==========

==========

CATEGORY: WindowsStorage

==========

==========

Storage operations directly affect data integrity, system availability, provisioning workflows, and disaster-recovery readiness. These SOPs ensure every storage-related action performed through RDAM Script Wizard is **controlled**, **auditable**, and aligned with enterprise operational and security standards.

SOP 1 – Get Disk Information

Script Name: Get Disk Information **Category:** WindowsStorage **Version:** 1.0 **Approved By:** IT Operations / Engineering

1. Purpose

This script retrieves detailed information about physical disks, supporting diagnostics, provisioning, and capacity planning.

2. Scope

- Windows servers and workstations
- Physical and virtual disks

3. Definitions

- **Disk:** A physical storage device.
- **Partition Style:** MBR or GPT.

4. Preconditions

- Operator must have permission to query disk configuration.

5. Required Inputs

- Optional: Disk number filter

6. Procedure Steps

1. Input Collection
    - Wizard prompts for optional disk filter.
2. Disk Enumeration
    - Retrieve all disks.
3. Attribute Retrieval
    - Extract:
        - Disk number
        - Size
        - Partition style
        - Health status
        - Bus type
        - Operational status
4. Output Formatting
    - Present structured disk list.
5. Logging
    - Log filter, operator, timestamp.

7. Expected Output

- Detailed disk information.

8. Post-Execution Validation

- Operator may verify via `Get-Disk`.

9. Error Handling

- Access denied
- Invalid filter
- Disk not found

10. Security Considerations

- Disk metadata may reveal sensitive infrastructure details.

11. Audit Logging Requirements

- Operator ID
- Filter used
- Timestamp

12. Organizational Benefit Statement

This script provides a consistent, auditable method for retrieving disk information, supporting provisioning and diagnostics.

SOP 2 – Get Volume Information

Script Name: Get Volume Information **Category:** WindowsStorage**

1. Purpose

This script retrieves detailed information about volumes, supporting troubleshooting, capacity planning, and configuration validation.

2. Scope

- Windows servers and workstations
- All mounted volumes

3. Definitions

- **Volume:** A logical storage unit formatted with a file system.

4. Preconditions

- Operator must have permission to query volume configuration.

5. Required Inputs

- Optional: Drive letter or volume label

6. Procedure Steps

1. Input Collection
    - Wizard prompts for optional filter.
2. Volume Enumeration
    - Retrieve all volumes.
3. Attribute Retrieval

- Extract:
    - Drive letter
    - File system
    - Size and free space
    - Health status
    - BitLocker status

4. Output Formatting
    - Present structured volume list.
5. Logging
    - Log filter, operator, timestamp.

7. Expected Output

- Detailed volume information.

8. Post-Execution Validation

- Operator may verify via `Get-Volume`.

9. Error Handling

- Access denied
- Invalid filter
- Volume not found

10. Security Considerations

- Volume metadata may reveal sensitive storage layout.

11. Audit Logging Requirements

- Operator ID
- Filter used
- Timestamp

12. Organizational Benefit Statement

This script provides a reliable, auditable method for retrieving volume information, supporting troubleshooting and capacity planning.

SOP 3 – Initialize Disk

Script Name: Initialize Disk **Category:** WindowsStorage**

1. Purpose

This script initializes a disk with a partition style, supporting provisioning and deployment workflows.

2. Scope

- Windows servers and workstations
- Uninitialized disks

3. Definitions

- **Initialization:** Preparing a disk with GPT or MBR.

4. Preconditions

- Operator must have administrative rights.
- Disk must be uninitialized.
- Action must be authorized.

5. Required Inputs

- Disk number
- Partition style (GPT/MBR)

6. Procedure Steps

1. Input Collection
    - Wizard prompts for disk number and partition style.
2. Validation
    - Confirm disk exists and is uninitialized.
3. Initialization Operation
    - Apply selected partition style.
4. Post-Initialization Verification
    - Confirm disk is initialized.
5. Logging
    - Log disk number, partition style, operator, timestamp.

7. Expected Output

- Confirmation of disk initialization.

8. Post-Execution Validation

- Operator may verify via `Get-Disk`.

9. Error Handling

- Disk already initialized
- Access denied
- Invalid disk number

10. Security Considerations

- Initialization destroys existing partition data.

11. Audit Logging Requirements

- Operator ID
- Disk number
- Partition style
- Timestamp

12. Organizational Benefit Statement

This script ensures disk initialization is performed safely and consistently, supporting provisioning workflows.

SOP 4 – Create Partition

Script Name: Create Partition **Category:** WindowsStorage**

1. Purpose

This script creates a new partition on a disk, supporting provisioning, deployment, and storage expansion.

2. Scope

- Windows servers and workstations
- GPT and MBR disks

3. Definitions

- **Partition:** A defined region of a disk.

4. Preconditions

- Operator must have administrative rights.
- Disk must have unallocated space.

