**Blackfoot Cloud Security Assessment Methodology**

**A Comprehensive Guide for AWS, Azure, GCP, Kubernetes, and Microsoft 365**

**Version:** 0.1  
**Date:** June 2025  
**Classification:** Internal Use

Table of Contents

[Blackfoot Cloud Security Assessment Methodology 1](#_Toc201178736)

[A Comprehensive Guide for AWS, Azure, GCP, Kubernetes, and Microsoft 365 1](#_Toc201178737)

[Document Control 4](#_Toc201178738)

[1. Executive Summary 5](#_Toc201178739)

[2. Introduction 5](#_Toc201178740)

[3. Assessment Framework and Standards 6](#_Toc201178741)

[**3.1 CIS Benchmarks** 6](#_Toc201178742)

[**3.2 OWASP Cloud Top 10** 6](#_Toc201178743)

[**3.3 Cloud Security Alliance (CSA) Cloud Controls Matrix** 6](#_Toc201178744)

[**3.4 NIST Cybersecurity Framework** 6](#_Toc201178745)

[**3.5 PCI Data Security Standard (PCI-DSS)** 7](#_Toc201178746)

[**3.6 General Data Protection Regulation (GDPR)** 7](#_Toc201178747)

[4. Assessment Methodology 8](#_Toc201178748)

[**4.1 Phase 1: Reconnaissance and Scoping** 8](#_Toc201178749)

[**4.2 Phase 2: Vulnerability Analysis and Configuration Review** 8](#_Toc201178750)

[**4.3 Phase 3: Exploitation** 9](#_Toc201178751)

[**4.4 Phase 4: Reporting and Remediation** 9](#_Toc201178752)

[5. Recommended Assessment Tools 11](#_Toc201178753)

[6. Required Permissions and Access 12](#_Toc201178754)

[**6.1 Amazon Web Services (AWS)** 12](#_Toc201178755)

[**6.2 Microsoft Azure** 12](#_Toc201178756)

[**6.3 Google Cloud Platform (GCP)** 12](#_Toc201178757)

[**6.4 Kubernetes** 13](#_Toc201178758)

[**6.5 Microsoft 365** 13](#_Toc201178759)

[7. Tool Installation and Configuration 15](#_Toc201178760)

[**7.1 Powerpipe** 15](#_Toc201178761)

[**7.2 Prowler** 16](#_Toc201178762)

[**7.3 ScoutSuite** 19](#_Toc201178763)

[**7.4 Kubernetes Security Tools** 21](#_Toc201178764)

[**7.5 Microsoft 365 Security Tools** 22](#_Toc201178765)

[8. Authentication Configuration 24](#_Toc201178766)

[**8.1 AWS Authentication** 24](#_Toc201178767)

[**8.2 Azure Authentication** 24](#_Toc201178768)

[**8.3 GCP Authentication** 25](#_Toc201178769)

[**8.4 Kubernetes Authentication** 25](#_Toc201178770)

[**8.5 Microsoft 365 Authentication** 27](#_Toc201178771)

[9. Assessment Execution 30](#_Toc201178772)

[**9.1 Complete Assessment Workflow** 30](#_Toc201178773)

[**9.2 Compliance Framework Testing** 31](#_Toc201178774)

[**9.2 Compliance Framework Testing** 32](#_Toc201178775)

[**9.3 Output and Reporting Options** 34](#_Toc201178776)

[10. Quick Reference Guide 35](#_Toc201178777)

[**Tool Capability Matrix** 35](#_Toc201178778)

[**Common Command Patterns** 35](#_Toc201178779)

[11. Additional Resources 37](#_Toc201178780)

[**Documentation Links** 37](#_Toc201178781)

[**Framework References** 38](#_Toc201178782)

[**Platform-Specific Security Resources** 38](#_Toc201178783)

[12. Appendices 39](#_Toc201178784)

[**Appendix A: Troubleshooting Common Issues** 39](#_Toc201178785)

# Document Control

|  |  |  |
| --- | --- | --- |
| Author | Job Title | Contact Email |
| Victor Majiso | Principal Security Consultant | Victor.majiso@blackfootuk.com |

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Comments | Reviewed by | Date |
| 0.1 | Draft | Victor Majiso | 18 June 2025 |
| 0.2 |  |  |  |
| 0.3 |  |  |  |

**1. Executive Summary**

This methodology document provides a comprehensive framework for conducting cloud security assessments across the three major cloud service providers: Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), as well as Kubernetes environments and Microsoft 365 (M365) tenants. The document outlines a systematic approach to identifying security vulnerabilities, configuration weaknesses, and compliance violations in cloud and hybrid environments.

