A COMPLETE VIDEO COURSE

Windows Server 2025 Complete System Administration

From PowerShell and Active Directory to Hotpatching, Hyper-V, Windows Containers, and modern storage — for junior, intermediate, and senior administrators.

- 60 Slides · 7 Modules
- Hands-on PowerShell Examples Throughout
- Windows Server 2025 (LTSC)



```
PS> Get-ComputerInfo | select WindowsProductName

WindowsProductName :

"Windows Server 2025"


PS> _
```

COURSE ROADMAP

What You Will Learn



01 Foundations & Architecture

Isolation models, editions & licensing, Active Directory

02 Package & System Management

PowerShell, winget, Server Core, services, logging

03 Networking

DNS, DHCP, PowerShell networking, Windows Firewall

04 Security & Cryptography

Secured-core, BitLocker, PKI, hotpatching, MFA, LAPS

05 Containers & Virtualization

Hyper-V, GPU-P, Storage Spaces Direct, Windows Containers

06 Development Toolchains

PowerShell 7, .NET, Python, Node.js, Java, C/C++, WSL2

+ BONUS MODULE 07 The real Copilot-on-Windows-Server story, plus free chapters & the complete book

01

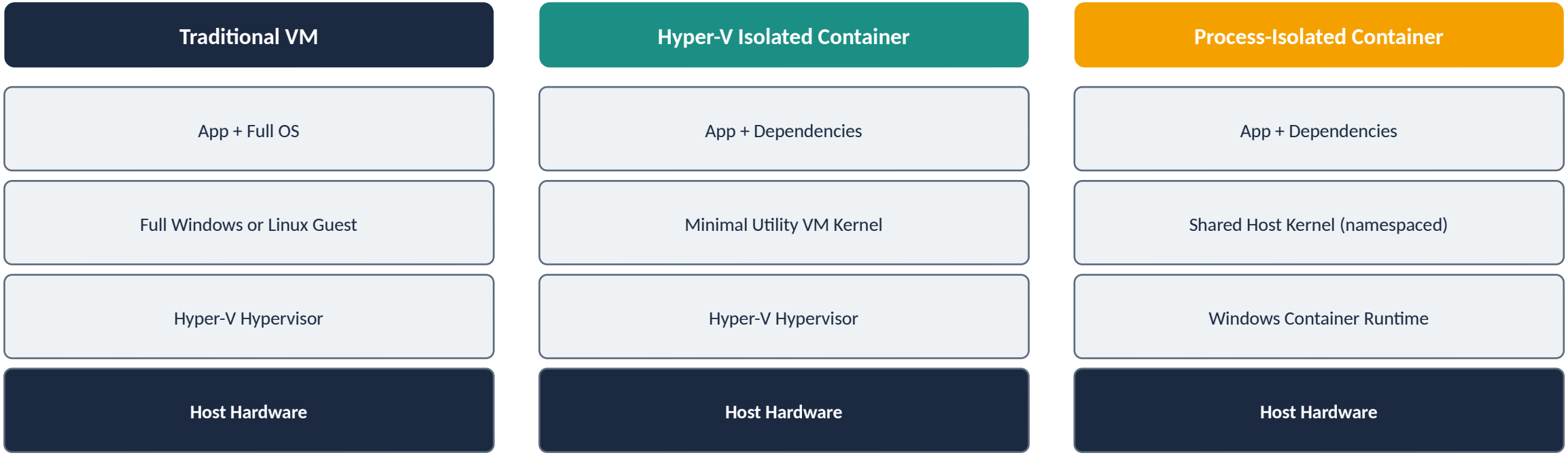
MODULE 1

Foundations & Architecture

Understand the three isolation models unique to Windows Server, how editions and licensing work, and how Active Directory Domain Services is structured.

- Isolation Models
- Editions & Licensing
- Active Directory
- Domain Controllers
- Group Policy

Three Isolation Models on Windows Server



Process isolation shares the kernel (fast, less isolation); Hyper-V isolation gives each container its own lightweight kernel (slower, VM-grade isolation).

Windows Server 2025 Editions

Standard	Datacenter	Datacenter: Azure Edition	Essentials
Physical or minimally virtualized environments. Up to 2 OS environments (VMs) per license.	Highly virtualized environments. Unlimited OS environments, Storage Spaces Direct, Shielded VMs.	Runs as an Azure VM or on Azure Local — adds Hotpatching and innovation delivered continuously.	Entry-level edition for small businesses, capped at 25 users / 50 devices.

Licensed per physical core (minimum 8 cores/16 core-licenses per server) plus Client Access Licenses (CALs) for Standard and Datacenter.

Windows Server 2025 vs. 2022: What Changed

Windows Server 2022

- Secured-core server introduced, TLS 1.3 support added
- 8 KB Active Directory database pages
- Standard NVMe support and S2D
- Extended support until October 2031

Windows Server 2025

- Hotpatching via Azure Arc, Credential Guard on by default
- 32 KB Active Directory database pages — faster on huge object trees
- NVMe-optimized I/O, live ReFS block-level dedup, SMB over QUIC
- Mainstream support to Nov 2029, extended support to Nov 2034

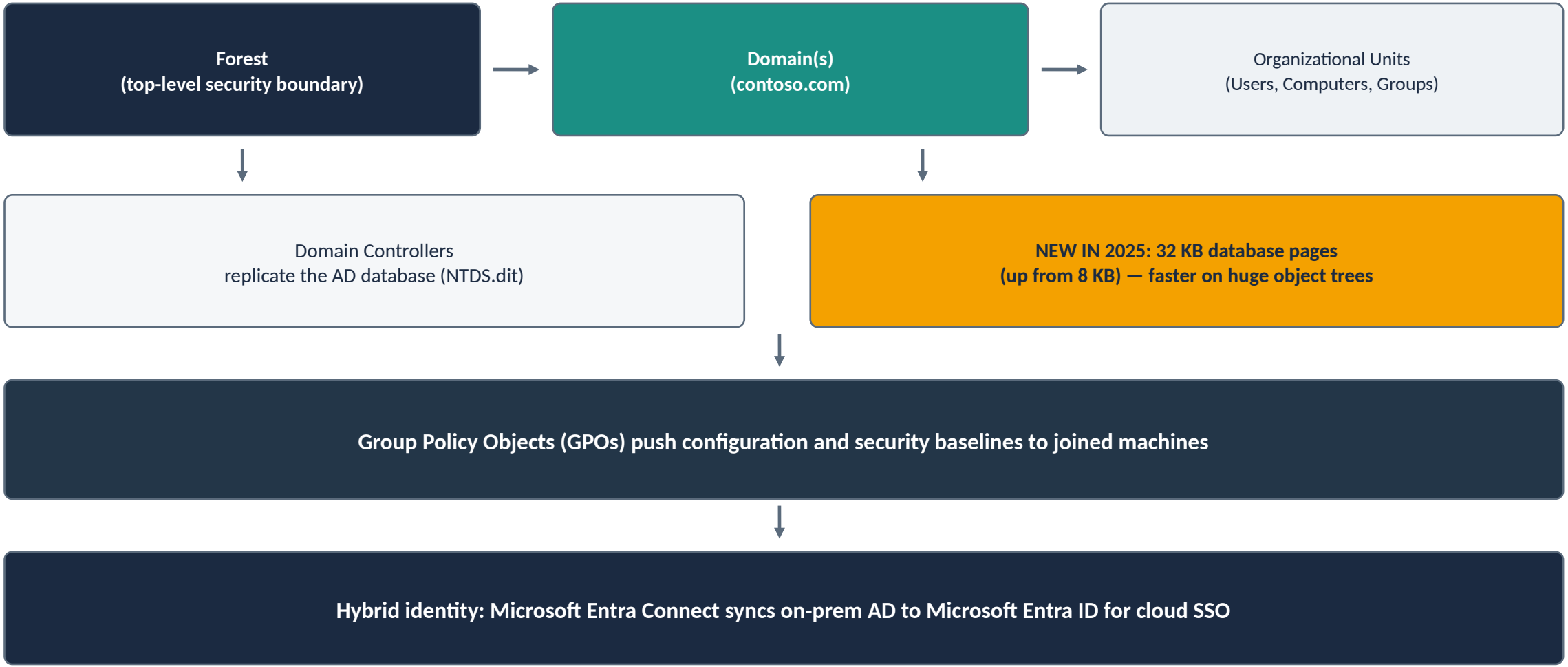
This course targets Windows Server 2025, the current Long-Term Servicing Channel (LTSC) release.

