AD Domain Migration and Infrastructure Realignment

by Rob Das

DTEK Consulting Services Ltd.
Alberta, Canada

January 2025 – June 2025

This project report documents the Active Directory domain migration and infrastructure realignment project. The target audience is: sys admins.

1. Executive Summary

This project was initiated in response to an Active Directory domain-level disruption that triggered a formal review of the underlying infrastructure. The failure exposed the potential single point of failure posed by the original primary domain controller, a Windows Server 2012 domain controller for the company's Active Directory domain. An attempted failover to the secondary domain controller revealed that it was non-functional, likely due to extensive legacy data corruption and aging replication metadata.

Given the instability and age of the domain environment, a decision was made to transition away from the original domain entirely, which led to the creation of a new, cleanly deployed Active Directory domain built using best practices and updated Windows Server systems. The project entailed building new domain controllers, rearchitecting infrastructure roles, migrating workstations and services, centralizing windows updates, and establishing a repeatable, standards-based configuration for long-term stability.

The result is a more reliable, secure, and maintainable domain environment that removes legacy dependencies and lays a foundation for future scalability.

Note: for security reasons since this document is published online, some of the internal domain and computer names have been changed.

2. Project Overview

The domain migration project began after a Windows Update maintenance cycle caused the original primary domain controller for the company's internal Active Directory domain, *dtek.internal*, to temporarily go offline. Despite the presence of a secondary domain controller, domain services did not fail over properly, resulting in a temporary domain-wide outage. Extensive troubleshooting revealed that replication between the controllers had long been broken, and the stale replication metadata within the domain's internal databases made recovery infeasible. the original secondary domain controller was demoted and removed from the domain, exposing the original primary domain controller as a potential single point of failure.

A temporary remediation effort was undertaken to stabilize the original primary domain controller so that core authentication services could continue functioning while a new environment was planned. During this period, attempts to promote a new secondary domain controller into the existing domain failed due to persistent data integrity issues.

At that point, the decision was made to build a brand-new domain, *d1.internal*, from the ground up, with clean Active Directory configuration, hardened DNS, and improved operational practices. A pair of domain controllers (*dc-01* and *dc-02*) were deployed on updated Windows Server systems hosted on dedicated Hyper-V hypervisors. Existing systems were assessed for migration, and a staged migration plan was developed to move production systems either into the new domain or into standalone WORKGROUP configurations, depending on their role.

This project also included a significant infrastructure realignment:

- Replacing multi-purpose physical servers with role-dedicated hypervisors.
- Creating a new virtualized file server.
- Migrating user profiles and application services.
- Establishing operational standards and checklists for long-term maintainability and disaster recovery.
- Introducing centralized Windows Update management to streamline patching and reduce admin costs.

With the completion of the migration and decommissioning of the original primary domain controller, the company now operates on a more resilient and standards-compliant foundation.

3. Infrastructure Summary

This section provides a summary of the core infrastructure elements that were implemented, replaced, or restructured as part of the domain migration project. It reflects the current state of the company's IT environment following the completion of the transition from dtek.internal to d1.internal.

3.1 Active Directory and Domain Controllers

The new d1.internal domain is served by two Windows Server 2019 virtual machine domain controllers hosted on separate Hyper-V hypervisors. These domain controllers were configured from scratch using Microsoft-recommended defaults, with replication configured and verified

- dc-01: Primary domain controller hosted on Hyper-V server sirius.
- dc-02: Secondary domain controller hosted on Hyper-V server titan.

3.2 Hyper-V Hosts

The company's virtualization infrastructure was reorganized to move away from multipurpose physical servers. All critical workloads were moved to virtual machines hosted on dedicated Hyper-V servers:

- lunar (Windows Server 2016, retained due to long-standing stability)
- europa (Windows Server 2019)
- titan (Windows Server 2019)
- sirius (Windows Server 2019)

Each hypervisor is configured as a standalone host in WORKGROUP mode and does not participate in domain services.

3.3 File Server Services

The file server infrastructure was reorganized to simplify the role of Hyper-V hosts and establish clear separation of responsibilities. Previously, europa served both as a Hyper-V host and as a file server. A new virtual machine named file-01, running Windows Server 2016, was created to assume all file server responsibilities previously handled by europa.

- file-01 is now the second file server in the organization.
- It is hosted on europa and uses a dedicated VHDX volume backed by local storage.

- File shares were migrated from europa's physical F: drive to file-01, and a new access model based on domain security groups was implemented to follow the principle of least privilege.

In parallel, significant cleanup efforts were made on an existing file server named lunar-file. While structural improvements were implemented, such as removing redundant shares and standardizing folder layout, additional work is still planned to bring its permissions model in line with the standards used on file-01.

3.4 Domain Membership

Each computer in the organization was reviewed for its function, and then reassigned to one of two network roles:

- domain members: systems requiring Active Directory services were joined to d1.internal
- workgroup systems: systems with no dependency on domain services remained or were moved to WORKGROUP

This structure ensures minimal reliance on domain services where unnecessary, reducing risk and simplifying future disaster recovery.

3.5 Legacy System Handling

Legacy systems to support dtek.internal were retired, demoted, or repurposed:

- the original primary domain controller was demoted and removed from Active Directory.
- the original secondary domain controller was decommissioned earlier in the project.
- Systems dependent on dtek.internal were reviewed and migrated case-by-case.

3.6 Database Servers

Two database servers are maintained within the updated infrastructure to support both production and development workloads:

- lunar-sql: A legacy SQL Server 2014 system that hosts production databases used by multiple internal applications. It was migrated from the dtek.internal domain to d1.internal as part of the project.

- w16-cosmosdb: A Windows Server 2016 virtual machine that runs the Azure Cosmos DB Emulator for on-premises development and testing. This server was configured using Microsoft's official guidance for emulated environments.