The methodology is aligned with industry-standard frameworks including CIS Benchmarks, OWASP Cloud Top 10, Cloud Security Alliance (CSA) Cloud Controls Matrix, NIST Cybersecurity Framework, PCI Data Security Standard (PCI-DSS), and General Data Protection Regulation (GDPR). It provides detailed guidance on tool selection, configuration, and execution to ensure consistent and thorough security assessments across all supported platforms.

**2. Introduction**

Cloud security assessments are critical for organizations to maintain a strong security posture and ensure compliance with industry standards and regulations. This methodology covers penetration testing and configuration assessment techniques specifically designed for cloud environments, container orchestration platforms, and cloud-based productivity suites.

The assessment approach is structured around four key phases that ensure comprehensive coverage of cloud security domains while maintaining a systematic and repeatable process. The methodology supports both automated scanning and manual testing techniques to provide thorough coverage of potential security issues across AWS, Azure, GCP, Kubernetes, and Microsoft 365 environments.

**3. Assessment Framework and Standards**

When conducting cloud security assessments, it is essential to align the methodology with established frameworks to ensure comprehensive coverage and industry compliance.

**3.1 CIS Benchmarks**

The Centre for Internet Security provides detailed, prescriptive guidance for securing AWS, Azure, and GCP. The tools mentioned in this methodology, particularly Powerpipe, Prowler, and ScoutSuite, have built-in checks for these benchmarks. CIS Benchmarks provide:

* Detailed configuration recommendations
* Prioritised security controls
* Automated assessment capabilities
* Regular updates to address emerging threats

**3.2 OWASP Cloud Top 10**

This project from the Open Web Application Security Project outlines the most critical security risks in cloud environments. It provides a valuable lens through which to analyse findings and prioritize remediation efforts. The top risks include:

* Misconfigured cloud storage
* Insecure APIs
* Inadequate identity and access management
* Insufficient logging and monitoring
* Insecure interfaces and APIs

**3.3 Cloud Security Alliance (CSA) Cloud Controls Matrix**

The CCM is a cybersecurity control framework for cloud computing that provides a comprehensive set of security controls mappable to various industry standards and regulations. It serves as an excellent resource for developing a holistic cloud security program.

**3.4 NIST Cybersecurity Framework**

The National Institute of Standards and Technology (NIST) Cybersecurity Framework provides a policy framework of computer security guidance for how private sector organizations can assess and improve their ability to prevent, detect, and respond to cyber-attacks. The framework consists of:

* **Identify** - Develop organizational understanding to manage cybersecurity risk
* **Protect** - Implement appropriate safeguards to ensure delivery of critical services
* **Detect** - Implement appropriate activities to identify cybersecurity events
* **Respond** - Implement appropriate activities to take action regarding detected cybersecurity incidents
* **Recover** - Implement appropriate activities to maintain plans for resilience and restore capabilities

**3.5 PCI Data Security Standard (PCI-DSS)**

The Payment Card Industry Data Security Standard is an information security standard for organisations that handle branded credit cards. PCI-DSS applies to all entities involved in payment card processing and includes requirements for:

* Secure network architecture
* Cardholder data protection
* Vulnerability management programs
* Strong access control measures
* Network monitoring and testing
* Information security policies

**3.6 General Data Protection Regulation (GDPR)**

GDPR is a comprehensive data protection regulation that affects how organizations collect, process, and protect personal data. Key requirements include:

* Data protection by design and by default
* Privacy impact assessments
* Data breach notification requirements
* Individual rights and consent management
* International data transfer restrictions
* Accountability and governance requirements

**4. Assessment Methodology**

The cloud security assessment methodology consists of four distinct phases that ensure comprehensive coverage of cloud security domains regardless of the cloud service provider.

