

=====

=====

CATEGORY: WindowsDefenderAdvanced

=====

=====

Advanced Microsoft Defender operations directly affect system security posture, malware remediation, compliance, and incident response. These SOPs ensure every Defender-related action performed through RDAM Script Wizard is **controlled, auditable**, and aligned with enterprise security standards.

SOP 1 – Get Defender Exclusions

Script Name: Get Defender Exclusions **Category:** WindowsDefenderAdvanced **Version:** 1.0
Approved By: Security Operations / IT Leadership

1. Purpose

This script retrieves all configured Microsoft Defender exclusions, supporting security reviews, compliance checks, and troubleshooting.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Exclusion:** A file, folder, process, or extension ignored by Defender.

4. Preconditions

- Operator must have permission to query Defender configuration.

5. Required Inputs

- None

6. Procedure Steps

1. Retrieve Exclusion Lists

- File exclusions
 - Folder exclusions
 - Process exclusions
 - Extension exclusions
2. Output Formatting
 - Present structured exclusion summary.
 3. Logging
 - Log operator and timestamp.

7. Expected Output

- List of all Defender exclusions.

8. Post-Execution Validation

- Operator may verify via Windows Security Center or PowerShell.

9. Error Handling

- Access denied
- Defender service unavailable

10. Security Considerations

- Exclusions may expose security gaps; restrict access.

11. Audit Logging Requirements

- Operator ID
- Timestamp

12. Organizational Benefit Statement

This script provides a controlled, auditable method for reviewing Defender exclusions, supporting compliance and security hardening.

SOP 2 – Add Defender Exclusion

Script Name: Add Defender Exclusion **Category:** WindowsDefenderAdvanced**

1. Purpose

This script adds a new Defender exclusion, supporting application compatibility, performance tuning, and troubleshooting.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Exclusion Type:** File, folder, process, or extension.

4. Preconditions

- Operator must have administrative rights.
- Exclusion must be authorized by security policy.

5. Required Inputs

- Exclusion type
- Exclusion value

6. Procedure Steps

1. Input Collection
 - Wizard prompts for exclusion type and value.
2. Validation
 - Confirm exclusion type is valid.
 - Confirm path or process exists (if applicable).
3. Add Operation
 - Apply exclusion to Defender configuration.
4. Post-Add Verification
 - Confirm exclusion appears in list.
5. Logging
 - Log exclusion type/value, operator, timestamp.

7. Expected Output

- Confirmation of exclusion creation.

8. Post-Execution Validation

- Operator may verify via PowerShell.

9. Error Handling

- Invalid exclusion type
- Access denied
- Path not found

10. Security Considerations

- Exclusions reduce protection; ensure strict approval.

11. Audit Logging Requirements

- Operator ID
- Exclusion type/value
- Timestamp

12. Organizational Benefit Statement

This script ensures exclusion creation is performed safely and consistently, supporting compatibility while maintaining auditability.

SOP 3 – Remove Defender Exclusion

Script Name: Remove Defender Exclusion **Category:** WindowsDefenderAdvanced**

1. Purpose

This script removes an existing Defender exclusion, supporting security hardening, cleanup, and compliance.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Exclusion Removal:** Deleting an exclusion entry.

4. Preconditions

- Operator must have administrative rights.
- Exclusion must exist.

5. Required Inputs

- Exclusion type
- Exclusion value

6. Procedure Steps

1. Input Collection
 - Wizard prompts for exclusion type and value.
2. Validation
 - Confirm exclusion exists.
3. Removal Operation
 - Remove exclusion from configuration.
4. Post-Removal Verification
 - Confirm exclusion no longer appears.
5. Logging
 - Log exclusion type/value, operator, timestamp.

7. Expected Output

- Confirmation of exclusion removal.

8. Post-Execution Validation

- Operator may verify via PowerShell.

9. Error Handling

- Exclusion not found
- Access denied

10. Security Considerations

- Removing exclusions may impact application behavior.