3.7 WSUS Server

A new Windows Server Update Services (WSUS) server was deployed to centralize Windows update distribution within the domain.

- WSUS2, a Windows Server 2016 powered WSUS server, serves as a local Windows update patch repository management system.
- WSUS reduces bandwidth usage and streamlines the monthly Windows update cycles by eliminating the need for each system to download Windows updates individually.

4. Domain Architecture

This section describes the Active Directory domain architecture.

4.1 Purpose and design of dc-01 and dc-02

The domain d1.internal is hosted by two domain controllers, both running Windows Server 2019 edition:

- dc-02 is hosted on titan
- dc-01 is hosted on sirius

All four Hyper-V hosts in the environment (lunar, europa, titan, sirius) operate independently and are treated as equal peers. Domain controller placement across different physical hosts was deliberately chosen to ensure domain availability in the event of hardware failure.

Both domain controllers provide:

- Active Directory Domain Services (AD DS)
- DNS Server
- Global Catalog

Checkpoints are explicitly avoided for both dc-01 and dc-02. On 2025-05-07, both domain controller VMs were shut down and exported to form a backup named:

```
d1-domain-controllers_2025-05-07.zip
```

4.2 Active Directory domain design and OU structure

The domain was created from scratch using the default Server Manager GUI workflow. No objects, configurations, or policies were imported from the legacy domain dtek.internal.

As of deployment, the directory contains only the default containers automatically created by Windows:

- Builtin
- Computers
- Domain Controllers
- ForeignSecurityPrincipals
- Managed Service Accounts
- Users

No Organizational Units (OUs) have been created.

4.3 DNS and DHCP configuration summary

DNS is AD-integrated and redundantly hosted on both domain controllers (dc-01 and dc-02).

- Primary zone: d1.internal
- Replication scope: To all domain controllers in the domain
- Reverse lookup zone for the 192.168.11.0/24 network was created as part of initial setup

DHCP is not managed by Windows Server. It continues to be provided by the network router at 192.168.11.1, which has not been modified and continues to serve leases to the entire subnet. This decision was made to avoid disrupting long-established network infrastructure.

4.4 Global catalog and FSMO role assignments

Global Catalog is a complete list of objects in the domain, and FSMO roles, or Flexible Single Master Operations roles, dictate which domain controller is responsible for specific types of updates and changes, preventing replication conflicts and ensuring the directory operates correctly. Both domain controllers (dc-01 and dc-02) are Global Catalog servers, as verified in Active Directory Sites and Services.

FSMO Role Holders are shown below (confirmed via “netdom query fsmo”):

- Schema Master: dc-01.d1.internal

- Domain Naming Master: dc-01.d1.internal
- PDC Emulator: dc-01.d1.internal
- RID Master: dc-01.d1.internal
- Infrastructure Master: dc-01.d1.internal

All FSMO roles are currently held by dc-01.

4.5 Group Policy overview

No custom Group Policies have been created or modified at this time. The following default GPOs exist:

- Default Domain Policy
- Default Domain Controllers Policy

These GPOs are created automatically when a domain is first established and apply baseline settings to all domain members and DCs respectively. They have not been manually edited or extended.

5. System Configuration and Services

This section describes how the various systems are configured and the services they provide.

5.1 File Server Transition and Design Intent

Historically, europa served dual roles as both a Hyper-V host and a file server. To simplify its responsibilities and improve manageability, file services were migrated to a new, dedicated file server named file-01. This system was created specifically to offload user and departmental file shares from europa, allowing europa to operate solely as a Hyper-V hypervisor going forward.

In parallel, substantial cleanup was performed on lunar-file, an older file server, to consolidate, remove redundant shares, and improve organizational clarity. Although more work remains—particularly in aligning lunar-file share permissions with the least-privilege model established on file-01—this effort marked a key step toward standardized and streamlined file service management.

Both file-01 and lunar-file are Windows Server systems joined to the d1.internal domain, managed via Server Manager and AD-integrated access controls.

5.2 Role Separation: Repurposing Multi-Role Servers into Dedicated Hypervisors

Early in the environment's lifecycle, some systems were configured with multiple roles. Over time, these systems were repurposed to isolate hypervisor responsibilities from other server roles. Specifically:

- europa was reconfigured to remove file services and act solely as a Hyper-V host.
- lunar was converted from one edition of Windows Server 2016 to another to support Hyper-V features and licensing. It remains on 2016 for stability reasons.
- sirius and titan were built as Windows Server 2019 installations from the start, dedicated to Hyper-V hosting only.

This role separation ensures better isolation, performance, and ease of maintenance.

5.3 Local Admin vs Domain Access Control

Domain-joined servers and workstations now rely on domain-based user authentication via d1.internal. Where possible, local administrator access is restricted to domain-level admin groups (D1\Administrators), rather than relying on per-machine local accounts. This approach simplifies credential management and improves auditing consistency.

Exceptions exist on standalone systems (such as those in WORKGROUP configurations) where local credentials are required. These systems are limited in scope and isolated by design.

5.4 Security Hardening and Least-Privilege Improvements

Security configurations follow a pragmatic model of high trust among internal staff. By default, all internal users are granted full access to shared file resources, consistent with company policy. File servers such as file-01 and lunar-file are configured so that most shares use "Everyone / Full Control" to streamline collaboration and minimize support overhead.

Exceptions are made for a small number of shares containing sensitive information—typically related to finance or system administration. These shares are explicitly secured using named domain accounts and group-based permissions to limit access.

Domain-based authentication is enforced across all servers and shares, ensuring only authenticated users can access internal resources. This model balances ease of access with reasonable protection of sensitive information.