**4.1 Phase 1: Reconnaissance and Scoping**

The initial phase focuses on gathering comprehensive information about the target cloud environment and establishing clear assessment boundaries.

**Objective:** To identify the external attack surface, enumerate cloud assets, and understand the relationships between them.

**Key Activities:**

* Identifying public-facing services (websites, APIs, storage buckets)
* Enumerating subdomains and related cloud services
* Gathering information about organisational cloud usage patterns
* Defining scope and testing windows
* Establishing communication protocols with stakeholders

**Deliverables: TAF**

* Asset inventory
* Scope definition document
* Testing schedule and contact information

**4.2 Phase 2: Vulnerability Analysis and Configuration Review**

This phase involves active scanning for vulnerabilities and misconfigurations within the cloud environment, utilizing automated tools to identify deviations from security best practices.

**Objective:** To identify security weaknesses, misconfigurations, and compliance violations.

**Key Activities:**

* Scanning for open ports and vulnerable services
* Reviewing Identity and Access Management (IAM) policies for excessive permissions
* Assessing network security configurations (security groups, firewalls)
* Checking for publicly exposed storage and databases
* Auditing logging and monitoring configurations
* Running compliance benchmark assessments
* Analysing encryption configurations

**Deliverables:**

* Vulnerability scan results
* Configuration assessment reports
* Compliance gap analysis
* Risk-rated findings inventory

**4.3 Phase 3: Exploitation**

Once vulnerabilities are identified, this phase involves controlled attempts to exploit them to understand their potential impact and business risk.

**Objective:** To demonstrate the real-world risk of identified vulnerabilities through controlled exploitation.

**Key Activities:**

* Attempting to gain unauthorized access to data or services
* Testing for privilege escalation paths
* Simulating lateral movement across the cloud environment
* Validating the impact of configuration weaknesses
* Testing data exfiltration scenarios

**Important Note:** All exploitation activities must be conducted in a controlled and non-disruptive manner with appropriate safeguards and rollback procedures.

**Deliverables:**

* Proof-of-concept demonstrations
* Impact assessment reports
* Evidence documentation
* Risk validation results

**4.4 Phase 4: Reporting and Remediation**

The final phase involves comprehensive documentation of all findings with clear, actionable remediation guidance prioritized by risk level.

**Objective:** To provide a comprehensive overview of the security posture and a roadmap for improvement.

**Key Activities:**

* Creating detailed reports with executive summaries and technical findings
* Providing evidence of vulnerabilities (screenshots, logs, configurations)
* Offering clear and concise remediation steps
* Presenting findings to stakeholders
* Discussing remediation plans and timelines
* Providing follow-up support for remediation validation

**Deliverables:**

* Executive summary report
* Detailed technical findings report
* Remediation roadmap
* Presentation materials
* Remediation validation plan

**5. Recommended Assessment Tools**

The following tools provide a good coverage for cloud security assessments across AWS, Azure, GCP, Kubernetes, and Microsoft 365. It must be noted that there are some parts of the assessment that require manual verification- you need to login to the GUI and check them. The automated tools below are not able to check those configuration items.

**Multi-Cloud Tools:**

* **Prowler** - Multi-cloud security tool for assessments, audits, and compliance checks
* **ScoutSuite** - Multi-cloud security auditing tool with HTML reporting interface
* **Powerpipe** - Modern dashboard and benchmarking tool for compliance visualization

**Cloud-Specific Tools:**

* **Stormspotter** - Azure-focused attack path analysis tool
* **Pacu** - AWS exploitation framework for penetration testing
* **GCPBucketBrute** - GCP storage bucket enumeration tool
* **gcp-scanner** - Comprehensive GCP security scanner

**Kubernetes Security Tools:**

* **kube-bench** - CIS Kubernetes benchmark assessment tool
* **kube-hunter** - Kubernetes penetration testing tool
* **falco** - Runtime security monitoring for Kubernetes
* **OPA Gatekeeper** - Policy enforcement for Kubernetes
* **Polaris** - Kubernetes configuration validation

**Microsoft 365 Security Tools:**

* **Microsoft 365 Defender** - Native security platform
* **Hawk** - PowerShell-based Office 365 incident response tool
* **SPARROW** - Microsoft 365/Azure incident response tool
* **ROADtools** - Azure AD enumeration framework
* **Microsoft Secure Score** - Built-in security posture assessment

Each tool serves specific purposes within the assessment methodology and provides complementary capabilities for thorough security analysis.