The Windows Server 2025 Platform at a Glance

240 TB	2,048	60%	2034
Max RAM per Gen2 VM	Max virtual processors per VM	Higher NVMe IOPS vs. 2022	Extended support ends

- Hyper-V maximums are theoretical architectural headroom — always validate against your actual hardware
- Azure Arc is installed and connected by default, unifying on-prem and cloud management
- GPU Partitioning (GPU-P) now supports live migration of GPU-accelerated VMs
- Native sudo command brings a Linux-like elevation experience to Windows Server as well

Active Directory Domain Services Architecture



02

MODULE 2

Package & System Management

Master PowerShell as your primary interface, manage software with winget and the PowerShell Gallery, and get comfortable with services, scheduled tasks, and Event Viewer logging.

PowerShell | winget | Server Core | Services | Event Viewer | DSC

PowerShell Fundamentals

- PowerShell is the primary administrative interface for Windows Server — GUI tools call the same cmdlets underneath
- Cmdlets follow a consistent Verb-Noun naming pattern (Get-Service, Set-NetIPAddress)
- Get-Help and Get-Command are your built-in reference for any cmdlet
- Objects flow through the pipeline, not just text — filter and reshape with Where-Object and Select-Object

powershell-basics.ps1

```
PS>
Get-Command -Module Microsoft.PowerShell.Management
PS>
Get-Help Get-Service -Examples
PS>
Get-Service | Where-Object {$_.Status -eq "Running"}
PS>
Get-Process | Sort-Object CPU -Descending | Select-
Object -First 5
PS>
Get-ComputerInfo | Select-Object WindowsProductName,
OsHardwareAbstractionLayer
```

Package Management: winget & PowerShell Gallery

- winget is Microsoft's official CLI package manager, built into current Windows Server images
- The PowerShell Gallery distributes reusable PowerShell modules — install with Install-Module
- Chocolatey remains popular for third-party software not yet in winget's catalog
- Both tools resolve dependencies and support silent, scriptable installs

package-mgmt.ps1

```
PS>
winget search sysinternals
PS>
winget install Microsoft.Sysinternals.ProcessExplorer
PS>
winget upgrade --all
PS>
Install-Module -Name Az -Scope CurrentUser
PS>
Find-Module -Name *ActiveDirectory*
PS>
Get-InstalledModule
```

Server Core vs. Desktop Experience

Server Core

- No Windows Explorer shell — administered via PowerShell or remotely
- Smaller footprint: fewer updates, smaller attack surface, faster patching
- Recommended default for most production roles (AD DS, Hyper-V, file servers)
- Switch to Desktop Experience only requires a fresh install — it's a one-way choice at setup

Desktop Experience

- Full GUI — Windows Explorer, Server Manager, MMC snap-ins available locally
- Easier for admins early in their PowerShell journey
- Larger footprint means more monthly updates and more potential reboots
- Good fit for roles needing local GUI-only management tools

Windows Admin Center manages either installation option remotely through one consistent browser-based UI.

Windows Update, WSUS & Update Rings

- Windows Server Update Services (WSUS) centralizes patch approval and distribution for a fleet
- Update rings stagger rollout — pilot group first, broad deployment after validation
- PSWindowsUpdate module lets you script updates the same way you would any other cmdlet
- Azure Arc-connected servers can instead use Azure Update Manager for a cloud-native view

windows-update.ps1

```
PS>
Install-Module PSWindowsUpdate
PS>
Get-WindowsUpdate
PS>
Install-WindowsUpdate -AcceptAll -AutoReboot
PS>
Get-WUHistory | Select-Object -First 10
# On the WSUS server:
PS>
Get-WsusServer | Get-WsusUpdate | Where-Object
{$_.State -eq "NeedsApproval"}
```

PowerShell Remoting & Desired State Configuration

- PowerShell Remoting (WinRM) lets you run commands on remote servers as if you were local
- Enter-PSSession opens an interactive remote shell; Invoke-Command runs a one-off script block
- Desired State Configuration (DSC) declares the end state a server should be in — DSC then enforces it
- This is Windows' closest parallel to Ansible playbooks or Terraform for OS-level config

remoting-dsc.ps1

```
PS>
Enable-PSRemoting -Force
PS>
Enter-PSSession -ComputerName SRV01
PS>
Invoke-Command -ComputerName SRV01,SRV02 -ScriptBlock
{ Get-Service W32Time }
Configuration WebServerConfig {
  Node "SRV01" { WindowsFeature IIS { Name = "Web-
Server"; Ensure = "Present" } }
}
PS>
Start-DscConfiguration -Path .\WebServerConfig -Wait
-Verbose
```

Windows Terminal: Tabs, Panes & Profiles

- Windows Terminal hosts PowerShell, Command Prompt, and WSL sessions side by side in one window
- Split panes (like tmux) let you watch a log in one pane while running commands in another
- Named profiles and color schemes make it easy to tell production sessions apart from test
- Sessions can be saved and restored as a Terminal 'workspace' across reboots

windows-terminal-tips.ps1

```
# Split pane vertically:    Alt+Shift+ (plus sign)
# Split pane horizontally:  Alt+Shift+ (minus sign)
# New tab:                  Ctrl+Shift+t
# Switch tabs:               Ctrl+Tab
PS>
wt -p "Windows PowerShell" ; split-pane -p "Ubuntu"
PS>
$PROFILE   # path to your PowerShell profile script
```

w32tm: Accurate Time & Domain Sync

- The Windows Time service (W32Time) keeps every domain-joined machine in sync with the PDC emulator
- Kerberos authentication fails if client and domain controller clocks drift more than 5 minutes
- w32tm /query shows current source and offset; /resync forces an immediate correction
- The forest root PDC emulator should sync against an external, reliable NTP source

time-sync.ps1

```
PS>
w32tm /query /status
PS>
w32tm /query /source
PS>
w32tm /resync /force
PS>
w32tm /config /manualpeerlist:"time.windows.com"
/syncfromflags:manual /update
PS>
Get-Date
```

Windows Services Fundamentals

Windows Services are the equivalent of systemd units — background processes managed by the Service Control Manager (SCM).

