Training Guide

End-to-end walkthrough with screenshots and labs

PSScript Manager Training

PSScript Training Guide

This guide is a full training and enablement program for PSScript Manager. It includes walkthroughs, screenshots, labs, sample actions, and operational checklists.

Table of Contents

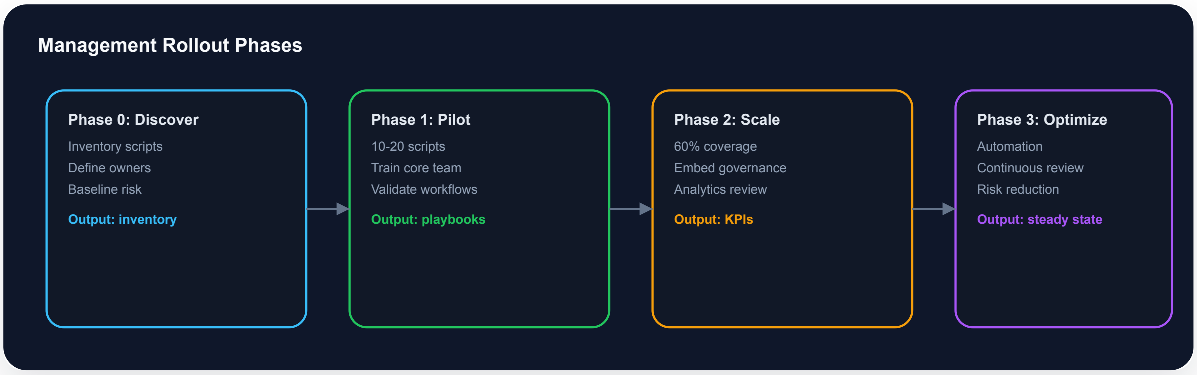
Program overview

PSScript Manager is a full workflow platform for PowerShell scripts. This training builds capability across authoring, analysis, search, and operations.

Expected outcomes: - Scripts are organized, tagged, and versioned - AI analysis is used to identify and remediate risks - Similar scripts can be found via vector search - Analytics and audit trails are used for governance

Management rollout details live in `../MANAGEMENT-PLAYBOOK.md`.

Program phases (management alignment)



Management Rollout Phases

Phase 0: Discover

Inventory scripts
Define owners
Baseline risk

Output: inventory

Phase 1: Pilot

10-20 scripts
Train core team
Validate workflows

Output: playbooks

Phase 2: Scale

60% coverage
Embed governance
Analytics review

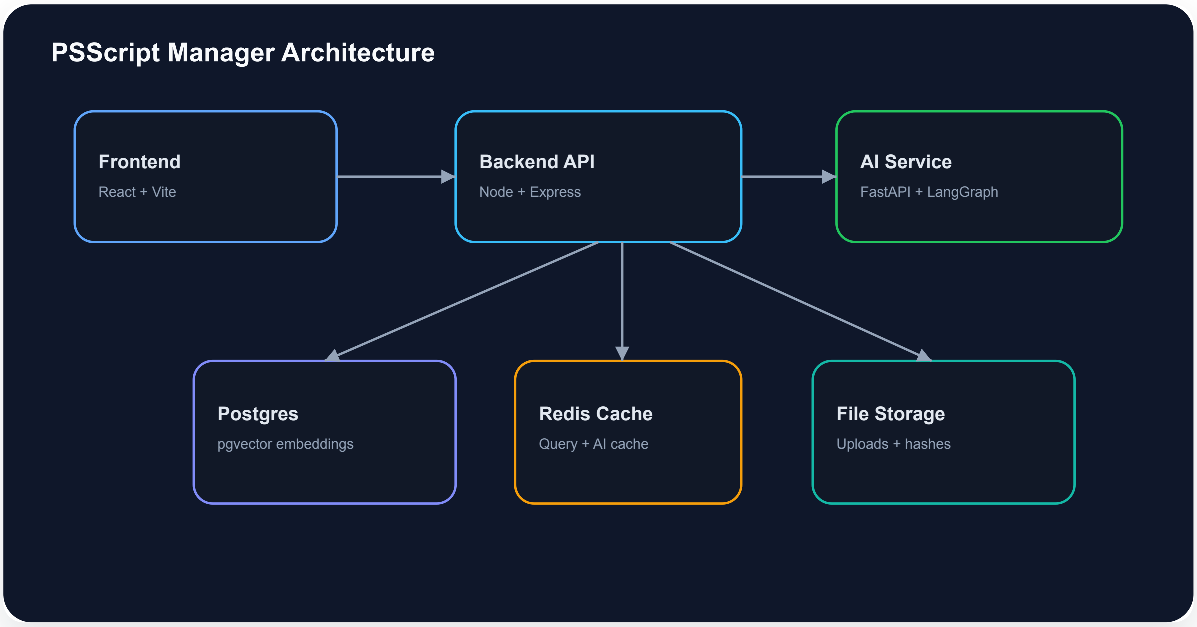
Output: KPIs

Phase 3: Optimize

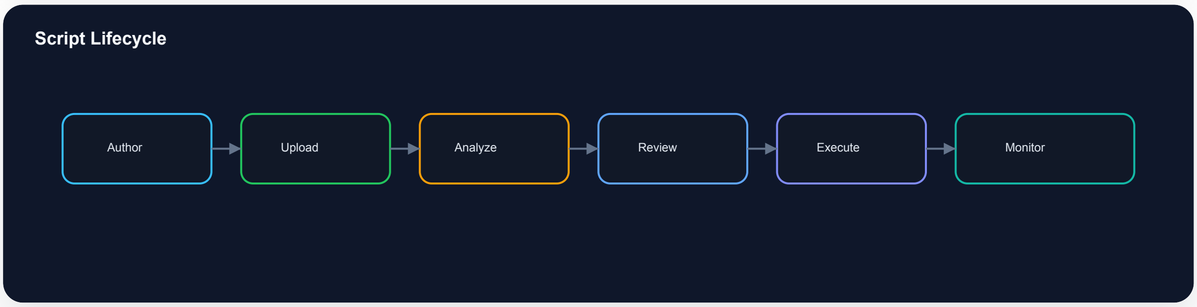
Automation
Continuous review
Risk reduction