**6. Required Permissions and Access**

To effectively conduct compliance assessments across cloud platforms, the assessment tools require specific IAM permissions. These permissions are designed to provide necessary read-only access without allowing modifications to ensure resource security.

**6.1 Amazon Web Services (AWS)**

**Required AWS Managed Policies:**

* SecurityAudit - Provides read-only access to security-relevant resources
* ViewOnlyAccess - Grants read-only access to all AWS services and resources

These policies provide comprehensive read access necessary for security assessments while maintaining the principle of least privilege.

**6.2 Microsoft Azure**

**Required Subscription-Level Roles:**

* Reader - Provides read access to all resources
* Security Reader - Grants read access to security centre and security policies

**Required Microsoft Graph API Permissions:**

* Application.Read.All - Read application registrations
* AuditLog.Read.All - Read audit log data
* Directory.Read.All - Read directory data
* Domain.Read.All - Read domain information
* Group.Read.All - Read group information
* IdentityProvider.Read.All - Read identity provider configurations
* Policy.Read.All - Read policy configurations
* User.Read.All - Read user information

**6.3 Google Cloud Platform (GCP)**

**Required Predefined IAM Roles:**

* Viewer - Provides read access to all resources
* Security Reviewer - Grants access to security and compliance information
* Stackdriver Account Viewer - Required for some ScoutSuite functionality

These roles ensure comprehensive access to security-relevant information across all GCP services.

**6.4 Kubernetes**

**Required Kubernetes RBAC Permissions:**

For comprehensive Kubernetes security assessments, the following ClusterRole permissions are recommended:

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRole

metadata:

name: security-assessor

rules:

- apiGroups: [""]

resources: ["\*"]

verbs: ["get", "list"]

- apiGroups: ["apps"]

resources: ["\*"]

verbs: ["get", "list"]

- apiGroups: ["networking.k8s.io"]

resources: ["\*"]

verbs: ["get", "list"]

- apiGroups: ["policy"]

resources: ["\*"]

verbs: ["get", "list"]

- apiGroups: ["rbac.authorization.k8s.io"]

resources: ["\*"]

verbs: ["get", "list"]

- apiGroups: ["security.openshift.io"]

resources: ["\*"]

verbs: ["get", "list"]

**Minimum Required Access:**

* cluster-reader role or equivalent read-only cluster access
* Access to pod security policies and network policies
* Ability to read secrets metadata (not contents)
* Access to RBAC configurations
* Node-level access for CIS benchmark compliance checks

**6.5 Microsoft 365**

**Required Microsoft 365 Roles:**

**Administrative Roles:**

* Security Reader - Read access to security features and reports
* Compliance Administrator - Access to compliance centre and policies
* Global Reader - Read-only access to all Microsoft 365 services

**API Permissions (Microsoft Graph):**

* AuditLog.Read.All - Read audit log data
* Directory.Read.All - Read directory data
* Policy.Read.All - Read conditional access and other policies
* SecurityEvents.Read.All - Read security events and alerts
* ThreatIndicators.Read.All - Read threat intelligence
* User.Read.All - Read user profiles and properties
* Group.Read.All - Read group information
* Application.Read.All - Read application registrations
* DeviceManagementConfiguration.Read.All - Read device compliance policies
* Mail.Read - Read mailbox configurations (if required)
* Files.Read.All - Read SharePoint and OneDrive configurations

**Exchange Online Permissions:**

* View-Only Audit Logs - Access to Exchange audit logs
* View-Only Configuration - Read Exchange configuration
* View-Only Recipients - Read recipient information

**SharePoint Online Permissions:**

* SharePoint Service Administrator (read-only operations)
* Site Collection Administrator (for specific site assessments)

**7. Tool Installation and Configuration**

**7.1 Powerpipe**

Powerpipe is a modern dashboard and benchmarking tool for visualizing cloud configurations and assessing security posture against compliance frameworks.