S

Service Control Manager

The SCM (services.msc) starts, stops, and monitors every registered service

T

Startup Types

Automatic, Automatic (Delayed Start), Manual, and Disabled control boot behavior

R

Recovery Actions

Configure automatic restart or a script to run if a service crashes

D

Dependencies

Services can depend on other services starting first, same as systemd's After=

Managing Windows Services

- Get-Service and Set-Service are the modern PowerShell way to inspect and control services
- sc.exe remains useful for scripting service dependencies and recovery options
- Restart-Service is the safe way to bounce a service without a full reboot
- Always check the Event Log first when a service won't start

service-mgmt.ps1

```
PS>
Get-Service -Name Spooler
PS>
Set-Service -Name Spooler -StartupType Automatic
PS>
Restart-Service -Name Spooler -Force
PS>
sc.exe failure Spooler reset=86400
actions=restart/60000
PS>
Get-Service | Where-Object {$_.Status -eq "Stopped" -
and $_.StartType -eq "Automatic"}
```

Event Viewer & Centralized Logging

- Event Viewer organizes logs into System, Application, Security, and Setup categories
- Get-WinEvent is the modern, faster cmdlet for querying event logs from PowerShell
- Windows Event Forwarding (WEF) centralizes logs from many servers onto one collector
- Filter by Event ID first — it's the fastest way to narrow down a known issue

event-viewer.ps1

```
PS>
Get-WinEvent -LogName System -MaxEvents 20
PS>
Get-WinEvent -FilterHashtable @{LogName='System';
Id=7034} -MaxEvents 5
PS>
Get-EventLog -LogName Application -EntryType Error -
Newest 10
PS>
wevtutil qe System /q:"*[System[(EventID=41)]]"
/f:text
```

03

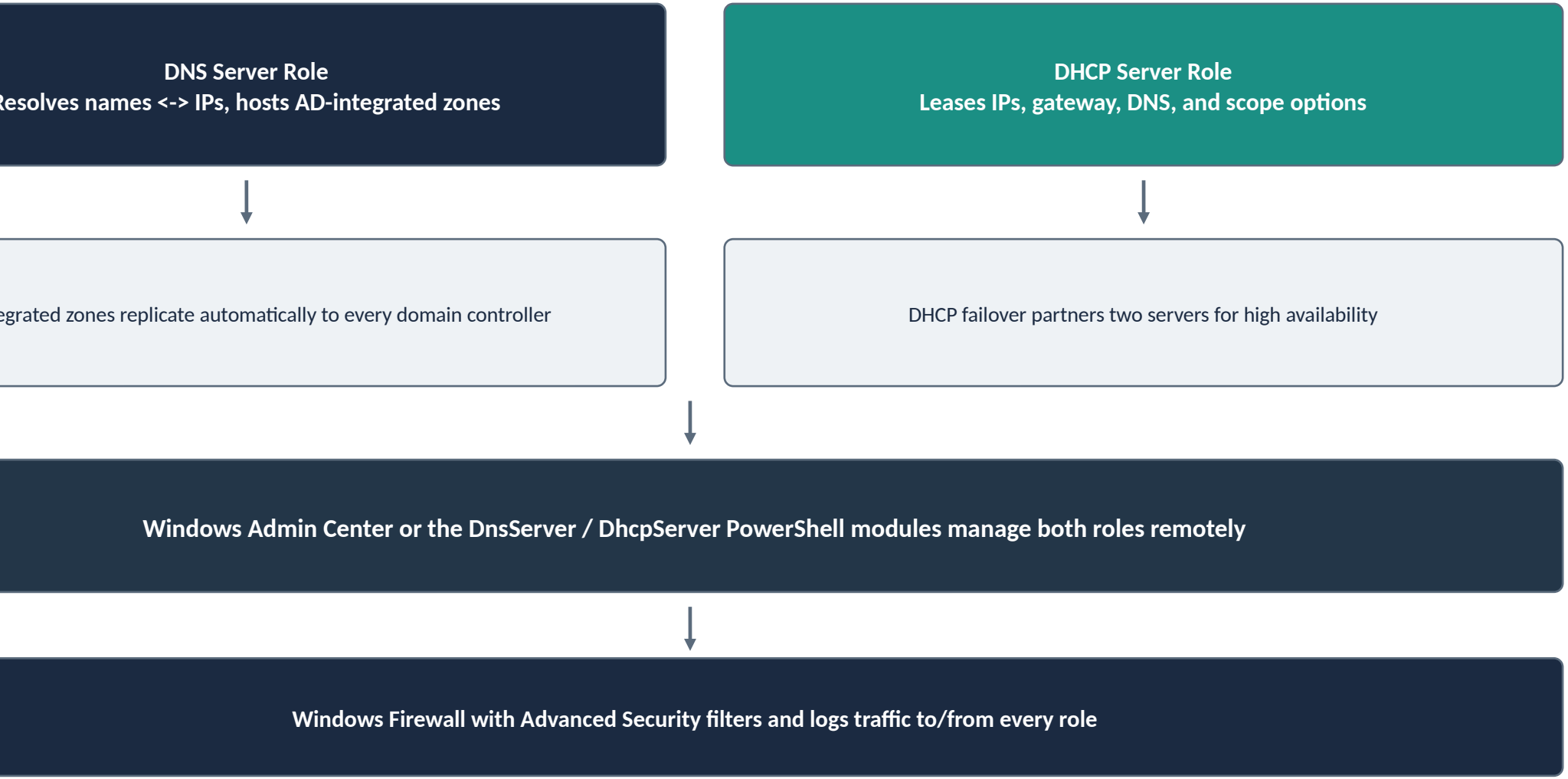
MODULE 3

Networking

Configure IP addressing with PowerShell, stand up DNS and DHCP server roles, and protect the host with Windows Firewall with Advanced Security.