Output: steady state

System overview


PSScript Manager Architecture

Analysis Pipeline


Script Lifecycle

Search Modes

Keyword Search

Exact matches for cmdlets and strings

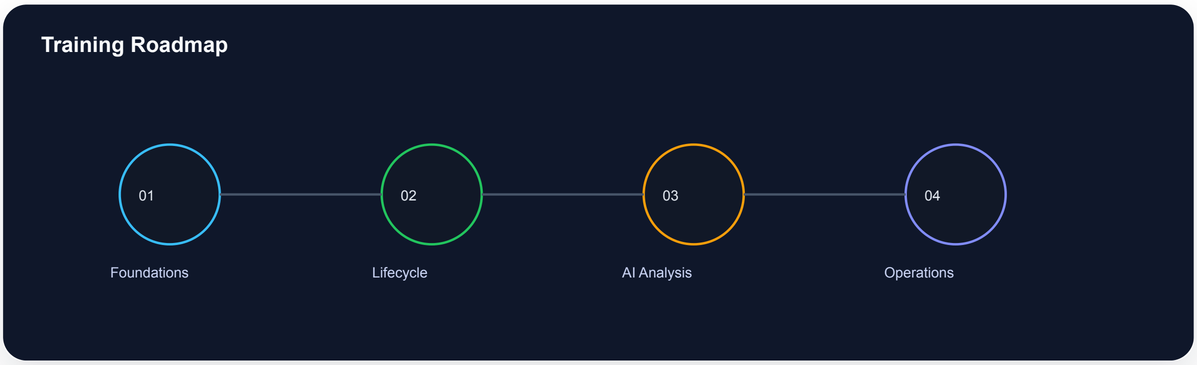
Vector Search

Semantic matches via embeddings

Hybrid Search

Keyword + semantic ranking

Training roadmap



Training Roadmap

01 Foundations

02 Lifecycle

03 AI Analysis

04 Operations

Environment setup

Recommended setup uses mock mode so training can run without external dependencies.

```
./start-all-mock.sh
```

Default ports: - Frontend: http://localhost:3002 - Backend API: http://localhost:4000/api - AI service: http://localhost:8000

Role-based paths

Role	Focus	Primary modules
Script author	Upload, analyze, improve	01, 02, 03
Security reviewer	Risk review, approvals	03, 04, 05
Platform admin	Operations, reliability	01, 05

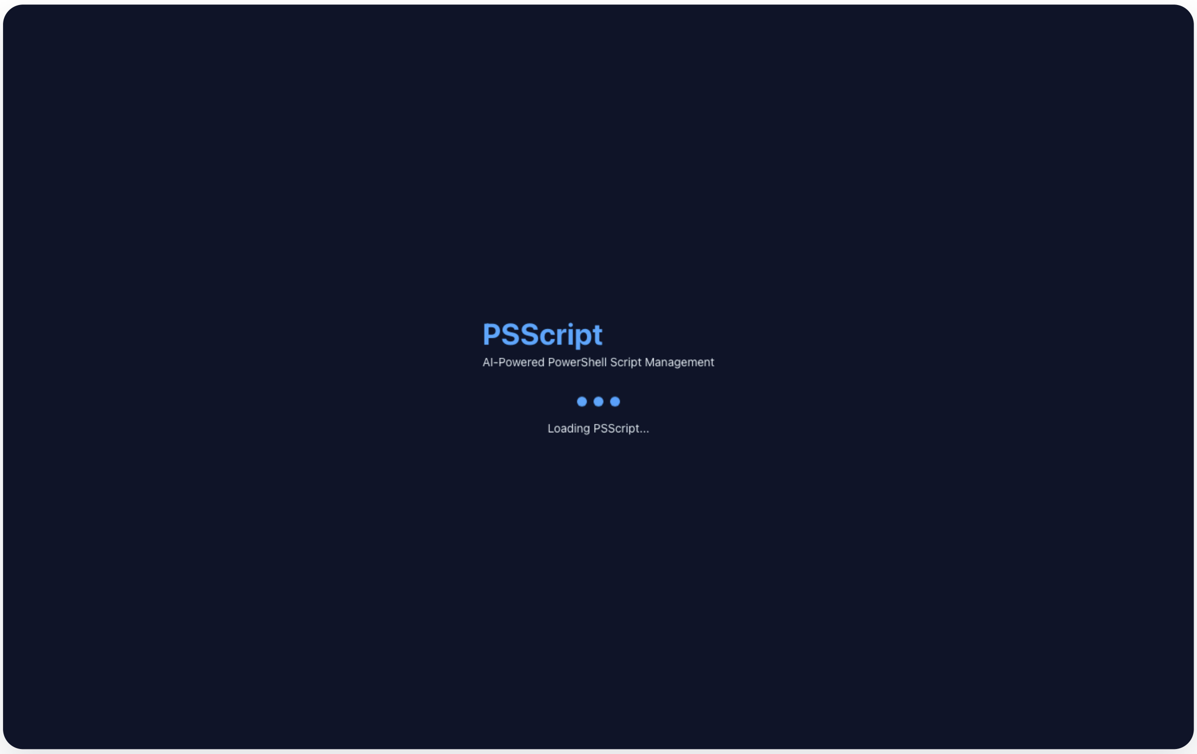
Module breakdown

Module	Objectives	Key screen
Module 01: Foundations	Navigate UI, find docs, understand services	Dashboard
Module 02: Script Lifecycle	Upload, tag, version scripts	Upload
Module 03: AI Analysis	Read scores and recommendations	Analysis
Module 04: Search	Use keyword + vector search	Scripts
Module 05: Operations	Review analytics and logs	Analytics