6. Migration Details

This section provides a detailed overview of the migration process, including the final placement of computers, rationale for domain membership decisions, and configuration of non-standard setups.

6.1 Final Placement of Computers

The table below summarizes the current placement of all known computers across the organization. Systems were either migrated to the d1.internal domain, placed in a WORKGROUP, or in special cases left unjoined due to technical constraints.

Domain / Workgroup	Role	Number of Computers
d1.internal	App Server	3
	Batch Jobs on Task Scheduler	1
	Database Server (SQL-Server)	1
	Developer Workstation	1
	Documentation Workstation	1
	File Server	1
	General Purpose Workstation	3
	Primary DC	1
	Secondary DC	1
	Web Hosting - Dev/Test	1
	Web Hosting - Production	1

Domain / Workgroup	Role	Number of Computers
WORKGROUP	Database Server (Cosmos DB)	1
	Decommissioned DC	1
	Developer Workstation	2
	General Purpose Server	3
	General Purpose Workstation	6
	Hyper-V Host	4
mylab.local	Laboratory Computer	1
n/a	Linux Computer	1

6.2 Rationale for Domain Membership Decisions

Most systems performing core infrastructure functions (e.g., file sharing, application hosting, SQL Server) were joined to the d1.internal Active Directory domain for centralized identity management and policy enforcement.

However, several systems were intentionally not joined to the domain due to the following factors:

- Hyper-V Hosts (sirius, titan, lunar, europa) were excluded from domain membership to avoid dependency on AD availability and to improve isolation for disaster recovery purposes.
- Unsupported Operating Systems such as Windows 10/11 Home Edition and Linux do not support AD domain joining.
- Application Incompatibility: The CosmosDB emulator on w16-cosmosdb is known to behave unpredictably in domain environments, and was deliberately left unjoined.

- Unassigned / Spare Systems: Some computers have not yet been assigned for production use, so domain decisions have been deferred.

6.3 Configuration of Non-Standard Setups

Some systems deviate from standard deployment due to historical decisions or unique workloads:

- lunar-sql and lunar-tfs retain legacy TFS + SQL Server roles with domain-linked authentication still being cleaned up.
- The web servers (eu-web, webserv-test) required careful DNS and authentication coordination with SQL-based services on lunar-sql using Windows-integrated credentials.
- A few legacy computers had dual identities (e.g., previously joined to dtek.internal, then rejoined to d1.internal) which occasionally caused duplicate SPNs or SID issues during migration. These were manually resolved.

6.4 Tools and Utilities Used

The following tools were instrumental throughout the planning, execution, and validation phases of the migration:

- Active Directory Domain Services (AD DS)
- DNS Manager
- Server Manager
- Hyper-V Manager
- PowerShell and Command Prompt
- Remote Desktop Connection (RDP)
- Visual Studio – for debugging web deployments and TFS integration
- IIS Management Console – for website configuration and bindings
- SQL Server Management Studio (SSMS) – for validating SQL connectivity and security
- Microsoft Excel / Word – for inventory tracking and report documentation
- ChatGPT – for architectural planning, PowerShell scripting, troubleshooting

7. Operational Standards

Below are concise operational checklists used during migration and any future setup:

7.1 Standards Followed

Where practical, we adhere to the following industry and internal standards when managing systems (however we favor a pragmatic rather than a dogmatic approach)

- Microsoft Best Practices for Active Directory, DNS, and file sharing security.
- Principle of Least Privilege for all file share and administrative permissions.
- Hyper-V Management Guidelines to ensure safe VM exports and imports.
- Repeatable Setup using internal checklists tailored to our domain structure (d1.internal) and decentralized Hyper-V hosts.
- Workgroup Isolation Policy: All Hyper-V hosts remain in WORKGROUP mode for administrative simplicity and reduce domain dependency at the virtualization layer.

7.2 Repeatable Setup Checklists

File Share Permissions

- Use "Everyone: Full Control" at the share level (if not sensitive).
- Use NTFS-level permissions for actual access control. (Specific permissions are determined by operational needs.)

AD-Integrated DNS Setup

- Enable dynamic updates (secure only).
- Define forwarders to public resolvers (e.g., 8.8.8.8, 1.1.1.1).
- Replicate DNS zones to:
    - dc-01
    - dc-02
- To verify reverse lookup zones exist:

```
Get-DnsServerZone -ComputerName dc-01 | Where-Object { $_.IsReverseLookupZone }
```

Domain Join and Workstation Profile Transition

- Backup local user data (Desktop/Documents/Downloads).
- Join workstation to d1.internal domain.
- Log in once with domain user to create profile.
- Reconfigure network drives and printers manually.

Scheduled Task Migration

Each task scheduled on lunar-batch was updated with the appropriate domain account username and password.

Hyper-V Export and Disaster Recovery Preparation

Exporting both domain controllers together (one-time disaster recovery snapshot):

1. Shut down both DCs via Hyper-V Manager.
2. On each hypervisor host:
   - Right-click the VM > Export.
   - Target: External volume (e.g., E:\VM-Exports\YYYYMMDD\).
3. Confirm export folder includes the full VM config, VHDs, and snapshots (if any).
4. Label folders with computer name and export date.

For all other hyper-v vm exports, follow company policy.

8. Testing and Validation

Always test everything!

8.1 Domain Resilience Tests

- Each domain controller was shut down individually to verify that authentication, DNS resolution, and Group Policy processing continued without interruption.
- Verified login success from domain-joined workstations during each simulated DC outage.

8.2 Post-Migration Validation

- Confirmed domain join success and user profile creation on migrated workstations.
- Verified DNS resolution using nslookup for local and external domains.
- Tested file share access and permissions using domain credentials.
- Confirmed TFS, SQL Server, and IIS services were reachable and functional.

