Suggestion

/WSL2_CrossPlatform_Environment

```
 setup
    install_wsl2.ps1
    wslconfig_sample.ini
 docs
    WSL2_Setup_Guide.pdf
    screenshot-setup.png
 README.md
 scripts
     cross_platform_tools.sh
```