Hands-on labs

Lab 01: Sign in and orient

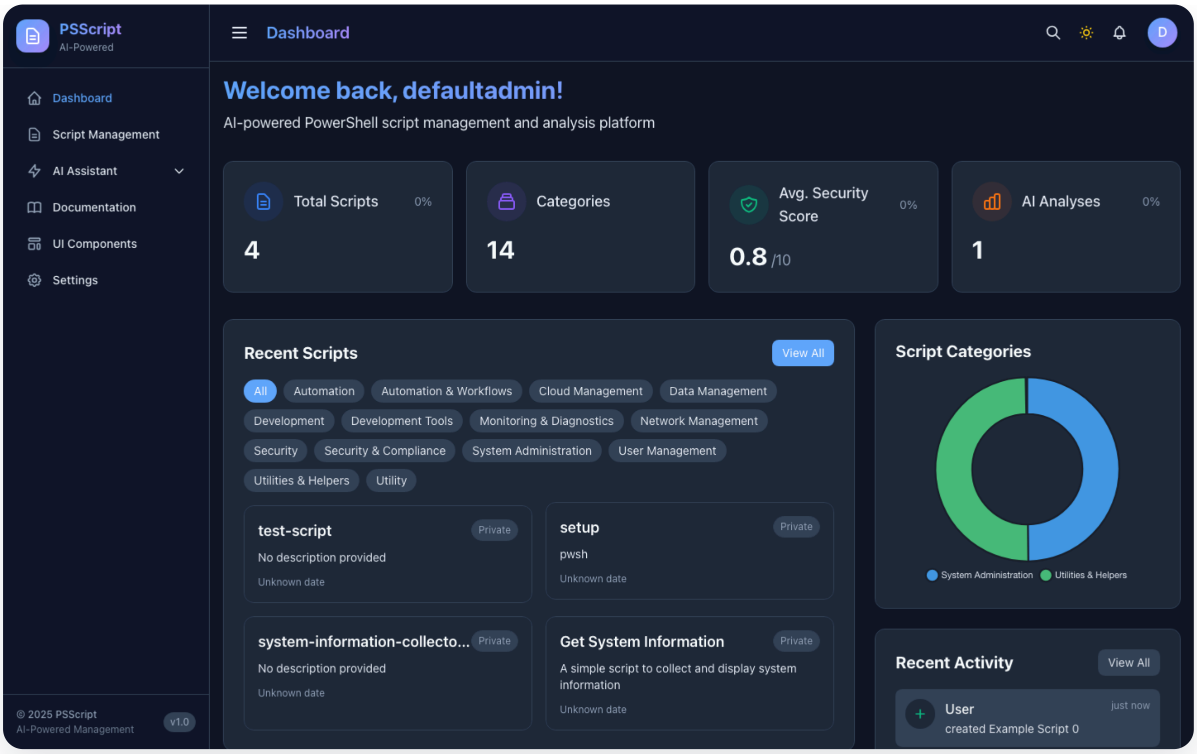
1. Open the login screen and use Default Login.
2. Review the dashboard stats cards and recent scripts.
3. Locate Scripts, Documentation, and Analytics in the sidebar.



PSScript

AI-Powered PowerShell Script Management

Loading PSScript...



Welcome back, defaultadmin!

AI-powered PowerShell script management and analysis platform

Total Scripts 0% 4

Categories 14

Avg. Security Score 0% 0.8 /10

AI Analyses 0% 1

Recent Scripts

test-script — Private — No description provided — Unknown date

setup — Private — pwsh — Unknown date

system-information-collecto... — Private — No description provided — Unknown date

Get System Information — Private — A simple script to collect and display system information — Unknown date