DNS | DHCP | PowerShell Networking | Windows Firewall

Core Networking Roles: DNS & DHCP



PowerShell Networking Basics

- The NetTCPIP module replaces most netsh commands with discoverable, object-based cmdlets
- Get-NetIPConfiguration gives a one-screen summary of every active adapter
- New-NetIPAddress and Set-DnsClientServerAddress configure static addressing without touching the GUI
- Test-NetConnection is the modern replacement for ping + telnet-style port checks

network-basics.ps1

```
PS>
Get-NetIPConfiguration
PS>
New-NetIPAddress -InterfaceAlias Ethernet -IPAddress
192.168.1.50 \
    -PrefixLength 24 -DefaultGateway 192.168.1.1
PS>
Set-DnsClientServerAddress -InterfaceAlias Ethernet -
ServerAddresses 1.1.1.1,1.0.0.1
PS>
Test-NetConnection -ComputerName SRV01 -Port 443
```

DHCP Server Role Hands-On



- Install the DHCP role, then create a scope defining the address pool and options
- DHCP failover pairs two servers so leases keep renewing if one goes offline
- Reservations bind a MAC address to a fixed IP for servers, printers, and infrastructure
- Authorize the DHCP server in Active Directory before clients will accept its offers

dhcp-role.ps1

```
PS>
Install-WindowsFeature DHCP -IncludeManagementTools
PS>
Add-DhcpServerv4Scope -Name "Main Office" -StartRange
192.168.1.100 \
    -EndRange 192.168.1.200 -SubnetMask 255.255.255.0
PS>
Set-DhcpServerv4OptionValue -Router 192.168.1.1 -
DnsServer 192.168.1.10
PS>
Add-DhcpServerInDC -DnsName dhcp01.contoso.com
```

DNS Server Role Hands-On

- Active Directory-integrated zones replicate automatically to every domain controller
- Forwarders send queries the server can't answer locally out to an upstream resolver
- Get-DnsServerResourceRecord is the PowerShell way to audit zone contents at scale
- Aging & scavenging clean up stale records left behind by decommissioned machines

dns-role.ps1

```
PS>
Install-WindowsFeature DNS -IncludeManagementTools
PS>
Add-DnsServerPrimaryZone -Name contoso.com -
ReplicationScope Forest
PS>
Add-DnsServerResourceRecordA -ZoneName contoso.com -
Name web01 -IPv4Address 192.168.1.20
PS>
Set-DnsServerForwarder -IPAddress 1.1.1.1,1.0.0.1
PS>
Get-DnsServerResourceRecord -ZoneName contoso.com
```

Windows Firewall with Advanced Security

- Three profiles — Domain, Private, Public — apply different rule sets automatically per network
- New-NetFirewallRule scripts rule creation instead of clicking through the MMC snap-in
- Rules can filter by port, program path, or even a specific Windows service
- Get-NetFirewallRule audits what's currently allowed before you make changes

firewall.ps1

```
PS>
New-NetFirewallRule -DisplayName "Allow HTTPS" -
Direction Inbound \
    -Protocol TCP -LocalPort 443 -Action Allow
PS>
New-NetFirewallRule -DisplayName "Allow RDP from
Office" -Direction Inbound \
    -Protocol TCP -LocalPort 3389 -RemoteAddress
10.0.0.0/24 -Action Allow
PS>
Get-NetFirewallRule -Direction Inbound -Enabled True
| Select-Object DisplayName
PS>
Set-NetFirewallProfile -Profile Public -
DefaultInboundAction Block
```

04

MODULE 4

Security & Cryptography

Go deep on secured-core protections, BitLocker and PKI, the new hotpatching model, modern compliance baselines, and multi-factor authentication.

Secured-Core | BitLocker | PKI / AD CS | Hotpatching | Compliance | MFA | LAPS

Secured-Core Server Fundamentals

Secured-core defenses start below Windows itself — in firmware and hardware — before the OS even finishes booting.

V

Virtualization-Based Security

Uses the hypervisor to isolate a secure region of memory from the rest of the OS

C

Credential Guard

Enabled by default in 2025 — isolates credentials in a hardware-protected container against Pass-the-Hash attacks

T

TPM 2.0 + Secure Boot

Verifies firmware and boot components haven't been tampered with before Windows loads

D

Driver Signature Enforcement

Blocks unsigned or vulnerable drivers from loading into the kernel

Credential Guard: Verify & Troubleshoot

- System Information (msinfo32) or PowerShell both confirm whether Credential Guard is actually running
- Credential Guard requires UEFI, Secure Boot, and virtualization extensions enabled in firmware
- Some legacy authentication methods (NTLMv1, unconstrained delegation) are incompatible and get blocked
- Group Policy controls Credential Guard centrally across an entire OU

credential-guard.ps1

```
PS>
Get-ComputerInfo | Select-Object
DeviceGuardSecurityServicesRunning
PS>
(Get-CimInstance -ClassName Win32_DeviceGuard -
Namespace root\Microsoft\Windows\
DeviceGuard).SecurityServicesRunning
# Group Policy: Computer Configuration >
Administrative Templates >
#   System > Device Guard > Turn On Virtualization
Based Security
```

Windows Server 2025 Security: The Big Picture



Secured-Core Server

- Virtualization-Based Security (VBS)
- Credential Guard enabled by default
- TPM 2.0 + Secure Boot + UEFI

Data Protection

- BitLocker full-volume encryption
- SMB over QUIC (VPN-free, encrypted file access)
- TLS 1.3 by default

Identity & Access

- Windows Hello for Business
- FIDO2 security keys via Entra ID
- LAPS for local admin password rotation

Secured-core defenses start below the OS — in firmware and hardware — before Windows even boots.

BitLocker Full-Volume Encryption

- BitLocker encrypts an entire volume, typically unlocked automatically via the TPM at boot
- Recovery keys should be backed up to Active Directory or Microsoft Entra ID, never left only on the machine
- Enable on both the OS volume and any data volumes holding sensitive information
- manage-bde is the classic CLI; the BitLocker PowerShell module is the modern equivalent

bitlocker.ps1

```
PS>
Enable-BitLocker -MountPoint "C:" -EncryptionMethod
XtsAes256 -TpmProtector
PS>
Add-BitLockerKeyProtector -MountPoint "C:" -
RecoveryPasswordProtector
PS>
Backup-BitLockerKeyProtector -MountPoint "C:" -
KeyProtectorId <id>
PS>
Get-BitLockerVolume
PS>
manage-bde -status
```

PKI Basics with Active Directory Certificate Services

- AD CS turns a domain controller (or dedicated server) into your own internal Certificate Authority
- Certificate templates define what a cert can be used for and who can request one
- Auto-enrollment via Group Policy issues machine and user certs with zero manual steps
- Useful for internal TLS, smart-card logon, and code signing without buying public certs for everything

adcs.ps1

```
PS>
Install-WindowsFeature AD-Certificate -
IncludeManagementTools
PS>
Install-AdcsCertificationAuthority -CAType
EnterpriseRootCA
PS>
Get-CertificationAuthority
PS>
certreq -submit request.req certnew.cer
PS>
Get-ChildItem Cert:\LocalMachine\My
```

