Workspace ONE Recovery Tool – Complete Documentation

Table of Contents

Part 1: Configuration & Installation in Workspace ONE UEM

1. Overview

Workspace ONE Recovery Tool is a PowerShell-based solution designed to monitor the health of the Workspace ONE Intelligent Hub (formerly AirWatch Agent) on Windows endpoints. If the agent is missing, outdated, or broken, this tool can automatically uninstall and/or re-install the necessary components. This helps keep endpoints consistently enrolled and healthy in Workspace ONE UEM.

2. Prerequisites

1. **Operating System**
   - Windows 10 or later for running the scripts locally and for target endpoints.
2. **Permissions**
   - Administrative rights to run PowerShell scripts on your local machine.
   - Appropriate permissions in Workspace ONE UEM to add/modify Windows applications.
3. **PowerShell Environment**
   - Recommended: PowerShell 5.1 or later.
   - Script execution policy set to allow local scripts (e.g., `RemoteSigned`).
4. **Workspace ONE UEM API Credentials**
   - Ensure you have valid API keys or admin credentials to create and update applications.
5. **Internet Connectivity**
   - The script will connect to Workspace ONE UEM endpoints over the internet or intranet.

3. Download and Extract the Solution

1. **Clone or Download**
   - Go to the GitHub repository and either clone or download the ZIP.
2. **Extract**
   - If you downloaded the ZIP, extract it (e.g., `C:\WorkspaceOneRecoveryTool`).

4. Configure the Tool (`config.json`)

1. **Edit `config.json`**
   - Located in the extracted folder.
   - Common settings include Application Name, Version, Install/Uninstall commands, Workspace ONE UEM Server URL, and credentials.
2. **Save Changes**
   - Provide descriptive names and versions to easily identify the package in Workspace ONE UEM.

5. Run the Upload Script (`upload_to_ws1.ps1`)

1. **Open PowerShell** as Administrator.
2. **Navigate** to the folder containing `upload_to_ws1.ps1`.
3. **Execute**:

   ```
   powershell
   CopyEdit
   .\upload_to_ws1.ps1
   ```

   - The script may prompt for credentials if not already stored.

What the Script Does

1. **Generates a ZIP** of required files (scripts, icons, etc.).
2. **Calculates a SHA256 Hash**.
3. **Checks if the App Exists** in Workspace ONE UEM (updates if it does).
4. **Uploads** the ZIP, uninstall/detection scripts, and icon.
5. **Creates the Application** in Workspace ONE UEM.

6. Verifying in Workspace ONE UEM

1. **Log into** the UEM console.
2. Go to **Apps & Books** > **Native** > **Internal**.
3. **Find** the new or updated application (“Workspace ONE Autorecovery”).
4. **Assign** it to your desired Smart Groups or Device Groups.

7. Troubleshooting & Tips

- **Execution Policy**: `Set-ExecutionPolicy RemoteSigned`.
- **Proxy/Firewall**: Ensure connectivity to UEM’s API endpoints.

- **Logs**: Check logs in the script folder for upload errors.
- **config.json**: Ensure valid JSON formatting.
- **Credentials**: Verify your UEM API key or admin credentials are correct.

8. Maintaining and Updating

1. **Versioning**
   - Increment the version in `config.json` for each change.
2. **Changelog**
   - Optionally maintain a `CHANGELOG.md`.
3. **Pull Requests**
   - If collaborating publicly, use GitHub's PR process.

Part 2: Installation on the Device

1. Introduction

Once you have **packaged** and **uploaded** the Workspace ONE Recovery Tool to Workspace ONE UEM, you can **assign** it to Windows devices. The installation steps described here occur on each device after the assignment is pushed from UEM.

2. Flow Overview

1. **App gets assigned** in UEM.
2. **App installation** begins on the device.
3. **Files copied** to `C:\Windows\UEMRecovery`.
4. **Scheduled Task** created.
5. **SQLite DB** and SQL tables (if configured).
6. **Credentials** encrypted.
7. **`config.json` deleted** (no plain-text credentials remain).
8. **DB access** restricted to local admins & SYSTEM.

3. Detailed Steps

3.1 App Assignment

- The administrator assigns the app via Workspace ONE UEM (e.g., to a Smart Group).
- The device receives an installation command through the Intelligent Hub.

3.2 Files Copied

- The packaged ZIP is extracted into `C:\Windows\UEMRecovery` (or another folder if configured).

- Validations may occur (hash check, version check).

3.3 Scheduled Task Creation

- A Windows Scheduled Task is created to run the main script periodically (often every 4 hours).
- It typically runs as SYSTEM or a specified admin account.

3.4 Database Setup

- **SQLite**: The script creates necessary tables for logging or storing configuration data.
- **SQLite DB**: A local `.sqlite` file is generated in `C:\Windows\UEMRecovery`, and config data is imported.