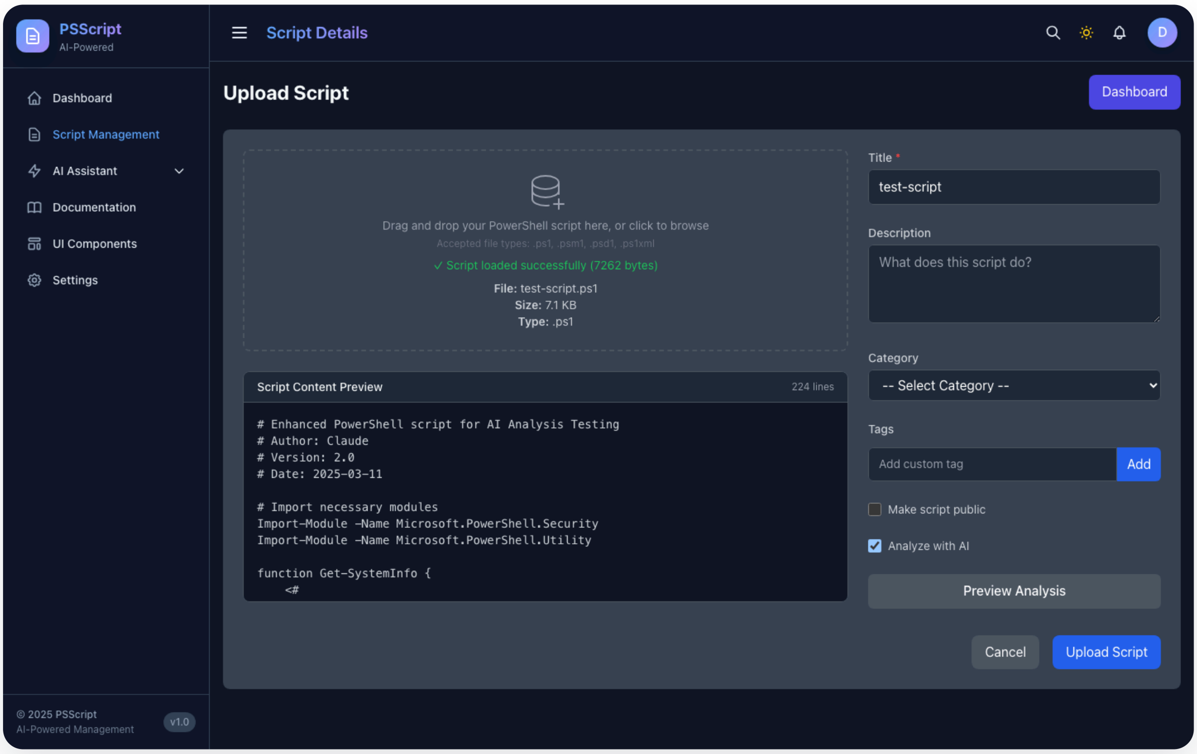
Script Categories

Recent Activity

User created Example Script 0 — just now

Lab 02: Upload and analyze a script

1. Navigate to Upload.
2. Upload `test-script.ps1` and add tags.
3. Preview analysis, then submit.
4. Open the script detail view and review scores.



Upload Script

Drag and drop your PowerShell script here, or click to browse

Accepted file types: .ps1, .psm1, .psd1, .ps1xml

✓ Script loaded successfully (7262 bytes)

File: test-script.ps1
Size: 7.1 KB
Type: .ps1

Script Content Preview — 224 lines

```
# 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

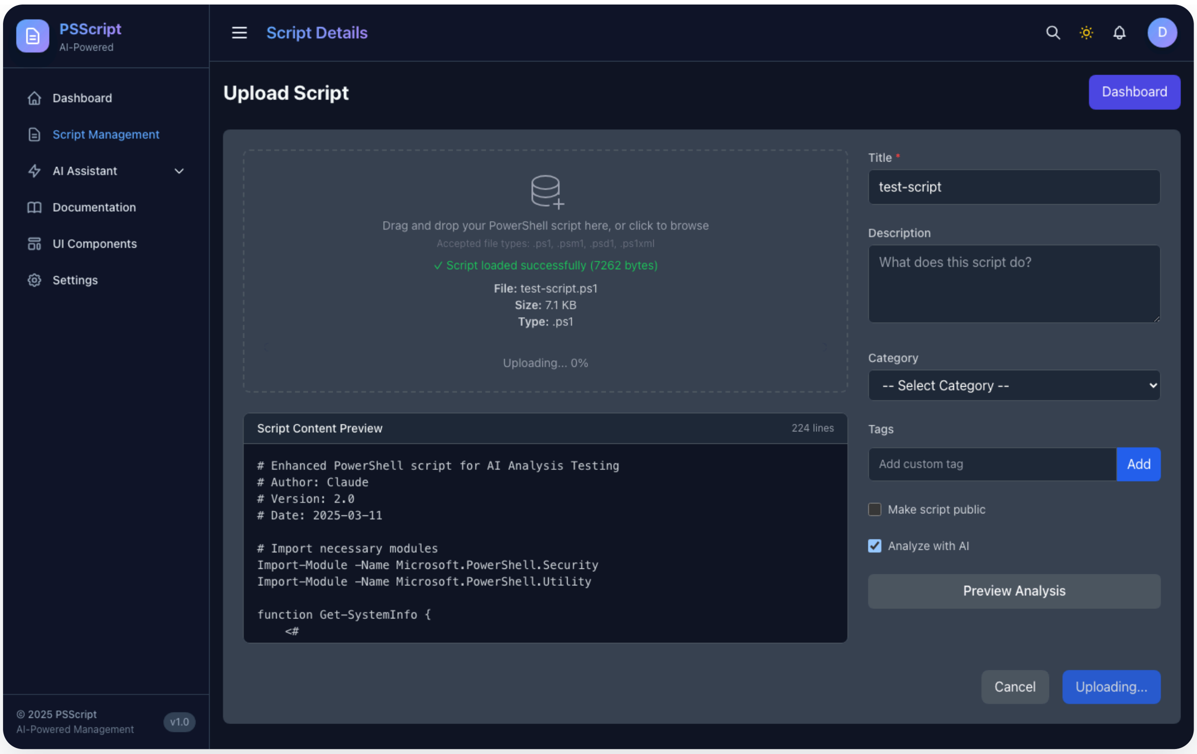
Description: What does this script do?

Category: -- Select Category --

Tags: Add custom tag

☐ Make script public

☑ Analyze with AI



Upload Script

Drag and drop your PowerShell script here, or click to browse

Accepted file types: .ps1, .psm1, .psd1, .ps1xml

✓ Script loaded successfully (7262 bytes)

File: test-script.ps1
Size: 7.1 KB
Type: .ps1

Uploading... 0%

Script Content Preview — 224 lines