11. Audit Logging Requirements

- Operator ID
- Exclusion type/value
- Timestamp

12. Organizational Benefit Statement

This script ensures exclusion removal is performed safely and with full accountability, supporting security hardening and compliance.

SOP 4 – Get Defender Threat History

Script Name: Get Defender Threat History **Category:** WindowsDefenderAdvanced**

1. Purpose

This script retrieves Defender threat detections and remediation history, supporting incident response, forensic analysis, and compliance.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Threat History:** Log of malware detections and actions taken.

4. Preconditions

- Operator must have permission to query threat history.

5. Required Inputs

- Optional: Date range
- Optional: Threat name filter

6. Procedure Steps

1. Input Collection

- Wizard prompts for optional filters.

2. History Retrieval

- Retrieve threat events.
- Apply filters if provided.

3. Attribute Extraction

- Threat name
- Severity
- Action taken
- Timestamp
- File path

4. Output Formatting

- Present structured threat history.

5. Logging

- Log filters, operator, timestamp.

7. Expected Output

- List of threat detections and remediation actions.

8. Post-Execution Validation

- Operator may verify via Windows Security Center.

9. Error Handling

- Access denied
- History unavailable

10. Security Considerations

- Threat data may contain sensitive file paths.

11. Audit Logging Requirements

- Operator ID
- Filters used
- Timestamp

12. Organizational Benefit Statement

This script provides a controlled, auditable method for retrieving threat history, supporting incident response and compliance.

SOP 5 – Restore Quarantined Item

Script Name: Restore Quarantined Item **Category:** WindowsDefenderAdvanced**

1. Purpose

This script restores a quarantined file, supporting troubleshooting, false-positive remediation, and application recovery.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Quarantine:** Isolated storage for suspected malware.

4. Preconditions

- Operator must have administrative rights.
- Restoration must be authorized by security.

5. Required Inputs

- Threat ID or quarantined item ID

6. Procedure Steps

1. Input Collection
 - Wizard prompts for item ID.
2. Validation
 - Confirm item exists in quarantine.
3. Restoration Operation
 - Restore quarantined file to original location.
4. Post-Restore Verification
 - Confirm file restored successfully.

5. Logging

- Log item ID, operator, timestamp.

7. Expected Output

- Confirmation of file restoration.

8. Post-Execution Validation

- Operator may verify via Defender history.

9. Error Handling

- Item not found
- Access denied
- Restoration blocked by policy

10. Security Considerations

- Restoring quarantined files may reintroduce malware; strict approval required.

11. Audit Logging Requirements

- Operator ID
- Item ID
- Timestamp

12. Organizational Benefit Statement

This script ensures quarantined-item restoration is performed safely and with full accountability, supporting troubleshooting and false-positive remediation.

SOP 6 – Update Defender Signatures

Script Name: Update Defender Signatures **Category:** WindowsDefenderAdvanced**

1. Purpose

This script forces an immediate update of Defender malware definitions, supporting incident response, compliance, and security posture.

2. Scope

- Windows servers and workstations
- Defender Antivirus

3. Definitions

- **Signature Update:** Downloading the latest malware definitions.

4. Preconditions

- Operator must have permission to update Defender.
- Internet or WSUS connectivity required.

5. Required Inputs

- None

6. Procedure Steps

1. Trigger Update
 - Force Defender to download latest signatures.
2. Status Retrieval
 - Retrieve new signature version and timestamp.
3. Output Formatting
 - Present structured update summary.
4. Logging
 - Log operator and timestamp.

7. Expected Output

- Confirmation of signature update.

8. Post-Execution Validation

- Operator may verify via Windows Security Center.

9. Error Handling

- Connectivity failure
- Access denied
- Update service unavailable

10. Security Considerations

- Outdated signatures reduce protection; updates should be frequent.

11. Audit Logging Requirements

- Operator ID
- Timestamp

12. Organizational Benefit Statement

This script ensures signature updates are performed safely and consistently, supporting security and compliance.