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 a critical failure in the company's legacy Active Directory domain infrastructure, which had evolved over many years without formal management. 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, 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, 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, *domain1.local*, 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, *domain2.local*, from the ground up, with clean Active Directory configuration, hardened DNS, and improved operational practices. A pair of domain controllers (*domain2-dc1* and *domain2-dc2*) 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.

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 domain1.local to domain2.local.

3.1 Active Directory and Domain Controllers

The new domain2.local 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

- domain2-dc1: Primary domain controller hosted on Hyper-V server hyp4.
- domain2-dc2: Secondary domain controller hosted on Hyper-V server hyp3.

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:

- hyp1 (Windows Server 2016, retained due to long-standing stability)
- hyp2 (Windows Server 2019)
- hyp3 (Windows Server 2019)
- hyp4 (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, hyp2 served both as a Hyper-V host and as a file server. A new virtual machine named hyp2-file, running Windows Server 2016, was created to assume all file server responsibilities previously handled by hyp2.

- hyp2-file is now the second file server in the organization.
- It is hosted on hyp2 and uses a dedicated VHDX volume backed by local storage.
- File shares were migrated from hyp2's physical F: drive to hyp2-file, 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 hyp1-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 hyp2-file.

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 domain2.local
- 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 domain1.local 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 domain1.local 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:

- hyp1-sql: A legacy SQL Server 2014 system that hosts production databases used by multiple internal applications. It was migrated from the domain1.local domain to domain2.local as part of the project.
- hyp3-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.

4. Domain Architecture

This section describes the Active Directory domain architecture.

4.1 Purpose and design of domain2-dc1 and domain2-dc2

The domain domain2.local is hosted by two domain controllers, both running Windows Server 2019 edition:

- domain2-dc2 is hosted on hyp3
- domain2-dc1 is hosted on hyp4

All four Hyper-V hosts in the environment (hyp1, hyp2, hyp3, hyp4) 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 domain2-dc1 and domain2-dc2. On 2025-05-07, both domain controller VMs were shut down and exported to form a backup named:

```
domain2-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 domain1.local.

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 (domain2-dc1 and domain2-dc2).

- Primary zone: domain2.local
- 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 (domain2-dc1 and domain2-dc2) 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: domain2-dc1.domain2.local
- Domain Naming Master: domain2-dc1.domain2.local
- PDC Emulator: domain2-dc1.domain2.local
- RID Master: domain2-dc1.domain2.local
- Infrastructure Master: domain2-dc1.domain2.local

All FSMO roles are currently held by domain2-dc1.

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, hyp2 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 hyp2-file. This system was created specifically to offload user and departmental file shares from hyp2, allowing hyp2 to operate solely as a Hyper-V hypervisor going forward.

In parallel, substantial cleanup was performed on hyp1-file, an older file server, to consolidate, remove redundant shares, and improve organizational clarity. Although more work remains—particularly in aligning hyp1-file share permissions with the least-privilege model established on hyp2-file—this effort marked a key step toward standardized and streamlined file service management.

Both hyp2-file and hyp1-file are Windows Server systems joined to the domain2.local 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:

- hyp2 was reconfigured to remove file services and act solely as a Hyper-V host.
- hyp1 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.
- hyp4 and hyp3 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 domain2.local. Where possible, local administrator access is restricted to domain-level admin groups (DOMAIN2\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 hyp2-file and hyp1-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 domain2.local domain, placed in a WORKGROUP, or in special cases left unjoined due to technical constraints.

Domain / Workgroup	Role	Number of Computers
domain2.local	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
WORKGROUP	Database Server (Cosmos DB)	1
	Decommissioned DC	1
	Developer Workstation	2
	General Purpose Server	3
	General Purpose Workstation	6
	Hyper-V Host	4

Domain / Workgroup	Role	Number of Computers
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 domain2.local 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 (hyp4, hyp3, hyp1, hyp2) 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 hyp3-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:

- hyp1-sql and hyp1-tfs retain legacy TFS + SQL Server roles with domain-linked authentication still being cleaned up.
- The web servers (hyp2-web1, hyp2-web2) required careful DNS and authentication coordination with SQL-based services on hyp1-sql using Windows-integrated credentials.

- A few legacy computers had dual identities (e.g., previously joined to domain1.local, then rejoined to domain2.local) 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 Users and Computers (ADUC)
- Server Manager
- Hyper-V Manager
- PowerShell and Command Prompt
- Remote Desktop Connection
- 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 troubleshooting guidance, PowerShell scripting help, and strategic review suggestions
- DNS Manager

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 (domain2.local) 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:
    - domain2-dc1
    - domain2-dc2
- To verify reverse lookup zones exist:

```
Get-DnsServerZone -ComputerName domain2-dc1 | Where-Object { $_.IsReverseLookupZone }
```

Domain Join and Workstation Profile Transition

- Backup local user data (Desktop/Documents/Downloads).
- Join workstation to domain2.local domain.
- Log in once with domain user to create profile.
- Reconfigure network drives and printers manually.

Scheduled Task Migration

Each task scheduled on hyp1-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:

- Hyp3-dev1: Developer workstation for legacy Visual Studio projects
- hyp2-web1: Production front-end web server (MVC application)
- hyp2-web2: Staging/test web server (mirrors production setup)

- hyp1-sql: Database server hosting Webserv-dev, Webserv-test, and Webserv-prod (SQL Server 2014)
- hyp1-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 domain1.local to domain2.local 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 domain2.local, which required:

- Adding new SQL Server logins for the domain-joined machine accounts: DOMAIN2\HYP2-WEB1$, DOMAIN2\HYP2-WEB2$, and DOMAIN2\HYP3-DEV1$
- 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 hyp1-tfs server hosted a legacy TFS environment backed by its own SQL Server instance. The migration involved:

- Domain join to domain2.local 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 hyp2-web1.

10. Project Outcomes

10.1 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 domain1.local was fully retired, and all eligible systems now operate under the unified domain2.local 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.

10.2 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.

11. Outstanding Tasks and Future Plans

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

11.1 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.

11.2 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

11.3 TFS Migration to Git repo in Azure DevOps

To modernize version control and simplify access, all Team Foundation Server (TFS) projects on hyp1-tfs will be migrated to Git repositories hosted in Azure DevOps. This will improve developer experience, support remote collaboration, and align with industry trends.

11.4 Final Cleanup of File Share Permissions on hyp1-file

Goal: Ensure all users—regardless of domain or WORKGROUP membership—have full access to shared folders hosted on hyp1-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 hyp1-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 hyp1-file to scope to the local computer.
    - Type: Administrator → click Check Names → verify it resolves to hyp1-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 hyp1-file.

Step 5: Confirm Effective Access

- On a separate domain-joined machine (domain2.local) and a WORKGROUP machine, attempt to access \\hyp1-file\sharename.
- Try file creation, editing, and deletion to confirm full access.
- If issues occur, double-check both NTFS and share-level permissions.

11.5 Verification and Testing of hyp2-web2 under domain2.local

Conduct end-to-end validation of hyp2-web2 to confirm post-migration functionality:

1. Ensure the server is joined to domain2.local 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).

11.6 Review and Removal of Legacy SQL Server Logins from domain1.local Domain

Remove obsolete domain1.local logins from SQL Server.

On hyp1-sql, delete the following under Security > Logins:

- DOMAIN1\Administrator
- DOMAIN1\HYP3-DEV1$
- DOMAIN1\HYP2-WEB1$
- DOMAIN1\HYP2-WEB2$

On hyp1-tfs, delete DOMAIN1\Administrator

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

---

12. References (Alphabetized)

1. **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
2. **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
3. **Azure Cosmos DB Emulator – Command-line and PowerShell reference**
   https://learn.microsoft.com/en-us/azure/cosmos-db/emulator-windows-arguments
4. **Azure Cosmos DB Emulator – Development and testing with the emulator**
   https://learn.microsoft.com/en-us/azure/cosmos-db/how-to-develop-emulator
5. **Azure Cosmos DB Emulator – Overview and usage**
   https://learn.microsoft.com/en-us/azure/cosmos-db/emulator
6. **Azure Cosmos DB Emulator – Release notes**
   https://learn.microsoft.com/en-us/azure/cosmos-db/emulator-release-notes

7. **Azure Cosmos DB Emulator – Troubleshooting guide**
   https://learn.microsoft.com/en-us/troubleshoot/azure/cosmos-db/tools-connectors/emulator
8. **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
9. **DNS Manager – Managing DNS resource records**
   https://learn.microsoft.com/en-us/windows-server/networking/dns/manage-resource-records
10. **Dynamic DNS Updates – How DNS dynamic updates work**
    https://learn.microsoft.com/en-us/windows-server/networking/dns/dynamic-update
11. **IIS Configuration – Configure application pool identity**
    https://learn.microsoft.com/en-us/iis/manage/configuring-security/application-pool-identities
12. **PowerShell – DNS Zone Listing – Get-DnsServerZone**
    https://learn.microsoft.com/en-us/powershell/module/dnsserver/get-dnsserverzone
13. **PowerShell – DNS Zone Management – Add-DnsServerPrimaryZone**
    https://learn.microsoft.com/en-us/powershell/module/dnsserver/add-dnsserverprimaryzone
14. **PowerShell – Invoke-Sqlcmd Cmdlet – Invoke-Sqlcmd**
    https://learn.microsoft.com/en-us/powershell/module/sqlserver/invoke-sqlcmd
15. **PowerShell – SQL Server Cmdlets – SQLServer module overview**
    https://learn.microsoft.com/en-us/powershell/module/sqlserver/
16. **PowerShell – SQL Server Module Installation – Installing the SqlServer module**
    https://learn.microsoft.com/en-us/sql/powershell/download-sql-server-ps-module
17. **Project-Specific Documentation – README for CosmosDB Emulator Deployment**
    https://rob-das-win.azurewebsites.net/html/README.html#development-database-azure-cosmos-db-emulator
18. **System Properties (sysdm.cpl) – Accessing system settings**
    https://learn.microsoft.com/en-us/windows/win32/shell/executing-control-panel-items

19. **TFS Migration – Migrating from TFS to Azure DevOps Services**
    https://learn.microsoft.com/en-us/azure/devops/migrate/migration-overview

20. **Task Scheduler – Overview and usage**
    https://learn.microsoft.com/en-us/windows/win32/taskschd/task-scheduler-start-page

21. **Windows Server Manager – Overview and usage**
    https://learn.microsoft.com/en-us/windows-server/administration/server-manager/server-manager