```
# 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

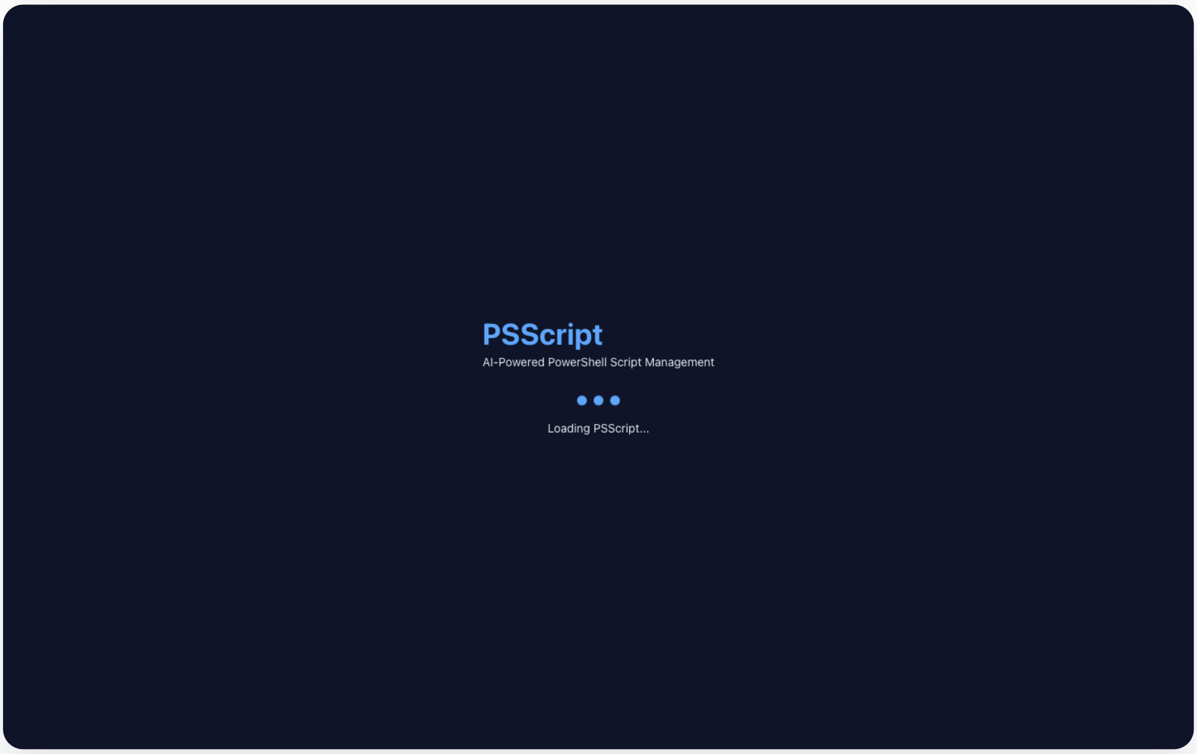
Description: What does this script do?

Category: -- Select Category --

Tags: Add custom tag

□ Make script public

☑ Analyze with AI



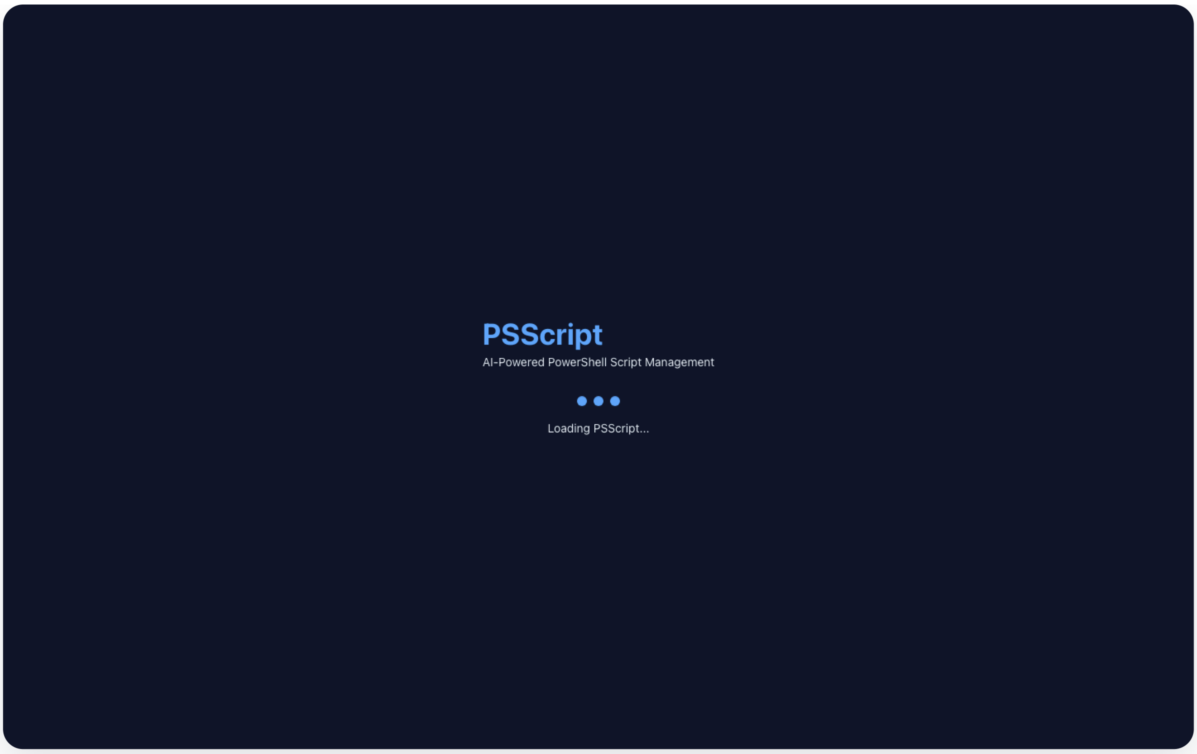
PSScript

AI-Powered PowerShell Script Management

Loading PSScript...

