**Citrix PVS Import Script - README**

**Overview**

This PowerShell script imports a previously exported Citrix Provisioning Services (PVS) configuration from XML files back into a PVS environment. It recreates the entire configuration including farms, sites, collections, devices, vDisks, and more, all with detailed logging throughout the process.

**Features**

* Imports all PVS configuration areas from XML files
* Processes site-specific configuration from individual site directories
* Uses DiskLocatorId or DiskLocatorName parameters when importing vDisks
* Handles existing objects gracefully (skips creation if objects already exist)
* Provides detailed logging with timestamps and severity levels
* Comprehensive error handling with detailed error reporting

**Requirements**

* Citrix Provisioning Services installed (either console or server components)
* PowerShell 3.0 or higher
* Administrative rights on the PVS server
* Citrix.PVS.SnapIn must be available
* A previously created export directory from the Export-PVSConfiguration.ps1 script

**Usage**

1. Save the script as Import-PVSConfiguration.ps1
2. Open PowerShell as Administrator
3. Navigate to the directory containing the script
4. Execute the script with the path to your export directory:

powershell

.\Import-PVSConfiguration.ps1 -ImportPath "C:\Path\To\PVSExport\_YourExportFolder"

**Parameters**

* ImportPath (Required): Full path to the directory containing the exported PVS configuration files

**Import Process**

The script follows a specific order during import to ensure proper dependency handling:

1. Farm configuration (if it doesn't already exist)
2. Farm properties
3. For each site:
   * Site configuration (if it doesn't already exist)
   * Site properties
   * Stores
   * Servers
   * Collections
   * Auth groups
   * vDisks
   * Devices (with vDisk assignments)

**Important Notes**

1. This script only imports configuration data, not the actual vDisk image files. These should be restored separately using standard file system restore procedures.
2. The vDisk files must be accessible in the store paths defined in the PVS configuration.
3. The script must be run on a machine with the PVS console or server components installed.
4. The user running the script must have administrative permissions in the PVS environment.
5. If objects already exist in the target environment, they will be skipped with a warning message.

**Logs**

The script creates a detailed log file in the import directory named PVSImport\_[date].log. This log file contains information about:

* Import operations
* Objects created during import
* Objects skipped (already existing)
* Any errors or warnings encountered during the import process

**Example**

powershell

PS C:\Scripts> .\Import-PVSConfiguration.ps1 -ImportPath "C:\Backups\PVSExport\_20250414\_101522"

[2025-04-14 14:30:22] [INFO] PVS Import started. Import directory: C:\Backups\PVSExport\_20250414\_101522

[2025-04-14 14:30:23] [INFO] Added Citrix.PVS.SnapIn

[2025-04-14 14:30:24] [INFO] Successfully connected to PVS

[2025-04-14 14:30:25] [INFO] Created farm: MyPVSFarm

[2025-04-14 14:30:26] [INFO] Imported farm properties

[2025-04-14 14:30:27] [INFO] Processing site: Site1

...

[2025-04-14 14:35:45] [INFO] PVS Import completed successfully

**Troubleshooting**

* If the script fails with a "SnapIn not registered" error, ensure the PVS console is installed properly.
* If you receive access denied errors, ensure you're running PowerShell as Administrator and have proper PVS permissions.
* If objects cannot be created, ensure that the necessary permissions and prerequisites are in place.
* Check the log file for detailed error information if the script fails.
* If specific objects are skipped, check if they already exist in the target environment.

**Limitations**

* The script cannot restore custom settings that aren't exported by the export script.
* Some complex relationships between objects may need manual verification after import.