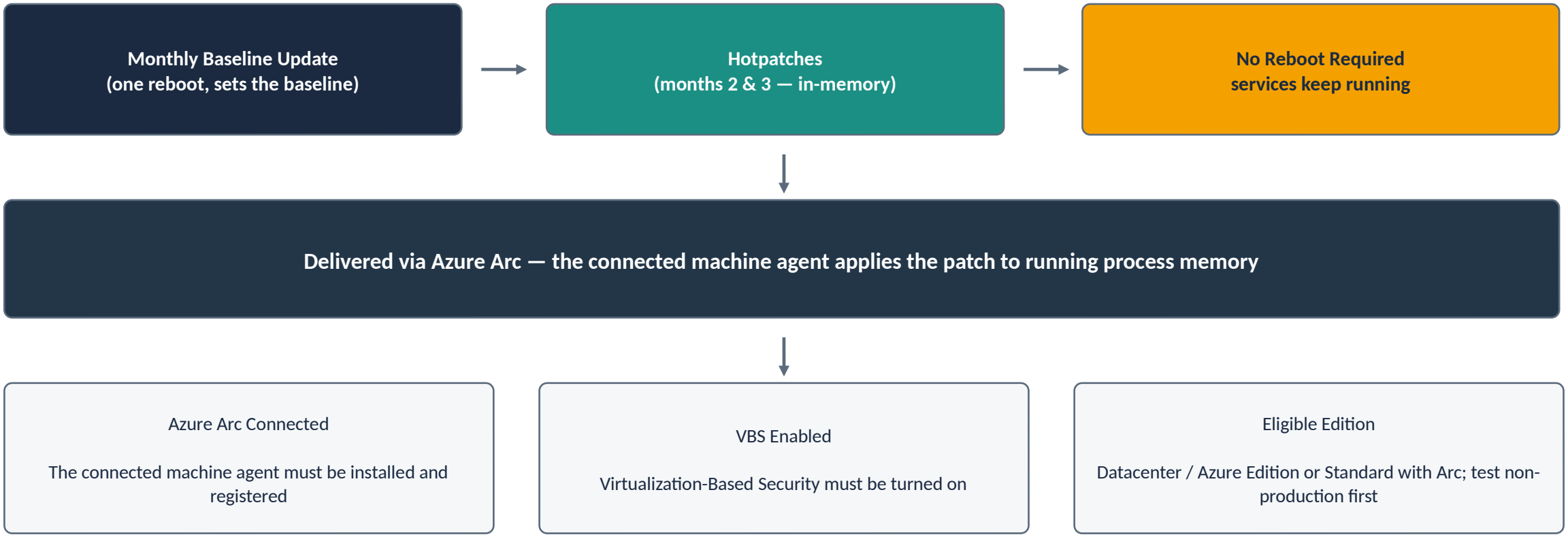
SSL/TLS: Certificates & IIS Bindings

- New-SelfSignedCertificate is handy for lab and internal testing scenarios
- IIS bindings tie a certificate to a specific site, hostname, and port
- TLS 1.3 is enabled by default on Windows Server 2025 for modern client compatibility
- Test-NetConnection and openssl (via WSL) both help verify what a server actually presents

tls-iis.ps1

```
PS>
New-SelfSignedCertificate -DnsName
"web01.contoso.com" -CertStoreLocation "cert:\
LocalMachine\My"
PS>
New-WebBinding -Name "Default Web Site" -Protocol
https -Port 443 -SslFlags 1
PS>
(Get-ChildItem IIS:\SslBindings)
PS>
Get-TlsCipherSuite | Select-Object Name
```

Hotpatching: Patch Without Rebooting



Cuts Patch Tuesday reboots from every month to roughly once a quarter for eligible systems.

Compliance & Hardening Baselines



1	STIG	0
Toolkit: Microsoft Security Compliance Toolkit	DoD hardening profile available	Manual GPO edits needed with baselines

- The Security Compliance Toolkit ships pre-built GPO baselines matching Microsoft's own hardening recommendations
- DISA STIG-compliant security templates are published for Windows Server 2025 for government/defense use
- Microsoft Defender for Identity and Microsoft Defender for Endpoint layer on top for continuous monitoring
- Compare a live server's actual configuration against a baseline with Policy Analyzer before rolling out

Post-Quantum Cryptography Readiness

- Microsoft is rolling out post-quantum algorithms (ML-KEM, ML-DSA) into the CNG/SymCrypt cryptographic stack
- The goal mirrors Linux's approach: hybrid classical + post-quantum key exchange during the transition period
- Watch Windows Update and the SymCrypt open-source repo for the current rollout status on Server 2025
- Start inventorying which of your applications call CNG directly versus relying on higher-level APIs

pqc-readiness.ps1

```
PS>
[System.Environment]::OSVersion.Version
PS>
Get-TlsCipherSuite | Select-Object -First 5
# Track rollout: aka.ms/pqcrypto and the SymCrypt
GitHub repository
```

NOTE This is an evolving area — always confirm current algorithm support against Microsoft's latest documentation before relying on it.

Multi-Factor Authentication: Windows Hello & FIDO2

- Windows Hello for Business replaces passwords with a PIN or biometric tied to the specific device
- FIDO2 security keys provide phishing-resistant sign-in for both cloud (Entra ID) and hybrid environments
- Conditional Access policies in Entra ID can require MFA based on location, device, or risk signals
- Legacy password-only authentication should be phased out wherever compliance allows

mfa-hello.ps1

```
PS>
Get-MsolUser | Select-Object DisplayName,
StrongAuthenticationRequirements
# Configure via Intune or Group Policy:
#   Computer Configuration > Administrative Templates
> Windows Components >
#   Windows Hello for Business
PS>
dsregcmd /status   # verify Entra ID hybrid join
state
```

Windows LAPS: Local Admin Password Solution

- Windows LAPS (built into the OS since 2022/2025) randomizes and rotates each machine's local admin password
- Passwords are stored securely in Active Directory or Microsoft Entra ID, readable only by authorized admins
- Eliminates the classic risk of one shared local admin password across an entire fleet
- Rotation can be triggered automatically on a schedule or manually after an admin retrieves a password

laps.ps1

```
PS>
Install-WindowsFeature -Name LAPS
PS>
Update-LapsADSchema
PS>
Set-LapsADComputerSelfPermission -Identity
"OU=Servers,DC=contoso,DC=com"
PS>
Get-LapsADPassword -Identity SRV01 -AsPlainText
PS>
Reset-LapsPassword
```

Hybrid Identity with Microsoft Entra Connect

Most real-world Windows Server environments bridge on-prem Active Directory with cloud identity in Microsoft Entra ID.