Lab 03: Documentation and AI chat

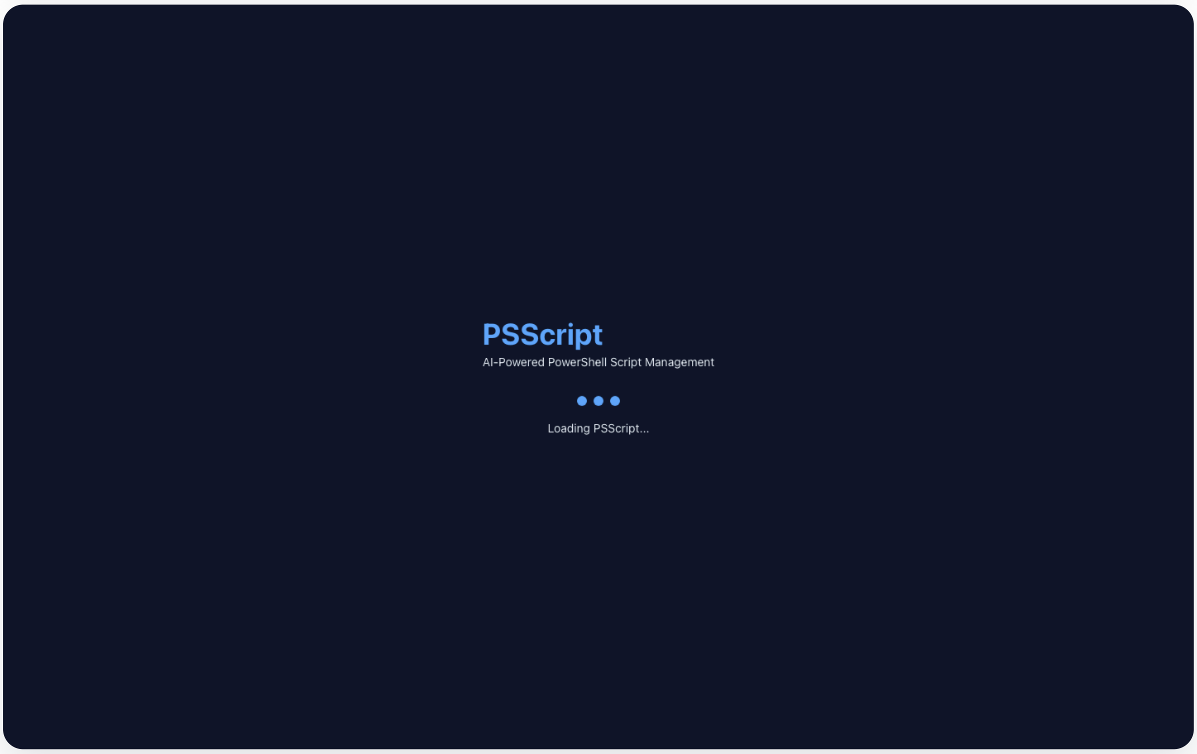
1. Open Documentation and search for a cmdlet.
2. Capture an example in your notes.
3. Open AI Chat and ask for safe usage patterns.



PSScript

AI-Powered PowerShell Script Management

Loading PSScript...