**Installation Options**

**Linux/Ubuntu:**

# Official installation script (recommended)

sudo /bin/sh -c "$(curl -fsSL https://powerpipe.io/install/powerpipe.sh)"

# Verify installation

powerpipe --version

**macOS:**

# Using Homebrew

brew install turbot/tap/powerpipe

# Verify installation

powerpipe --version

**Docker:**

# Pull the latest image

docker pull turbot/powerpipe

# Run Powerpipe in Docker

docker run --rm -it turbot/powerpipe --version

**Configuration**

Powerpipe requires Steampipe for data access:

# Install Steampipe

sudo /bin/sh -c "$(curl -fsSL https://steampipe.io/install/steampipe.sh)"

# Install cloud provider plugins

steampipe plugin install aws

steampipe plugin install azure

steampipe plugin install gcp

# Install Kubernetes plugin

steampipe plugin install kubernetes

# Install Microsoft 365 plugins

steampipe plugin install azuread

steampipe plugin install microsoft365

**Kubernetes Configuration**

# Configure Kubernetes connection

steampipe plugin install kubernetes

# Set kubeconfig path (if not default)

export KUBECONFIG=/path/to/kubeconfig

# Verify connection

steampipe query "select name, namespace from kubernetes\_pod limit 5"

**Microsoft 365 Configuration**

# Install M365 plugins

steampipe plugin install azuread

steampipe plugin install microsoft365

# Configure connection (requires app registration)

# Create ~/.steampipe/config/azuread.spc

cat > ~/.steampipe/config/azuread.spc << EOF

connection "azuread" {

plugin = "azuread"

tenant\_id = "your-tenant-id"

client\_id = "your-client-id"

client\_secret = "your-client-secret"

}

EOF

# Create ~/.steampipe/config/microsoft365.spc

cat > ~/.steampipe/config/microsoft365.spc << EOF

connection "microsoft365" {

plugin = "microsoft365"

tenant\_id = "your-tenant-id"

client\_id = "your-client-id"

client\_secret = "your-client-secret"

}

EOF

**Assessment Commands**

# Install compliance mods

powerpipe mod install github.com/turbot/steampipe-mod-aws-compliance

powerpipe mod install github.com/turbot/steampipe-mod-azure-compliance

powerpipe mod install github.com/turbot/steampipe-mod-gcp-compliance

# Run compliance benchmarks

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140

powerpipe benchmark run azure\_compliance.benchmark.cis\_v140

powerpipe benchmark run gcp\_compliance.benchmark.cis\_v120

# Run specific controls

powerpipe control run aws\_compliance.control.cis\_v140\_2\_1\_1

# Generate dashboard

powerpipe server

# Access via http://localhost:9033

**7.2 Prowler**

Prowler is an open-source cloud security tool that performs security assessments, audits, and compliance checks across multiple cloud platforms.