9. Migration of Interdependent Legacy Application Systems

One of the most intricate and time-consuming parts of the domain migration project was the coordinated transfer of five legacy systems that collectively supported a multi-tier, front-end web application and its supporting infrastructure. These systems had to be migrated together

due to tight operational coupling, shared authentication requirements, and historical design constraints.

9.1 Systems Involved

The following systems were involved:

- devbox-1: Developer workstation for legacy Visual Studio projects
- eu-web: Production front-end web server (MVC application)
- webserv-test: Staging/test web server (mirrors production setup)
- lunar-sql: Database server hosting Webserv-dev, Webserv-test, and Webserv-prod (SQL Server 2014)
- lunar-tfs: Source control server running TFS (Team Foundation Server) with its own local SQL Server instance

9.2 Domain Authentication Dependencies

The application stack relies on Windows-integrated authentication for both web-server-to-database communication and user-level access controls. This meant the migration from dtek.internal to d1.internal could not break or disrupt:

- Application pool identities used in IIS
- SQL Server logins mapped to Active Directory domain accounts
- Security roles in individual SQL databases
- External access via leased domain name and its static IP

All interdependencies had to be preserved or recreated during the transition to d1.internal, which required:

- Adding new SQL Server logins for the domain-joined machine accounts: D1\EU-WEB$, D1\WEBSERV-TEST$, and D1\DEVBOX-1$
- Ensuring corresponding roles and permissions were reassigned to these new accounts in the Webserv-dev, Webserv-test, and Webserv-prod databases
- Reconfiguring IIS application pools and web.config files
- Carefully staged system shutdowns and reboots to coordinate AD changes

9.3 TFS Source Control Migration

The lunar-tfs server hosted a legacy TFS environment backed by its own SQL Server instance. The migration involved:

- Domain join to d1.internal without breaking TFS service identity bindings
- Verifying that TFS could still access its databases and authenticate users
- Confirming continued access to project history, check-ins, and team settings

9.4 Testing and Validation

Significant testing was conducted to confirm:

- Database connectivity and application function post-migration
- Web application online availability through leased public domain name
- Correct mapping of SQL logins to domain accounts
- TFS service availability and user access

This combined migration was one of the most delicate operations in the project due to the number of moving parts, the use of Windows authentication across multiple tiers, and the live production nature of eu-web.

10. Implementing WSUS for Improved Patch Management

Windows Server Update Services (WSUS) is a software update management system that allows IT administrators to manage and deploy Microsoft product updates to client computers on a network.

10.1 Purpose and Overview

To improve the reliability and efficiency of monthly patch management across the company's internal Active Directory domain, we have implemented a centralized update infrastructure using Windows Server Update Services (WSUS). This system allows approved updates to be downloaded once and distributed internally to client systems, significantly reducing Internet bandwidth usage and ensuring consistent patch levels across servers and workstations. In contrast to the previous approach of manually updating each system from Microsoft's online services, the new WSUS-based method streamlines the update process, reduces administrative overhead, and enables better control and auditability of deployed updates.

10.2 WSUS Server Infrastructure

The WSUS server wsus2.d1.internal was deployed as a dedicated virtual machine running Windows Server 2016 Standard and joined to the d1.internal domain. It is hosted on the Hyper-V system Sirius, which was selected for its available capacity and role as a general-purpose virtualization host.

WSUS was installed using Server Manager with the Windows Internal Database (WID) as its backend. Update files are stored on a secondary virtual hard disk at E:\WSUS. The service is hosted under IIS and listens on ports 8530 (HTTP) and 8531 (HTTPS). No SSL certificate has been configured at this stage, so the system currently uses HTTP for client communication.

This setup was completed as part of a proof-of-concept and is not yet configured with a recurring synchronization schedule. Update checks were triggered manually for testing purposes only. Future revisions may include a scheduled sync and expanded update scope as more clients are onboarded.

Key configuration details:

- Hostname: wsus2.d1.internal
- OS: Windows Server 2016 Standard
- Domain: d1.internal
- Virtualization: Hosted on Sirius (Hyper-V)
- Database: Windows Internal Database (WID)
- Update Storage: E:\WSUS
- IIS Ports: 8530 (HTTP), 8531 (HTTPS)

The system was designed with recovery and maintenance in mind. Because it uses WID and stores all updates locally, the virtual machine can be exported and restored easily using Hyper-V Manager. No external database or file share dependencies exist, making the configuration highly portable and well-suited for disaster recovery planning.

10.3 Non-Default Configuration

Several configuration changes were made to optimize the WSUS server for its virtualized environment and to better support future scaling. These settings differ from the out-of-box defaults and should be reviewed and adjusted as more clients are onboarded.

Virtual Machine Settings (Hyper-V):

- Dynamic Memory enabled

    - Minimum: 2048 MB
    - Maximum: 8192 MB
- Processor: 1 virtual processor

This memory configuration strikes a balance between performance and the limited resources available on the shared Hyper-V host, Sirius. Testing confirmed this was sufficient for both server operation and update delivery to the initial client.

IIS Configuration:

- Application Pool: WsusPool
    - Private Memory Limit (KB): Set to 0 (unlimited)
- Connection Timeout: Increased to 600 seconds

The memory limit was raised to prevent the WSUS console from hanging during sync or approval operations. The connection timeout was increased to accommodate larger updates or slower response times without timeouts during management sessions.

WSUS Targeting and Grouping:

- Server-side targeting is enabled, meaning clients are assigned to WSUS computer groups manually through the WSUS console.
- A dedicated computer group was created for Windows Server 2016 systems.
- Additional groups will be created for other OS versions (e.g., Windows 10, Windows Server 2019) to allow selective approval of updates by operating system.