S

Password Hash Sync

The simplest sync method — a hash of the password hash syncs to the cloud, never the plaintext

P

Pass-Through Authentication

Cloud sign-in validates directly against on-prem AD in real time, no password stored in the cloud

F

Federation (AD FS)

Delegates authentication entirely to an on-prem federation server for the strictest control

S

Seamless SSO

Domain-joined devices sign in to cloud apps without an extra prompt

05

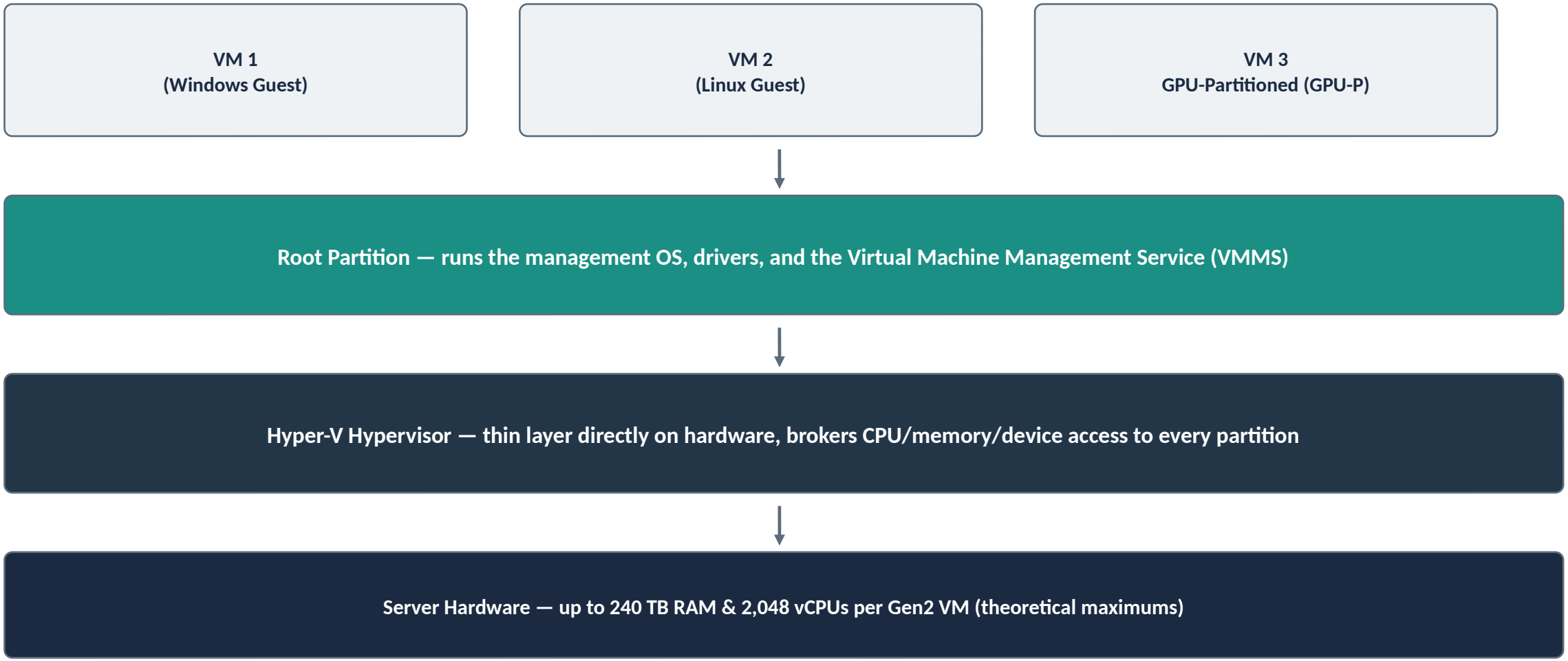
MODULE 5

Containers & Virtualization

Go hands-on with Hyper-V and GPU partitioning, explore modern storage with Storage Spaces Direct and ReFS, and get productive with Windows Containers.

Hyper-V | GPU-P | Live Migration | Storage Spaces Direct | Windows Containers

Hyper-V Architecture



Hyper-V Hands-On with PowerShell

- Every action available in Hyper-V Manager's GUI has an equivalent PowerShell cmdlet — perfect for automation
- New-VM creates a virtual machine; New-VHD creates its virtual disk
- Checkpoints (snapshots) let you roll back a VM to a known-good state before risky changes
- This course pairs well with a dedicated Hyper-V Manager walkthrough for the full GUI tour

hyperv-basics.ps1

```
PS>
New-VM -Name "WebServer01" -MemoryStartupBytes 4GB -
Generation 2 \
    -NewVHDPath "D:\VMs\WebServer01.vhdx" -
NewVHDSizeBytes 60GB
PS>
Start-VM -Name "WebServer01"
PS>
Checkpoint-VM -Name "WebServer01" -SnapshotName
"Before-Update"
PS>
Get-VM | Select-Object Name, State, CPUUsage,
MemoryAssigned
```

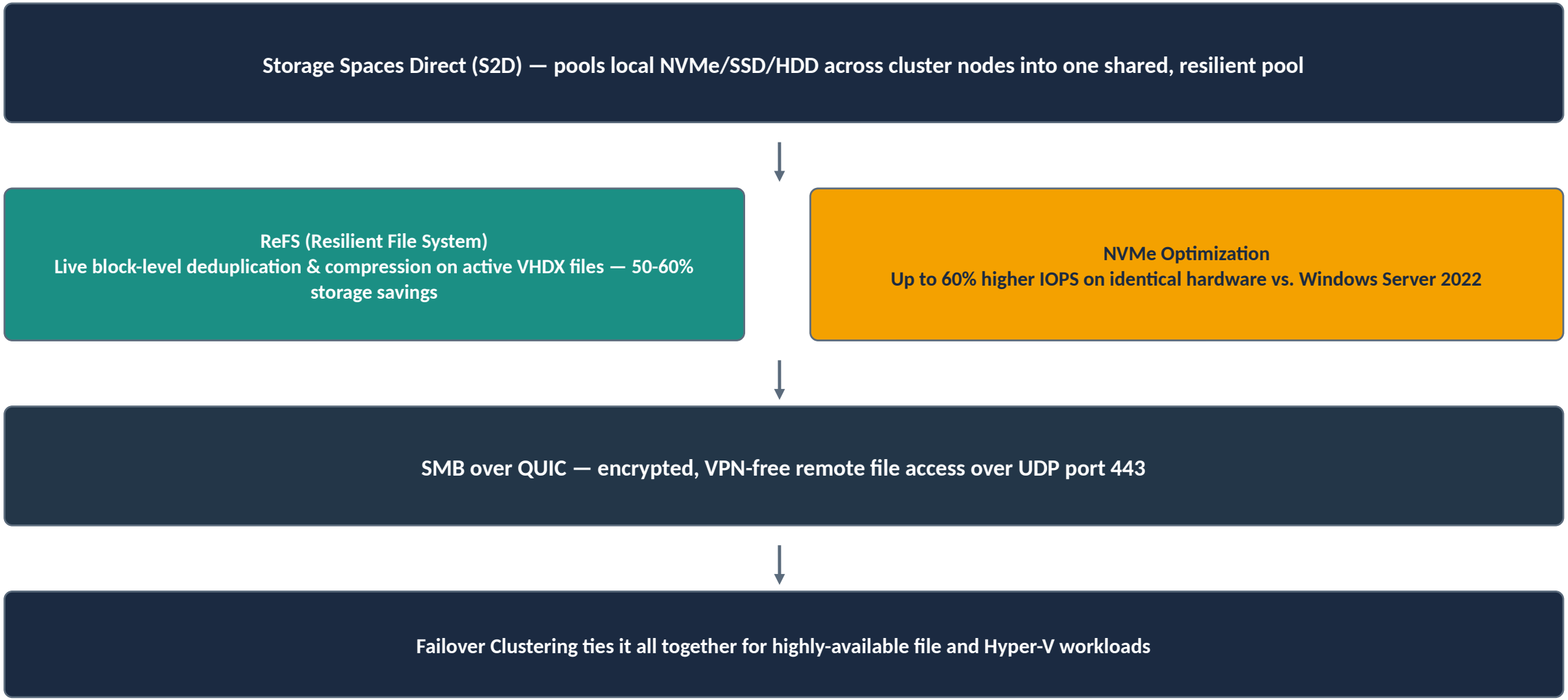
GPU Partitioning (GPU-P) & Live Migration

