

## **Module 02: Script Lifecycle**

---

## Objectives

---

- Upload scripts and validate metadata
- Apply categories and tags
- Understand versioning and integrity checks

## Walkthrough

---

1. Navigate to Scripts and select Upload.
2. Upload a PowerShell script from `test-script.ps1` or create a new file.
3. Add tags and a category, then save.
4. Open the script detail page and review metadata.

# Lifecycle snapshot



# Screenshots

The screenshot shows the PSScript AI-Powered Management interface. On the left is a dark sidebar with the PSScript logo, navigation links (Dashboard, Script Management, AI Assistant, Documentation, UI Components, Settings), and copyright information (@ 2025 PSScript, v1.0). The main area has a title bar with 'Script Details' and a search icon. Below it is a 'Upload Script' section with a file upload input, a preview area showing the script content, and a right panel for entering metadata like Title, Description, Category, Tags, and analysis options. A 'Preview Analysis' button and a 'Upload Script' button are at the bottom.

**Script Content Preview**  
# Enhanced PowerShell script for AI Analysis Testing  
# Author: Claude  
# Version: 2.0  
# Date: 2025-03-11  
  
# Import necessary modules  
Import-Module -Name Microsoft.PowerShell.Security  
Import-Module -Name Microsoft.PowerShell.Utility  
  
function Get-SystemInfo {  
 <#

**Title \***  
test-script

**Description**  
What does this script do?

**Category**  
-- Select Category --

**Tags**  
Add custom tag   
 Make script public  
 Analyze with AI

**Preview Analysis**

**Cancel** **Upload Script**

This screenshot is nearly identical to the one above, but the 'Upload Script' section shows a progress bar indicating the file is being uploaded (Uploading... 0%). The rest of the interface, including the sidebar, preview area, and right-hand metadata panel, remains the same.

**Script Content Preview**  
# Enhanced PowerShell script for AI Analysis Testing  
# Author: Claude  
# Version: 2.0  
# Date: 2025-03-11  
  
# Import necessary modules  
Import-Module -Name Microsoft.PowerShell.Security  
Import-Module -Name Microsoft.PowerShell.Utility  
  
function Get-SystemInfo {  
 <#

**Title \***  
test-script

**Description**  
What does this script do?

**Category**  
-- Select Category --

**Tags**  
Add custom tag   
 Make script public  
 Analyze with AI

**Preview Analysis**

**Cancel** **Uploading...**

## Metadata checklist

---

Field	Example	Notes
Title	Reset-UserPassword	Use Verb-Noun naming
Description	Reset AD user password	Keep it concise
Category	Identity	Match existing taxonomy
Tags	security,active-directory	2-5 meaningful tags
Owner tag	owner:team-identity	Use a consistent owner tag
Hash	auto	Calculated on upload

## **Integrity and deduplication**

---

- The backend calculates a file hash on upload.
- Duplicate uploads are detected and blocked or flagged.

# Verification checklist

---

- Upload succeeds and the script appears in the list.
- Tags and category are visible on the card and detail page.
- The script detail page shows a security and quality score.

Generated 2026-01-16 23:34 UTC