This structure allows administrators to approve updates either globally or for specific platform groups. It also provides a scalable framework for phased rollout and testing of future patches.

10.4 Proof-of-Concept Summary

The WSUS proof-of-concept was conducted using a single test client:

win2016-5.d1.internal, a freshly installed Windows Server 2016 system joined to the d1.internal domain.

To validate WSUS functionality, three updates were successfully deployed via the WSUS server (wsus2.d1.internal) and confirmed to have been delivered internally rather than through direct contact with Microsoft servers. The updates installed were:

- 2025-06 Cumulative Update for Windows Server 2016 for x64-based Systems (KB5061010)

- 2025-06 Servicing Stack Update for Windows Server 2016 for x64-based Systems (KB5060954)
- Windows Malicious Software Removal Tool x64 - v5.134 (KB890830)

Client Configuration

The WSUS configuration was intentionally decoupled from domain controllers, so that local Group Policy was used on the client to configure update settings instead of using domain group level policies.

Applied Policy Objects:

- Specify intranet Microsoft update service location:
  - Set the intranet update service for detecting updates: http://wsus2.d1.internal:8530
  - Set the intranet statistics server: http://wsus2.d1.internal:8530
- Configure Automatic Updates: Enabled

Not configured:

- Enable client-side targeting — server-side targeting is used exclusively.

After applying the Group Policy settings, the following sequence was run on the client to ensure policies were applied and WSUS registration was initiated:

```
gpupdate /force
wuauclt /detectnow
wuauclt /reportnow
net stop wuauserv
net start wuauserv
```

These commands were issued primarily for troubleshooting when updates did not initially appear. It was later determined that the root cause was a memory-constrained IIS application pool (WsusPool), which had stopped. Once memory settings were tuned (see Section 10.3), update detection began functioning correctly without needing further client-side intervention.

How was win2016-5 registered in WSUS during the PoC?

In WSUS, clients using server-side targeting register themselves automatically when:

1. The "Specify intranet Microsoft update service location" policy is applied on the client and the Windows Update service connects to the specified URL (in this case, http://wsus2.d1.internal:8530).
2. The client successfully checks for updates, either automatically or when manually triggered via the Windows Update GUI or by running wuauclt /detectnow or usoclient StartScan.

In the PoC, win2016-5 appeared in the WSUS console under Unassigned Computers after we opened Settings > Windows Update and clicked "Check for updates" on the client. That action initiated contact with wsus2, and the WSUS server registered the client in its database.

Verification

- Updates were approved manually in the WSUS console and assigned to the appropriate computer group, either Win2016 or All Computers.
- After successful update installation, Get-WindowsUpdateLog was used to generate an update log on the desktop, which confirmed WSUS was the source.

The successful test confirmed:

- WSUS communication via local Group Policy is functional.
- Updates can be targeted to specific OS versions using WSUS computer groups.
- Updates are properly delivered from the internal server without contacting Microsoft.
- Manual client registration steps are available when needed for testing.

This establishes a repeatable baseline for onboarding additional domain-joined systems in future rollout phases.

10.5 Steps for Adding Additional Clients

To onboard new clients into the WSUS infrastructure, follow these steps to ensure proper registration and update delivery. This procedure assumes that WSUS server-side targeting is used and that WSUS-related Group Policy is applied locally on each client (not domain-wide).

Step 1 – Domain Join

Ensure the client system is joined to the d1.internal Active Directory domain.

Step 2 – Apply Local Group Policy

On the client system, open the Local Group Policy Editor (gpedit.msc) and configure the following policies:

- Specify intranet Microsoft update service location
    - Set both intranet update detection and statistics server to http://wsus2.d1.internal:8530
    - Do not configure the alternate download server.
- Configure Automatic Updates
    - Enable the policy.
    - Leave the setting at its default value ("3 – Auto download and notify for install").

After setting these policies, run the following to apply them

```
gpupdate /force
```

Note: Client-side targeting is not used. Server-side targeting is configured on the WSUS server.

Step 3 – Initial Contact with WSUS

To trigger registration with the WSUS server, open Settings > Windows Update on the client and click Check for updates. This will initiate contact with wsus2 and cause the client to appear in the WSUS console under Unassigned Computers. Wait for automatic registration and check-in.

Once the client appears:

Step 4 – Assign to a WSUS Computer Group

In the WSUS console on wsus2, locate the new client under Unassigned Computers. Right-click the entry and move it to the appropriate group, such as Windows Server 2016.

Note: This organization uses WSUS computer groups based on OS version to allow update approvals by group.

Step 5 – Monitor Status

To confirm successful update detection and installation:

- In the WSUS console, review the client's status and update report.
- On the client, run

```
Get-WindowsUpdateLog
```

This command generates a human-readable WindowsUpdate.log file on the Desktop, which can be reviewed for troubleshooting or confirmation of update sources.

Note: No manual use of `wuauclt` or `usoclient` is necessary under normal conditions, provided WSUS is functioning and Group Policy is applied correctly.

Troubleshooting WSUS Client Registration and Communication

If a new client fails to appear in the WSUS console or doesn't receive updates, use the checklist below to isolate and fix the issue.

Confirm Group Policy Application

On the client, run:

```
gpresult /r
```

Check that "Specify intranet Microsoft update service location" is listed under applied policies.

Confirm registry entries exist under:

```
HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\WindowsUpdate
```

Expected values:

- WUServer = http://wsus2.d1.internal:8530
- WUStatusServer = http://wsus2.d1.internal:8530

If these values are missing, re-check local Group Policy or use `gpupdate /force`.

Trigger WSUS Communication

If the client doesn't check in automatically, try the following on the client

```
wuauclt /detectnow
wuauclt /reportnow
net stop wuauserv
net start wuauserv
```

Note: wuauclt is deprecated on Windows 10/11 but still works on Server 2016. Do not use usoclient unless troubleshooting Windows 10/11 specifically.