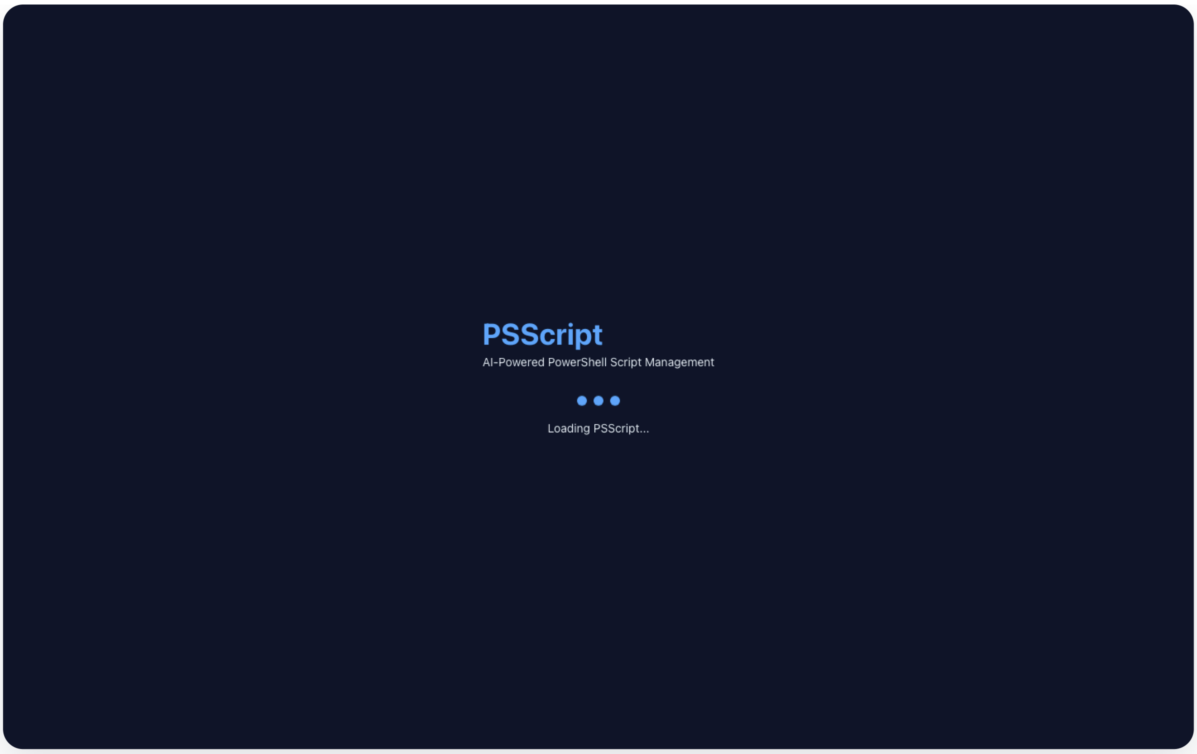
PSScript

AI-Powered PowerShell Script Management

Loading PSScript...

Lab 04: Analytics and governance

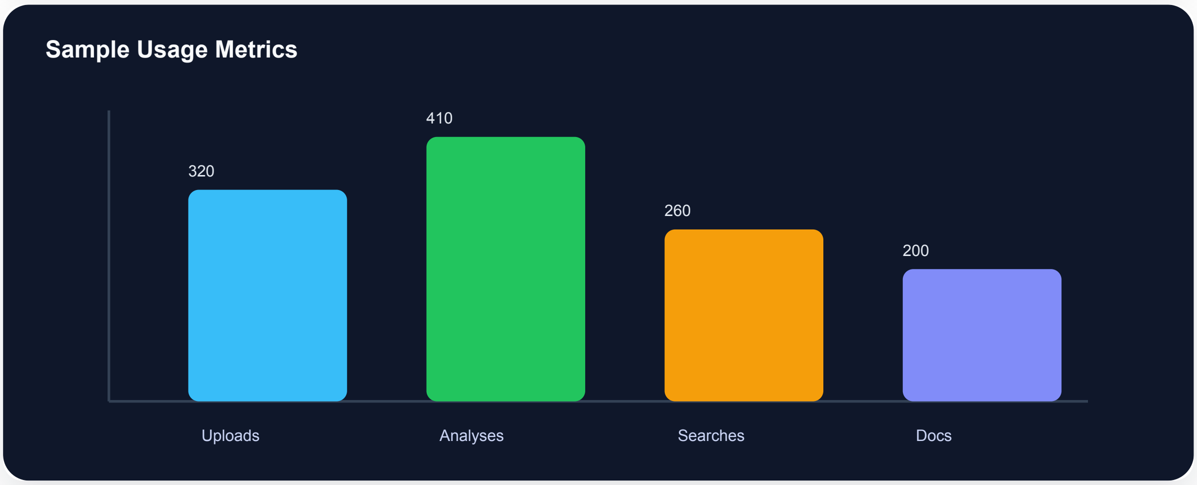
1. Open Analytics and review usage metrics.
2. Identify top activity areas.
3. Document a governance checklist for releases.



PSScript

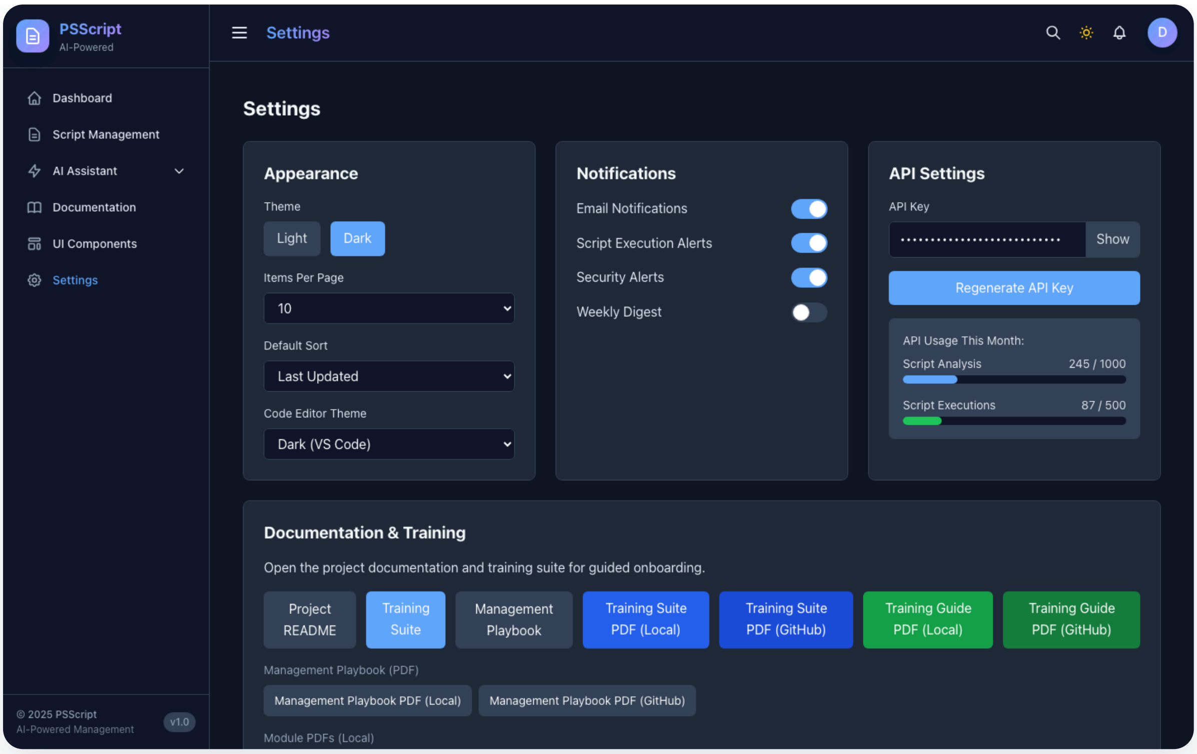
AI-Powered PowerShell Script Management

Loading PSScript...