- GPU-P securely splits one physical GPU across multiple VMs — ideal for AI inferencing and graphics workloads
- 2025 adds live migration support for GPU-partitioned VMs, previously a major limitation
- Accelerated Networking (AccelNet) reduces latency and CPU load for clustered VM networking via SR-IOV
- Dynamic processor compatibility mode eases live migration across mixed-generation cluster hardware

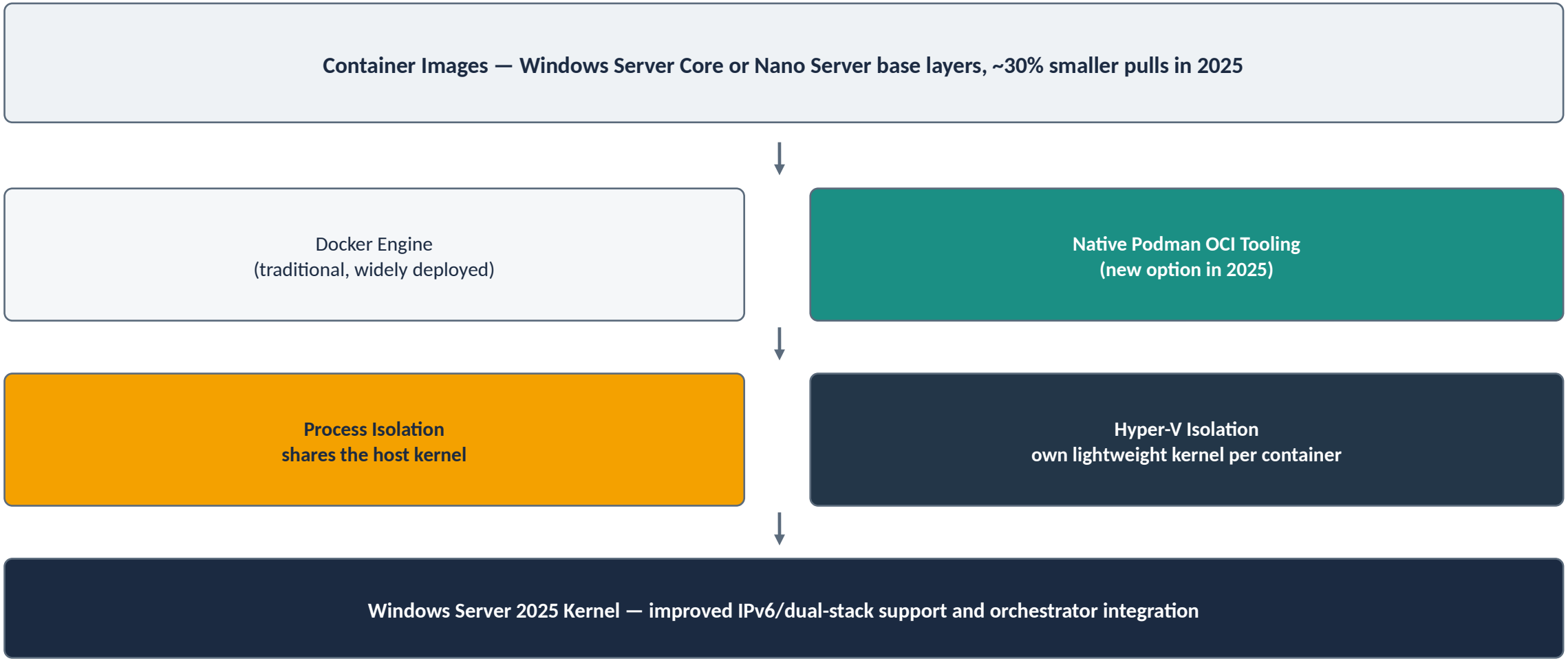
gpu-p.ps1

```
PS>
Add-VMGpuPartitionAdapter -VMName "AI-Worker01"
PS>
Set-VMGpuPartitionAdapter -VMName "AI-Worker01" -
MinPartitionVRAM 80000000
PS>
Move-VM -Name "AI-Worker01" -DestinationHost "HV-
NODE02"
PS>
Get-VMNetworkAdapter -VMName "AI-Worker01" | Select-
Object IovWeight
```

Modern Storage: S2D, ReFS & NVMe



Windows Containers Architecture



Windows Containers Hands-On

- Windows Server 2025 supports both Docker and native Podman OCI tooling for containers
- Base images are roughly 30% smaller in 2025, speeding up pulls and cold starts
- Choose process isolation for speed, or Hyper-V isolation when you need VM-grade separation
- IPv6 and dual-stack networking now work cleanly inside Windows containers

windows-containers.ps1

```
PS>
docker pull
mcr.microsoft.com/windows/servercore:ltsc2025
PS>
docker run -d --name web -p 8080:80
mcr.microsoft.com/windows/servercore:ltsc2025
PS>
docker run --isolation=hyperv -d
mcr.microsoft.com/windows/nanoserver:ltsc2025
PS>
podman run -d
mcr.microsoft.com/windows/nanoserver:ltsc2025
```

06

MODULE 6

Development Toolchains

Tour the compiler and language ecosystem on Windows Server 2025 — PowerShell 7, .NET, Python, Node.js, Java, and C/C++ — both directly on the host and inside containers.

PowerShell 7 / .NET | Python | Node.js | Java | MSVC C/C++ | WSL2

Dev Toolchains on Windows Server 2025

Native Windows Tools

- PowerShell 7 + .NET 8/9 for cross-platform scripting and apps
- Visual Studio Build Tools provide the MSVC C/C++ compiler toolchain
- winget installs Python, Node.js, and OpenJDK straight from official sources

WSL2: A Real Linux Kernel

- Windows Subsystem for Linux runs an actual Linux kernel alongside Windows Server
- Gives you apt, gcc, and every tool from the earlier Ubuntu module on the same box
- Popular for teams standardizing build scripts across Linux and Windows targets

Many shops build Linux containers via WSL2 and Windows containers natively, side by side, on the same Windows Server host.