WSUS Server Health

Ensure WsusPool in IIS is running. If not, check memory settings (see Section 10.3).

In the WSUS console, run a manual synchronization to confirm server can retrieve updates from Microsoft.

Confirm that the client is not stuck in Unassigned Computers. If so, manually assign it to the proper group.

WindowsUpdate.log Review

On the client, run

```
Get-WindowsUpdateLog
```

This generates a log file on the Desktop. Search for lines referencing wsus2.d1.internal to confirm the correct server is being contacted.

10.6 Update Approval and Operational Workflow

This section outlines the standard operational process for approving and deploying updates using WSUS in our environment. These practices were developed during the proof-of-concept phase and will serve as a baseline for future WSUS operations.

WSUS Computer Groups

Client systems are organized in WSUS using server-side targeting, with each supported operating system assigned to its own WSUS computer group. For example:

- Win2016 – Windows Server 2016 systems
- (Planned) Win10, Win2019, etc. – for other OS versions

This structure allows updates to be approved either for specific operating systems or for all computers, depending on the nature of the update.

Clients are automatically added to the Unassigned Computers group when they first report in. From there, an administrator must manually move each system to the appropriate OS-specific group using the WSUS console. Group assignment is critical for update approval.

Update Approval Process

Update approvals are performed using the WSUS console. Updates can be approved for:

- Individual computer groups (e.g., Win2016)
- All Computers group (useful for critical or universally applicable updates)

We generally follow this convention:

- OS-specific updates (e.g., Windows 10 cumulative updates) are approved for the relevant group only.
- Universal updates (e.g., the Windows Malicious Software Removal Tool) may be approved for all computers.

Updates are not auto-approved. Each new batch of updates (typically released on Patch Tuesday) should be reviewed and approved manually by an administrator. This gives the team control over what gets deployed and when.

Client Update Retrieval

Once an update has been approved for the appropriate group, client systems will detect and download the update automatically based on their local Group Policy settings and Windows Update's internal schedule. In normal circumstances, this process requires no manual intervention. (During the proof-of-concept, this behavior was not directly observed, as updates were manually triggered to expedite testing).

Clients do not require domain-level Group Policy for WSUS settings. All update configuration is applied via local Group Policy, ensuring decoupling from domain controllers and simplifying troubleshooting.

Manual Overrides (if needed)

In rare cases where a client does not appear to be updating or checking in:

- A manual sync can be triggered using wuauclt /detectnow and wuauclt /reportnow

- Review the client's WindowsUpdate.log using Get-WindowsUpdateLog
- Confirm the client is in the correct WSUS group and the update has been approved

Server Synchronization

During the proof-of-concept, update synchronization with Microsoft Update was manually triggered using the WSUS console. This process retrieved metadata and binaries for the latest updates. Going forward, we recommend enabling scheduled synchronizations (e.g., daily) to ensure that WSUS2 stays current without requiring manual intervention. This can be configured under Options > Synchronization Schedule in the WSUS console.

Client Polling Interval

Windows clients that are configured via local group policy will automatically check for updates approximately every 22 hours. This schedule is randomized slightly per system to prevent simultaneous polling across the network. Once an update is approved and synchronized, no manual action is needed on the client for detection.

Staged Approval Workflow

Updates should first be approved for a small test group (e.g., a handful of Win2016 machines) and monitored for successful installation. Once verified, the same updates can then be approved for broader groups, such as all production Windows Server 2016 systems. This staged approach helps mitigate risk and isolate potential issues before widespread rollout.

11. Project Outcomes

11.1 IT Functionality Improved or Modernized

The domain migration project delivered multiple infrastructure and operational upgrades that increased manageability, security, and resilience:

- **Active Directory consolidation**: Legacy domain dtek.internal was fully retired, and all eligible systems now operate under the unified d1.internal domain with standardized Group Policy and account controls.
- **DNS structure and replication improved**: DNS zones were cleaned up, reverse lookup zones validated, and replication between domain controllers confirmed, improving name resolution reliability and administrative clarity.
- **Workstation standardization**: All domain-joined workstations now follow a consistent profile setup and permissions model. Manual transitions were verified and documented to simplify future onboarding.

- **Legacy systems preserved and stabilized**: A fragile, multi-tier application stack (involving Visual Studio, IIS, SQL Server, and TFS) was successfully migrated with no loss of functionality. Key dependencies were carefully re-established under the new domain.
- **Disaster recovery preparation introduced**: Clean Hyper-V exports of both domain controllers were created and archived, establishing a solid baseline for future DR planning.
- **Documentation culture reinforced**: Migration processes, DNS configuration, file share policies, and scheduled task management were all documented to a level suitable for junior administrators—building internal knowledge continuity.
- **Windows updates centralized and automated**: Monthly patches are now downloaded once to the WSUS server and distributed internally, reducing Internet bandwidth use and eliminating manual update checks on individual systems.

11.2 Lessons Learned and Recommendations

This project revealed several insights and improvement opportunities for future infrastructure changes:

- **System interdependencies must be mapped early**. The tight coupling of legacy systems required significant effort to trace and preserve during the transition. Documenting app-to-service dependencies should become a standard pre-migration step.
- **Reverse lookup zones are often overlooked**. Their absence can lead to subtle network issues. We recommend including zone verification in the checklist for any future DC deployment.
- **Isolated Hyper-V hosts simplify DR but require intentional exports**. Keeping Hyper-V in WORKGROUP mode worked well operationally, but it places responsibility for exports squarely on local admins. Regular DR export schedules should be institutionalized.
- **Incremental wins matter**. While this project involved legacy systems, it provided opportunities to clean up stale DNS entries, update file permissions, and align naming conventions. Even legacy migrations can result in a more modern and maintainable environment when handled deliberately.
- **Keep junior administrators in mind**. Many of the operational checklists developed during this project can now serve as reusable training material—reducing onboarding friction and institutionalizing best practices.