3.5 Credentials Encryption

- Credentials from `config.json` are read and encrypted.
- The encrypted data is stored in the SQLite DB.

3.6 `config.json` Removed

- `config.json` is deleted from disk once its contents are securely stored.

3.7 Restrict DB Permissions

- NTFS permissions on `C:\Windows\UEMRecovery` are set so that only SYSTEM and local administrators can read/write the database file.

4. Verifying Installation

1. **Check Folder**
   - Confirm `C:\Windows\UEMRecovery` contains the script files and SQLite DB.
2. **Task Scheduler**
   - Verify the scheduled task exists and runs with the correct trigger.
3. **Permissions**
   - Ensure only SYSTEM/admins have file-level access.
4. **Logs**
   - Look for any error messages during installation.

5. Tips & Troubleshooting

- **Permission Issues**: If `config.json` cannot be deleted, confirm no AV lock.
- **Encryption**: If encryption fails, verify .NET or DPAPI libraries.
- **Scheduled Task**: Sometimes tasks are created in subfolders within Task Scheduler.

Part 3: Runtime Logic – Health Checks & Remediation

1. Introduction

After installation, the **Workspace ONE Recovery Tool** runs periodically (by default every 4 hours) to:

1. Check the device enrollment status.
2. Validate Intelligent Hub/Agent health.
3. Check scheduled tasks, event logs, proxies, and other items.
4. Attempt re-enrollment or re-installation if errors are detected.

2. High-Level Flow

1. **Scheduled Task** starts the main script.
2. **Hash Validation → Read Config** from SQLite.
3. **Check Enrollment**: If **Not Enrolled** and **AutoEnrollment** is enabled, attempt enrollment.
4. **Health Checks** if device **Is Enrolled**.
5. **Test** scheduled tasks and agent status.
6. **Remediation** if thresholds or errors are found.
7. **HTML Report** generated and **automatic re-enrollment** triggered if needed.

3. Detailed Steps

3.1 Script Initialization

1. **Scheduled Task Launch**
   - The script (`recovery_main.ps1`) runs every 4 hours.
2. **Validate Hash**
   - Compares its own hash with a stored reference to ensure integrity.
3. **Read Config from SQLite**
   - Retrieves all settings (e.g., intervals, credentials, error thresholds).

3.2 Enrollment Check & Auto-Enrollment

1. **Check if Enrolled**
   - If **No**, check if auto-enrollment is enabled.
   - If auto-enrollment is **disabled**, script ends. If **enabled**, proceed with enrollment.
2. **Enrollment Flow**
   - Possibly create local user, run `automatic_reenrollment.ps1`, schedule tasks, etc.
   - Could involve a device restart and autologon sequence.
3. **Error Threshold**

    - If repeated enrollment failures occur, the script logs them and may generate a final report.

3.3 Health Checks (Device is Enrolled)

1. **Hub Validation**
    - Check the Workspace ONE Intelligent Hub version, running status, and logs.
2. **OMA-DM Scheduled Tasks**
    - Verify Windows MDM tasks are present/active.
3. **Event Logs**
    - Inspect Windows Event Viewer for critical errors or warnings.
4. **WNS (Windows Notification Service)**
    - Check status, expiry, last communication.
5. **AWCM Communication**
    - Confirm AWCM logs/registry entries indicate successful connections.
6. **SFD and Proxy**
    - Validate additional services and proxy configurations.
7. **Pending Reboot**
    - If Windows update or other process requires reboot, log or initiate it.

3.4 Test: Scheduled Tasks

- Verify **Hub Health** scheduled tasks (or others) ran within a configured time window (e.g., 24 hours).
- If not run, **trigger** them manually.

3.5 Remediation

- Re-run the Hub Health task if missing or failing.
- Attempt partial or full agent re-installation if errors persist.
- If an **Error Threshold** is reached (e.g., repeated failures), initiate a deeper re-enrollment or uninstall/reinstall cycle.

3.6 Reporting & Automatic Re-Enrollment

1. **Generate HTML Report**
    - Summarizes the checks performed, errors found, and actions taken.
2. **Error Threshold**
    - If the threshold is exceeded, the tool triggers additional re-enrollment logic or escalates the issue.

4. Tips & Troubleshooting

1. **Hash Failures**
    - Ensure the script wasn’t altered or corrupted.
2. **Database Access**

   - Check that NTFS permissions aren’t blocking the script from reading the DB.
3. **Event Log Overload**
   - Filter out benign warnings to avoid false remediation loops.
4. **Scheduling Frequency**
   - Adjust from 4 hours to a time that fits your environment needs.

5. Conclusion

The **Workspace ONE Recovery Tool** offers a self-healing mechanism for Windows endpoints by regularly validating enrollment, checking critical services, and remediating issues. With these three parts—**Packaging & Upload, Device Installation, and Runtime Logic**—you have a complete reference for deploying and operating the solution in your environment.