1. Retrieve user rights assignments.
2. Data Extraction
    - Extract:
        - Password policy
        - Lockout policy
        - Audit settings
        - Privilege assignments
3. Output Formatting
    - Present structured security policy summary.
4. Logging
    - Log operator and timestamp.

7. Expected Output

- Comprehensive local security policy overview.

8. Post-Execution Validation

- Operator may verify using `secpol.msc`.

9. Error Handling

- Access denied
- Policy retrieval failure

10. Security Considerations

- Security policy data is sensitive; restrict access.

11. Audit Logging Requirements

- Operator ID
- Timestamp

12. Organizational Benefit Statement

This script provides a consistent, auditable method for retrieving local security policy settings, supporting compliance and hardening.

SOP 2 – Check Windows Defender Status

Script Name: Check Windows Defender Status **Category:** Security

1. Purpose

This script retrieves the operational status of Microsoft Defender Antivirus, supporting security monitoring, compliance checks, and incident response.

2. Scope

- Windows servers and workstations
- Microsoft Defender Antivirus

3. Definitions

- **Real-Time Protection:** Continuous malware monitoring.
- **Engine Version:** Version of Defender scanning engine.

4. Preconditions

- Operator must have permission to query Defender status.

5. Required Inputs

- None

6. Procedure Steps

1. Status Query
    - Retrieve Defender operational state.
    - Retrieve real-time protection status.
    - Retrieve engine and signature versions.
2. Output Formatting
    - Present structured Defender status summary.
3. Logging
    - Log operator and timestamp.

7. Expected Output

- Defender health and configuration status.

8. Post-Execution Validation

- Operator may verify via Windows Security Center.

9. Error Handling

- Defender disabled
- Access denied
- WMI or API failure

10. Security Considerations

- Defender status reveals security posture; restrict access.

11. Audit Logging Requirements

- Operator ID
- Timestamp

12. Organizational Benefit Statement

This script provides a reliable, auditable method for validating Defender status, supporting security monitoring and compliance.

SOP 3 – Run Windows Defender Quick Scan

Script Name: Run Windows Defender Quick Scan **Category:** Security

1. Purpose

This script initiates a Microsoft Defender quick scan, supporting rapid malware detection and incident response.

2. Scope

- Windows servers and workstations
- Microsoft Defender Antivirus

3. Definitions

- **Quick Scan:** Scans common malware locations.

4. Preconditions

- Defender must be installed and enabled.

- Operator must have administrative rights.

5. Required Inputs

- None

6. Procedure Steps

1. Scan Initialization
    - Trigger Defender quick scan.
2. Monitoring
    - Capture scan start and completion status.
3. Output Formatting
    - Present scan results summary.
4. Logging
    - Log operator and timestamp.

7. Expected Output

- Confirmation of scan completion and findings.

8. Post-Execution Validation

- Operator may verify via Windows Security Center.

9. Error Handling

- Defender disabled
- Access denied
- Scan engine failure

10. Security Considerations

- Scan results may reveal sensitive system activity.

11. Audit Logging Requirements

- Operator ID
- Scan type
- Timestamp

12. Organizational Benefit Statement

This script provides a fast, auditable method for detecting malware, supporting incident response and security monitoring.

SOP 4 – Run Windows Defender Full Scan

Script Name: Run Windows Defender Full Scan **Category:** Security

1. Purpose

This script initiates a full system scan using Microsoft Defender, supporting deep malware detection and forensic investigation.

2. Scope

- Windows servers and workstations
- Microsoft Defender Antivirus

3. Definitions

- **Full Scan:** Scans all files, drives, and processes.

4. Preconditions

- Defender must be installed and enabled.
- Operator must have administrative rights.
- Sufficient time and resources must be available.

5. Required Inputs

- None

6. Procedure Steps

1. Scan Initialization
    - Trigger Defender full scan.
2. Monitoring
    - Track scan progress and completion.
3. Output Formatting
    - Present scan results summary.
4. Logging

- Log operator and timestamp.

7. Expected Output

- Confirmation of full scan completion and findings.

8. Post-Execution Validation

- Operator may verify via Windows Security Center.

9. Error Handling

- Defender disabled
- Access denied
- Scan engine failure

10. Security Considerations

- Full scans may impact system performance.
- Scan results may contain sensitive data.

11. Audit Logging Requirements

- Operator ID
- Scan type
- Timestamp

12. Organizational Benefit Statement

This script provides a thorough, auditable method for detecting malware, supporting deep investigations and security assurance.

SOP 5 – Get Installed Hotfixes / Security Updates

Script Name: Get Installed Hotfixes / Security Updates **Category:** Security

1. Purpose

This script retrieves installed Windows hotfixes and security updates, supporting patch compliance, vulnerability management, and forensic analysis.

2. Scope

- Windows servers and workstations
- Windows Update, WSUS, and manual patching

3. Definitions

- **Hotfix:** A specific update addressing a bug or vulnerability.
- **KB Number:** Knowledge Base identifier for an update.

4. Preconditions

- Operator must have permission to query update history.

5. Required Inputs

- Optional: KB filter
- Optional: Date range

6. Procedure Steps

1. Input Collection
    - Wizard prompts for optional filters.
2. Update Enumeration
    - Retrieve installed updates.
    - Apply filters if provided.
3. Attribute Retrieval
    - Extract:
        - KB number
        - Installation date
        - Description
        - Source (Windows Update, WSUS, manual)
4. Output Formatting
    - Present structured update list.
5. Logging
    - Log filters, operator, timestamp.

7. Expected Output

- List of installed hotfixes and security updates.

8. Post-Execution Validation

- Operator may verify via `wmic qfe` or Windows Update history.

9. Error Handling

- Access denied
- Update history unavailable
- Invalid filter

10. Security Considerations

- Patch data may reveal vulnerability exposure.

11. Audit Logging Requirements

- Operator ID
- Filters used
- Timestamp

12. Organizational Benefit Statement

This script provides a controlled, auditable method for retrieving patch history, supporting compliance and vulnerability management.