- **Roll out WSUS in stages**: Starting with a single test system allowed safe evaluation before affecting production servers—minimizing risk while confirming real benefits.

12. Outstanding Tasks and Future Plans

The following items were intentionally deferred or identified during the project as opportunities for future improvement, testing, or documentation.

12.1 Disaster Recovery Simulation

- A full domain recovery test has not yet been performed.
- Simulating loss of both domain controllers and testing full restore from Hyper-V exports is a critical task.
- This will help validate current backup strategy and improve confidence in disaster readiness.

12.2 File Share Permissions Audit

Although the share-level and NTFS permissions have been standardized, a complete audit of all sensitive shares should be performed:

- Check for over-permissive access
- Reconfirm Principle of Least Privilege adherence

12.3 TFS Migration to Git repo in Azure DevOps

To modernize version control and simplify access, all Team Foundation Server (TFS) projects on lunar-tfs will be migrated to Git repositories hosted in Azure DevOps. This will improve developer experience, support remote collaboration, and align with industry trends.

12.4 Final Cleanup of File Share Permissions on lunar-file

Goal: Ensure all users—regardless of domain or WORKGROUP membership—have full access to shared folders hosted on lunar-file, using the principle:

- Everyone: Full Control (Share-level)
- NTFS-level: Everyone: Full Control

This ensures simplicity and maximum accessibility for non-sensitive data.

Step 1: Confirm and Take Ownership (Including All Child Items)

1. Log in to lunar-file as a local administrator.
2. Navigate to the root of each shared folder (e.g., D:\Shared).
3. Right-click the folder → Properties → Security tab → Advanced.
4. At the top, click Change next to the Owner field.
5. In the "Select User or Group" dialog:
    - click "Locations..." and choose lunar-file to scope to the local computer.
    - Type: Administrator → click Check Names → verify it resolves to lunar-file\Administrator → click OK.
6. Back in the Advanced Security Settings window: check the box labeled "Replace owner on subcontainers and objects".
7. Click Apply. This will recursively take ownership of all child files and folders.
8. Click OK to close all dialogs.

Why this matters: If this step is skipped, subsequent NTFS permission changes may silently fail on files/folders you don't yet own—leading to inconsistent access or even total lockout when old entries are removed.

Step 2: Remove Unnecessary NTFS Permissions

1. Go back to Properties → Security tab → Advanced.
2. Click Disable inheritance.
3. When prompted, choose Convert inherited permissions into explicit permissions.
4. Carefully remove all entries except:
    - Administrators: Full Control
    - SYSTEM: Full Control
5. Now add: Everyone: Full Control

Be very cautious not to remove your current user’s access before applying the new entries.

You may wish to apply these changes one folder at a time to avoid getting locked out.

Step 3: Reset Share Permissions

1. Go to Computer Management → Shared Folders → Shares.
2. Right-click the share → Properties → Share Permissions tab.
3. Remove all entries.
4. Add: Everyone: Full Control

Step 4: Repeat for All Shared Folders

Repeat the NTFS and Share steps above for each folder shared on lunar-file.

Step 5: Confirm Effective Access

- On a separate domain-joined machine (d1.internal) and a WORKGROUP machine, attempt to access \\lunar-file\sharename.
- Try file creation, editing, and deletion to confirm full access.
- If issues occur, double-check both NTFS and share-level permissions.

12.5 Verification and Testing of webserv-test under d1.internal

Conduct end-to-end validation of webserv-test to confirm post-migration functionality:

1. Ensure the server is joined to d1.internal and rebooted.
2. Confirm SQL connectivity using Invoke-Sqlcmd or equivalent testing via SSMS.
3. Validate IIS application pools are using domain service accounts or ApplicationPoolIdentity, and that identity permissions are still valid.
4. Access hosted websites and confirm correct functionality.
5. Check Windows Firewall rules via GUI (Network and Sharing Center → Windows Defender Firewall → Advanced Settings).

12.6 Review and Removal of Legacy SQL Server Logins from dtek.internal Domain

Remove obsolete dtek.internal logins from SQL Server.

On lunar-sql, delete the following under Security > Logins:

- DTEK\Administrator
- DTEK\DEVBOX-1$
- DTEK\EU-WEB$
- DTEK\WEBSERV-TEST$

On lunar-tfs, delete DTEK\Administrator

Caution: Do not delete any login without first verifying the system's successful operation post-migration.

12.7 Pilot Upgrade of Windows 10 Virtual Machine to Windows 11

A pilot test will be performed to determine if domain-joined Windows 10 virtual machines hosted on Hyper-V can be upgraded to Windows 11.

- VM selected for pilot: w10p-3 (hosted on sirius).
- w10p-3 is a minimally configured Windows 10 workstation, joined to d1.internal.

- Objectives: Confirm feasibility, identify required configuration changes, evaluate process and stability.

Planned steps:

1. Checkpoint w10p-3 before upgrade.
2. Ensure VM uses Generation 2 with Secure Boot and TPM enabled.
3. Verify virtual hardware meets Windows 11 requirements (e.g. CPU, RAM, disk).
4. Upgrade the VM configuration version if needed.
5. Mount the Windows 11 ISO and run in-place upgrade.
6. Post-upgrade, confirm domain membership, GPO application, and system functionality.

Success criteria:

- VM remains domain-joined post-upgrade.
- No device or driver errors.
- System performs reliably with no loss of function.