5. Required Inputs

- Disk number
- Partition size
- Optional: Drive letter

6. Procedure Steps

1. Input Collection
    - Wizard prompts for disk number, size, and optional drive letter.
2. Validation
    - Confirm disk exists.
    - Confirm sufficient unallocated space.
3. Partition Creation
    - Create new partition.
4. Post-Creation Verification
    - Confirm partition exists.
5. Logging
    - Log disk number, size, operator, timestamp.

7. Expected Output

- Confirmation of partition creation.

8. Post-Execution Validation

- Operator may verify via `Get-Partition`.

9. Error Handling

- Insufficient space

- Access denied
- Invalid disk number

10. Security Considerations

- Incorrect partitioning may impact system boot or data layout.

11. Audit Logging Requirements

- Operator ID
- Disk number
- Partition size
- Timestamp

12. Organizational Benefit Statement

This script ensures partition creation is performed safely and consistently, supporting provisioning and storage expansion.

SOP 5 – Format Volume

Script Name: Format Volume **Category:** WindowsStorage**

1. Purpose

This script formats a volume with a specified file system, supporting provisioning, cleanup, and re-deployment.

2. Scope

- Windows servers and workstations
- NTFS, ReFS, FAT32

3. Definitions

- **Format:** Preparing a volume with a file system.

4. Preconditions

- Operator must have administrative rights.
- Formatting must be authorized.
- Volume must not contain required data.

5. Required Inputs

- Drive letter or volume ID
- File system type
- Optional: Volume label

6. Procedure Steps

1. Input Collection
    - Wizard prompts for volume and file system.
2. Validation
    - Confirm volume exists.
    - Confirm file system type is valid.
3. Format Operation
    - Format volume with selected file system.
4. Post-Format Verification
    - Confirm volume formatted successfully.
5. Logging
    - Log volume, file system, operator, timestamp.

7. Expected Output

- Confirmation of volume formatting.

8. Post-Execution Validation

- Operator may verify via `Get-Volume`.

9. Error Handling

- Access denied
- Invalid file system
- Volume not found

10. Security Considerations

- Formatting permanently deletes data.

11. Audit Logging Requirements

- Operator ID
- Volume
- File system
- Timestamp

12. Organizational Benefit Statement

This script ensures formatting is performed safely and consistently, supporting provisioning and cleanup workflows.

SOP 6 – Assign Drive Letter

Script Name: Assign Drive Letter **Category:** WindowsStorage**

1. Purpose

This script assigns or changes a drive letter for a volume, supporting provisioning, application requirements, and storage organization.

2. Scope

- Windows servers and workstations
- All mounted volumes

3. Definitions

- **Drive Letter:** Identifier used to access a volume.

4. Preconditions

- Operator must have administrative rights.
- Drive letter must be available.

5. Required Inputs

- Volume ID or drive letter
- New drive letter

6. Procedure Steps

1. Input Collection
    - Wizard prompts for volume and new letter.

2. Validation
    - Confirm volume exists.
    - Confirm letter is available.
3. Assignment Operation
    - Apply new drive letter.
4. Post-Assignment Verification
    - Confirm volume reflects new letter.
5. Logging
    - Log old/new letters, operator, timestamp.

7. Expected Output

- Confirmation of drive-letter assignment.

8. Post-Execution Validation

- Operator may verify via File Explorer or PowerShell.

9. Error Handling

- Letter already in use
- Access denied
- Volume not found

10. Security Considerations

- Changing drive letters may break application paths.

11. Audit Logging Requirements

- Operator ID
- Old/new letters
- Timestamp

12. Organizational Benefit Statement

This script ensures drive-letter assignments are performed safely and consistently, supporting storage organization and application requirements.

SOP 7 – Get SMART Status

Script Name: Get SMART Status **Category:** WindowsStorage**

1. Purpose

This script retrieves SMART (Self-Monitoring, Analysis, and Reporting Technology) data, supporting hardware-health monitoring and predictive failure analysis.

2. Scope

- Windows servers and workstations
- Physical disks supporting SMART

3. Definitions

- **SMART:** Technology that reports disk health metrics.

4. Preconditions

- Operator must have permission to query hardware health.

5. Required Inputs

- Optional: Disk number

6. Procedure Steps

1. Input Collection
    - Wizard prompts for optional disk filter.
2. SMART Retrieval
    - Query SMART attributes.
    - Extract:
        - Health status
        - Temperature
        - Reallocated sectors
        - Pending sectors
        - Read/write error rates
3. Output Formatting
    - Present structured SMART summary.

4. Logging
    - Log filter, operator, timestamp.

7. Expected Output

- SMART health report.

8. Post-Execution Validation

- Operator may verify via vendor tools.

9. Error Handling

- SMART unsupported
- Access denied
- Disk not found

10. Security Considerations

- SMART data may reveal hardware vulnerabilities.

11. Audit Logging Requirements

- Operator ID
- Filter used
- Timestamp

12. Organizational Benefit Statement

This script provides a controlled, auditable method for retrieving SMART data, supporting predictive maintenance and hardware lifecycle planning.