**Installation Methods**

**Method 1: Using pipx (Recommended):**

# Install pipx

sudo apt update

sudo apt install pipx

pipx ensurepath

# Install Prowler

pipx install prowler

# Verify installation

prowler --version

**Method 2: Using pip:**

# Install Python dependencies

sudo apt update

sudo apt install python3 python3-pip

# Install Prowler

pip3 install prowler

# Verify installation

prowler --version

**Method 3: Docker:**

# Pull Prowler image

docker pull prowler/prowler

# Run with Docker

docker run --rm -it \

-v ~/.aws:/home/prowler/.aws \

-v /tmp/prowler-output:/home/prowler/output \

prowler/prowler aws

**Assessment Commands**

**AWS Assessments:**

# Complete AWS assessment

prowler aws

# Specific regions only

prowler aws --filter-region eu-west-1 eu-west-2

# Compliance frameworks

prowler aws --compliance cis\_2.0\_aws

prowler aws --compliance pci\_3.2.1\_aws

prowler aws --compliance nist\_800\_53\_revision\_5\_aws

# Specific services

prowler aws --services s3 ec2 iam

# Custom output formats

prowler aws --output-modes csv json-ocsf html

**Azure Assessments:**

# Service Principal authentication

prowler azure --sp-env-auth

# Azure CLI authentication

prowler azure --az-cli-auth

# Compliance assessment

prowler azure --compliance cis\_2.0\_azure

**GCP Assessments:**

# Default credentials

prowler gcp

# Service account authentication

prowler gcp --credentials-file /path/to/service-account.json

# Specific projects

prowler gcp --project-ids project-1 project-2

# Compliance assessment

prowler gcp --compliance cis\_1.2.0\_gcp

**Kubernetes Assessments:**

# Basic Kubernetes assessment

prowler kubernetes

# Using specific kubeconfig file

prowler kubernetes --kubeconfig-file ~/.kube/config

# Scan specific namespaces

prowler kubernetes --namespaces kube-system default production

# Run CIS Kubernetes benchmark

prowler kubernetes --compliance cis\_1.6.0\_kubernetes

# Run NSA/CISA Kubernetes hardening guide

prowler kubernetes --compliance nsa\_cisa\_kubernetes\_hardening\_guidance\_v1.2

# Specific Kubernetes services

prowler kubernetes --services rbac network\_policies pod\_security

# Exclude specific checks

prowler kubernetes --excluded-checks kubernetes\_pod\_security\_context\_non\_root

# Custom output for Kubernetes

prowler kubernetes --output-modes csv json html

**Microsoft 365 Assessments:**

# Basic M365 assessment

prowler m365

# Using specific tenant

prowler m365 --tenant-id "your-tenant-id"

# Service principal authentication

prowler m365 --sp-env-auth

# Browser authentication

prowler m365 --browser-auth

# Run CIS Microsoft 365 benchmark

prowler m365 --compliance cis\_3.0.0\_microsoft365

# Run specific M365 services

prowler m365 --services azuread exchange sharepoint teams

# Specific checks for M365

prowler m365 --checks azuread\_conditional\_access\_policy\_block\_legacy\_authentication

prowler m365 --checks exchange\_dlp\_policy\_enabled

# M365 with custom output

prowler m365 --output-modes csv json html --output-directory ./m365-results/

# Integration with Microsoft 365 Security Center

prowler m365 --send-to-security-center

**7.3 ScoutSuite**

ScoutSuite is a multi-cloud security auditing tool that provides comprehensive security assessments with HTML report interfaces.

**Installation**

**Method 1: pip Installation (Recommended):**

# Install dependencies

sudo apt update

sudo apt install python3 python3-pip python3-venv

# Install ScoutSuite

pip3 install scoutsuite

# Verify installation

scout --help

**Method 2: Git Installation:**

# Clone repository

git clone https://github.com/nccgroup/ScoutSuite

cd ScoutSuite

# Create virtual environment

python3 -m venv venv

source venv/bin/activate

# Install requirements

pip install -r requirements.txt

# Verify installation

python scout.py --help

**Assessment Commands**

**AWS Assessments:**

# Basic AWS scan

scout aws

# Using specific profile

scout aws --profile myprofile

# Specific regions

scout aws --regions us-east-1 eu-west-1

# Custom report name

scout aws --report-name my-aws-assessment

**Azure Assessments:**

# Azure CLI authentication

scout azure --cli

# Service principal authentication

scout azure --sp-env-auth

# Browser authentication

scout azure --browser-auth --tenant-id your-tenant-id

# Specific subscriptions

scout azure --cli --subscription-ids sub1 sub2

**GCP Assessments:**

# Default credentials

scout gcp

# Service account key file

scout gcp --service-account /path/to/service-account.json

# Specific projects

scout gcp --project-id my-project-id

**7.4 Kubernetes Security Tools**

**kube-bench Installation and Usage**

**Installation:**

# Download and install kube-bench

curl -L https://github.com/aquasecurity/kube-bench/releases/latest/download/kube-bench\_linux\_amd64.tar.gz -o kube-bench.tar.gz