This pilot will help guide whether future upgrades of additional VMs to Windows 11 are technically viable and worth pursuing.

12.8 WSUS Adoption and Future Expansion

With the WSUS proof-of-concept now validated, the following next steps will help transition the environment toward broader adoption and stable, long-term operations.

Windows 10 Pilot

- Create a dedicated WSUS computer group for Windows 10 devices.
- Select a Windows 10 test client that has not yet received the June 2025 updates.
- Apply the same local WSUS configuration as used in the proof-of-concept.
- Monitor update detection, installation, and reporting via the WSUS console.

Gradual Onboarding of Production Systems

- Begin onboarding additional Windows Server 2016 production systems using the same local policy and group assignment process.
- Monitor update behavior closely, particularly on critical infrastructure roles.
- As confidence grows, extend WSUS coverage to include all supported OS versions in the domain.

Update Organization and Staging

- Define staging computer groups for each supported OS version (e.g., Win10-Test, Win10-Prod, Win2016-Prod, etc.).
- Use these groups to stage updates before full deployment.
- Consider establishing a convention for update approvals (e.g., weekly review, monthly rollout).

Ongoing WSUS Maintenance

- Configure automatic synchronization in the WSUS console to ensure updates are regularly pulled from Microsoft Update.
- Periodically review and adjust the list of enabled product categories and classifications as organizational needs evolve.
- Re-evaluate IIS performance tuning and storage usage if the WSUS workload increases over time.

Operational Monitoring

- Confirm that newly added clients are automatically polling the WSUS server and checking in as expected.
- Use the WSUS console's reports and computer group views to track compliance and detect any client registration issues.
- Document any recurring manual steps or interventions that could be scripted or automated in future phases.

---

13. References

- **Active Directory Domain Services Overview – Introduction to AD DS**
  https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview
- **Active Directory Users and Computers – Managing user accounts**
  https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage-user-accounts-in-windows-server
- **Azure Cosmos DB Emulator – Command-line and PowerShell reference**
  https://learn.microsoft.com/en-us/azure/cosmos-db/emulator-windows-arguments
- **Azure Cosmos DB Emulator – Development and testing with the emulator**
  https://learn.microsoft.com/en-us/azure/cosmos-db/how-to-develop-emulator
- **Azure Cosmos DB Emulator – Overview and usage**
  https://learn.microsoft.com/en-us/azure/cosmos-db/emulator
- **Azure Cosmos DB Emulator – Release notes**
  https://learn.microsoft.com/en-us/azure/cosmos-db/emulator-release-notes
- **Azure Cosmos DB Emulator – Troubleshooting guide**
  https://learn.microsoft.com/en-us/troubleshoot/azure/cosmos-db/tools-connectors/emulator
- **DNS Client Configuration – Best practices for DNS client settings on domain controllers**
  https://learn.microsoft.com/en-us/troubleshoot/windows-server/networking/best-practices-for-dns-client-settings
- **DNS Manager – Managing DNS resource records**
  https://learn.microsoft.com/en-us/windows-server/networking/dns/manage-resource-records
- **Dynamic DNS Updates – How DNS dynamic updates work**
  https://learn.microsoft.com/en-us/windows-server/networking/dns/dynamic-update
- **Generation 2 virtual machine security settings for Hyper-V**
  https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/learn-more/generation-2-virtual-machine-security-settings-for-hyper-v
- **Hyper-V TPM config**
  https://techcommunity.microsoft.com/blog/itopstalkblog/how-to-run-a-windows-11-vm-on-hyper-v/3713948

- **IIS Configuration – Configure application pool identity**
  https://learn.microsoft.com/en-us/iis/manage/configuring-security/application-pool-identities
- **Microsoft TPM/CPU requirements for Windows 11**
  https://learn.microsoft.com/en-us/windows/whats-new/windows-11-requirements
- **PowerShell – DNS Zone Listing – Get-DnsServerZone**
  https://learn.microsoft.com/en-us/powershell/module/dnsserver/get-dnsserverzone
- **PowerShell – DNS Zone Management – Add-DnsServerPrimaryZone**
  https://learn.microsoft.com/en-us/powershell/module/dnsserver/add-dnsserverprimaryzone
- **PowerShell – Invoke-Sqlcmd Cmdlet – Invoke-Sqlcmd**
  https://learn.microsoft.com/en-us/powershell/module/sqlserver/invoke-sqlcmd
- **PowerShell – SQL Server Cmdlets – SQLServer module overview**
  https://learn.microsoft.com/en-us/powershell/module/sqlserver/
- **PowerShell – SQL Server Module Installation – Installing the SqlServer module**
  https://learn.microsoft.com/en-us/sql/powershell/download-sql-server-ps-module
- **Project-Specific Documentation – README for CosmosDB Emulator Deployment**
  https://rob-das-win.azurewebsites.net/html/README.html#development-database-azure-cosmos-db-emulator
- **System Properties (sysdm.cpl) – Accessing system settings**
  https://learn.microsoft.com/en-us/windows/win32/shell/executing-control-panel-items
- **TFS Migration – Migrating from TFS to Azure DevOps Services**
  https://learn.microsoft.com/en-us/azure/devops/migrate/migration-overview
- **Task Scheduler – Overview and usage**
  https://learn.microsoft.com/en-us/windows/win32/taskschd/task-scheduler-start-page
- **Windows Server Manager – Overview and usage**
  https://learn.microsoft.com/en-us/windows-server/administration/server-manager/server-manager
- **Windows Server Update Services (WSUS)**
  https://learn.microsoft.com/en-us/windows/deployment/update/waas-manage-updates-wsus

- **WSUS and Configuration Manager**
  https://learn.microsoft.com/en-us/troubleshoot/mem/configmgr/update-management/wsus-maintenance-guide