Lab 05: Settings and training resources

1. Open Settings and review API usage and notifications.
2. Locate the Documentation and Training section.
3. Confirm the Training Suite link opens the guide.
4. Verify the local PDF export loads at `http://localhost:4000/docs/exports/pdf/Training-Guide.pdf`.



Settings

Appearance

Theme

Light [✓ Dark]

Items Per Page

10

Default Sort

Last Updated

Code Editor Theme

Dark (VS Code)

Notifications

Email Notifications ●

Script Execution Alerts ●

Security Alerts ●

Weekly Digest ○

API Settings

API Key

•••••••••••••••••••••••••••• Show

Regenerate API Key

API Usage This Month:

Script Analysis	245 / 1000
Script Executions	87 / 500

Documentation & Training

Open the project documentation and training suite for guided onboarding.

Project README
Training Suite
Management Playbook
Training Suite PDF (Local)
Training Suite PDF (GitHub)
Training Guide PDF (Local)
Training Guide PDF (GitHub)

Management Playbook (PDF)

Management Playbook PDF (Local)
Management Playbook PDF (GitHub)

Module PDFs (Local)

Sample actions

Example: Upload via API

```
curl -X POST http://localhost:4000/api/scripts \
  -H "Authorization: Bearer <token>" \
  -F "file=@test-script.ps1" \
  -F "title=Reset-UserPassword" \
  -F "description=Reset AD user password" \
  -F "tags=security,active-directory"
```

Example: Run analysis

```
curl -X POST http://localhost:8000/analyze \
  -H "Content-Type: application/json" \
  -d '{"content": "Get-ADUser -Filter *", "type": "security"}'
```

Example: Vector search

```
curl http://localhost:4000/api/scripts/1/similar
```

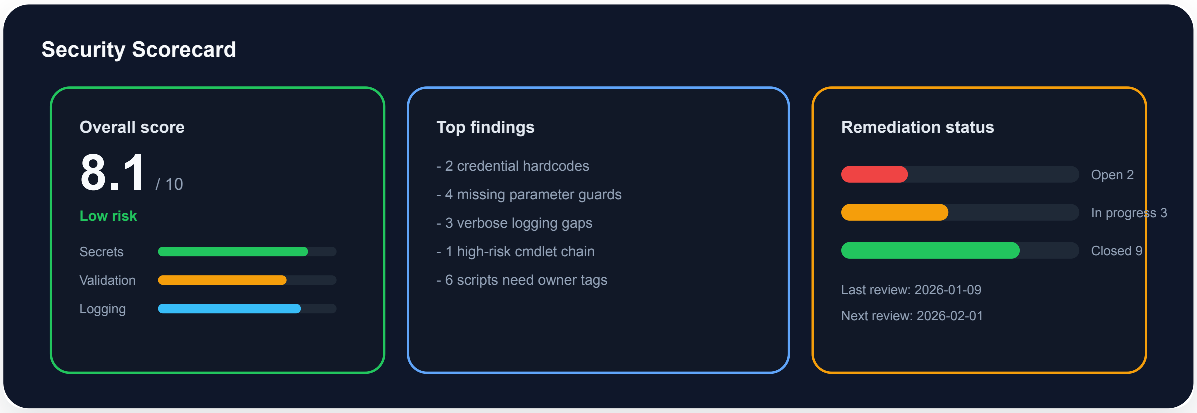
```
curl -X POST http://localhost:8000/similar \
  -H "Content-Type: application/json" \
  -d '{"content": "active directory onboarding", "limit": 5}'
```

Operational readiness

Governance checklist

- Scripts are tagged, categorized, and have an owner
- Security score is reviewed and exceptions are documented
- Duplicates are prevented by hash check
- Analysis results are logged and stored
- Analytics reflect script usage and review cadence

Security scorecard



Security Scorecard

Overall score

8.1 / 10

Low risk

Top findings

- 2 credential hardcodes
- 4 missing parameter guards
- 3 verbose logging gaps
- 1 high-risk cmdlet chain
- 6 scripts need owner tags

Remediation status

Last review: 2026-01-09

Next review: 2026-02-01

Scorecard rubric

Signal	Description	Target
Security score	Weighted risk score from AI analysis	>= 7.5
High-risk findings	Count of high severity issues	0
Remediation SLA	Days to close high-risk findings	<= 7
Ownership coverage	Scripts with owner tag	100%
Review cadence	Days between audits	<= 30

Operational checks

Area	Check	Target
Backend	Health endpoint	200 OK
Database	pgvector enabled	true
AI service	Analyze endpoint	200 OK
Cache	Redis reachable	Optional
UI	Dashboard load	< 3 seconds

For deeper triage and support patterns, see `../SUPPORT.md`.

Appendix: Troubleshooting

Issue	Likely cause	Fix
Upload fails	Backend not running	Start backend on port 4000
Analysis empty	AI service down	Start AI service or mock mode
Search returns none	Embeddings missing	Run embedding generation
Docs empty	Crawl not completed	Run documentation crawl

Generated 2026-01-13 06:26 UTC