PowerShell 7 & .NET Demo

- PowerShell 7 (built on .NET) runs side by side with the older Windows PowerShell 5.1
- dotnet is the CLI for building, running, and publishing .NET applications
- PowerShell 7 adds parallel pipelines (ForEach-Object -Parallel) not available in 5.1
- Cross-platform: the same PowerShell 7 scripts run on Windows, Linux, and macOS

pwsh-dotnet.ps1

```
PS>
winget install Microsoft.PowerShell
PS>
pwsh --version
PS>
dotnet new console -o MyApp
PS>
cd MyApp; dotnet run
PS>
1..5 | ForEach-Object -Parallel { "Processing $_" } -
ThrottleLimit 3
```

Python & Node.js on Windows Server

- winget installs the official Python and Node.js builds with no manual downloads
- py -0 lists every Python version installed side by side via the Python Launcher
- python -m venv keeps project dependencies isolated, exactly as on Linux
- npm works identically to any other platform once Node.js is installed

python-node.ps1

```
PS>
winget install Python.Python.3.13
PS>
py -0
PS>
python -m venv C:\venvs\app
PS>
C:\venvs\app\Scripts\Activate.ps1
PS>
winget install OpenJS.NodeJS.LTS
PS>
node --version; npm install express
```

Java (OpenJDK) & C/C++ (MSVC) Development

- winget installs OpenJDK builds directly; JAVA_HOME still needs to be set for some build tools
- Visual Studio Build Tools provide cl.exe (MSVC) without installing the full Visual Studio IDE
- The Developer PowerShell for VS shortcut sets up all MSVC environment variables automatically
- CMake is the common denominator for cross-platform C/C++ projects targeting both Windows and Linux

java-cpp.ps1

```
PS>
winget install Microsoft.OpenJDK.21
PS>
java -version; javac HelloWorld.java; java HelloWorld
PS>
winget install Microsoft.VisualStudio.2022.BuildTools
PS>
cl.exe /EHsc main.cpp /Fe:app.exe
PS>
cmake -B build && cmake --build build
```

Dev Toolchains: Host vs. Container



On the Host (Windows Server)

- winget install Microsoft.PowerShell (PowerShell 7)
- winget install Python.Python.3.13 / OpenJDK / Node.js
- Visual Studio Build Tools for MSVC C/C++
- WSL2 gives a real Linux kernel for gcc, apt, etc.

Inside a Windows Container

- FROM mcr.microsoft.com/windows/servercore:ltsc2025
- RUN Install-Package or winget inside the Containerfile
- Every image pins its own toolchain version
- Multi-stage builds keep the final runtime image small

```
FROM mcr.microsoft.com/windows/servercore:ltsc2025 AS build
RUN powershell -Command \
    Invoke-WebRequest -Uri <sdk-url> -OutFile sdk.exe; Start-Process sdk.exe -Wait
FROM mcr.microsoft.com/windows/nanoserver:ltsc2025
COPY --from=build C:\app C:\app
```

Building a Windows Container Image

- A Dockerfile for Windows containers looks familiar, but base images and shell syntax differ from Linux
- Multi-stage builds work the same way — build in one stage, copy only the output to a slim final image
- Nano Server makes an excellent minimal runtime base once the build stage is done
- Windows containers must match (or be compatible with) the host's kernel build

Dockerfile

```
FROM mcr.microsoft.com/windows/servercore:ltsc2025 AS
build
SHELL ["powershell", "-Command"]
COPY app/ C:\src\
RUN dotnet publish C:\src -o C:\out

FROM mcr.microsoft.com/windows/nanoserver:ltsc2025
COPY --from=build C:\out C:\app
ENTRYPOINT ["C:\\app\\MyApp.exe"]
```

Observability & Fleet Management



P

Performance Monitor

Built-in counters for CPU, memory, disk, and network — exportable for long-term trend analysis

W

Windows Admin Center

Browser-based management console for Server Core, clusters, and hyperconverged infrastructure

A

Azure Arc & Update Manager

Unified inventory, policy, and patch visibility across on-prem and cloud servers

E

Event Forwarding

Centralizes logs from many servers onto one collector for fleet-wide analysis

You're Ready to Administer Windows Server 2025

From PowerShell fundamentals to Hotpatching and GPU-partitioned Hyper-V — the full stack, junior to senior.

Practice	Rebuild every command shown in a lab VM — muscle memory beats memorization.
Go Deeper	Read the official Windows Server 2025 documentation on Microsoft Learn for full technical detail.
Stay Current	Windows Server ships security & feature updates on a predictable LTSC cadence.
Thank You	Questions and feedback are always welcome — happy administering!

07

MODULE 7

AI & Free Learning Resources

One more module: the real story of Copilot on Windows Server (it was added, then pulled), what AI tooling remains genuinely useful, and how to keep learning for free.

- The Copilot Story
- Security Copilot
- 43 Free Chapters
- eBook / Paperback / Hardcover
- Subscribe & Share

The Real Copilot-on-Windows-Server Story

Unlike RHEL's built-in Lightspeed assistant, Microsoft actually tried and then removed a desktop Copilot from Windows Server — a useful, honest case study.

1

It Shipped by Accident

Copilot appeared in a Windows Server 2025 preview build in early 2024, surprising admins

2

Admins Pushed Back

A chat assistant on a headless server role drew widespread criticism as unnecessary overhead

3

Microsoft Removed It

Copilot was pulled from the server desktop about a month later in a subsequent build

4

What Remains Useful

Copilot in the Microsoft 365 admin center, Security Copilot, and Edge-based Copilot are still available where they add real value

A Real AI-Assisted Troubleshooting Example

- Without a built-in server assistant, piping real diagnostic output to any AI chat tool works just as well
- Security Copilot (a separate, licensed product) does this natively for security investigations at enterprise scale
- Always read and understand a suggested fix before running it on a production domain controller
- This pattern works identically whether you're troubleshooting Windows, RHEL, Rocky, or Ubuntu

ai-troubleshoot.ps1

```
PS>
Get-WinEvent -LogName System -MaxEvents 20 | Out-
String > events.txt
# Paste events.txt into any AI chat tool and ask:
#   "Why is this service crashing, and what's the
safest fix?"
# Review the suggested fix, then run it yourself:
PS>
Restart-Service -Name Spooler -Force
```

NOTE No AI tool replaces understanding the fix — use it to learn the reasoning, not just to copy-paste commands.

Read the Complete 43-Chapter Linux Book — Free

43

Chapters free to read online

2

Languages: English & Dari

3

Starter chapters also as PDF

- All 43 chapters of "The Direct Path to Linux Ubuntu" are free to read online, in both English and Dari — no signup required
- The Preface + first 3 chapters are also offered as a downloadable PDF, formatted exactly like the printed book — a perfect starter kit
- A companion title on Windows Server is planned from the same author
- Every link lives in one place: the MyWebUniversity.com homepage

Prefer Print or Kindle? Get the Complete Book

Amazon KDP

- eBook — $9.99
- Paperback — $34.99
- Hardcover — $44.99

Barnes & Noble Press

- eBook — $9.99
- Paperback — $24.99
- Hardcover — $44.99

Available in English, with a full Dari-language edition too — every link is on the MyWebUniversity.com homepage.

Enjoyed This Course? Let's Stay Connected

Free Linux, Windows, and Enterprise IT training — new videos every week.

Subscribe	@MyWebUniversity (English) and @MyWebUniversity-Dari (Dari) — 2,600+ free videos and counting
Comment	Tell us which Windows Server topic you want covered next
Share	Send this video to a teammate studying for their next Microsoft certification
Read & Buy the Book	Free chapters, the full 43-chapter book, and every link — all on MyWebUniversity.com