tar -xvf kube-bench.tar.gz

sudo mv kube-bench /usr/local/bin/

# Verify installation

kube-bench version

**Assessment Commands:**

# Run CIS Kubernetes benchmark

kube-bench run

# Run specific version

kube-bench run --benchmark cis-1.6

# Generate JSON output

kube-bench run --json

# Run against specific targets

kube-bench run --targets master,node,etcd,policies

**kube-hunter Installation and Usage**

**Installation:**

# Using pip

pip install kube-hunter

# Using Docker

docker pull aquasec/kube-hunter

# Verify installation

kube-hunter --version

**Assessment Commands:**

# Remote scanning

kube-hunter --remote some.node.com

# Network scanning

kube-hunter --cidr 192.168.1.0/24

# Pod assessment (from inside cluster)

kube-hunter --pod

# Generate report

kube-hunter --report json --log none

**Prowler Kubernetes Assessment**

# Kubernetes assessment using Prowler

prowler kubernetes

# With specific kubeconfig

prowler kubernetes --kubeconfig-file ~/.kube/config

# Specific namespaces

prowler kubernetes --namespaces kube-system default

# CIS Kubernetes benchmark

prowler kubernetes --compliance cis\_1.6.0\_kubernetes

**7.5 Microsoft 365 Security Tools**

**Hawk Installation and Usage**

**Installation:**

# Install PowerShell (if not available)

# Ubuntu/Debian

sudo apt update

sudo apt install -y wget apt-transport-https software-properties-common

wget -q https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

sudo apt update

sudo apt install -y powershell

# Install Hawk module

pwsh -Command "Install-Module -Name HAWK -Force"

**Assessment Commands:**

# Connect to Microsoft 365

Connect-EXO

# Run comprehensive assessment

Start-HawkTenantInvestigation

# Investigate specific user

Start-HawkUserInvestigation -UserPrincipalName user@domain.com

# Search for specific activities

Search-HawkTenantActivityByIP -IPAddress "192.168.1.100"

**ROADtools Installation and Usage**

**Installation:**

# Install ROADtools

pip install roadtools

# Verify installation

roadrecon --help

**Assessment Commands:**

# Authenticate and gather data

roadrecon auth --device-code

roadrecon gather --config roadtools.conf

# Generate GUI interface

roadrecon gui

# Dump specific information

roadrecon dump --database roadrecon.db --output-dir ./dump/

**Microsoft 365 Defender Integration**

**PowerShell Assessment Commands:**

# Connect to Security & Compliance Center

Connect-IPPSSession

# Get security configuration

Get-OrganizationConfig

Get-DlpPolicy

Get-RetentionPolicy

Get-CASMailboxPlan

# Audit configuration

Get-AdminAuditLogConfig

Get-MailboxAuditBypassAssociation

# Get security alerts

Get-ProtectionAlert

**8. Authentication Configuration**

**8.1 AWS Authentication**

**Option 1: AWS CLI Configuration**

# Install AWS CLI

sudo apt install awscli

# Configure credentials

aws configure

**Credentials File Configuration:**

# ~/.aws/credentials

[default]

aws\_access\_key\_id = AKIAIOSFODNN7EXAMPLE

aws\_secret\_access\_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

region = eu-west-1

[assessment-profile]

aws\_access\_key\_id = AKIAI44QH8DHBEXAMPLE

aws\_secret\_access\_key = je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY

region = us-east-1

**Option 2: Environment Variables**

export AWS\_ACCESS\_KEY\_ID="AKIAIOSFODNN7EXAMPLE"

export AWS\_SECRET\_ACCESS\_KEY="wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"

export AWS\_DEFAULT\_REGION="eu-west-1"

**8.2 Azure Authentication**

**Option 1: Azure CLI**

# Install Azure CLI

curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash

# Authenticate

az login

# Set default subscription

az account set --subscription "your-subscription-id"

**Option 2: Service Principal**

# Create service principal

az ad sp create-for-rbac --name "security-assessment-sp" --role "Reader"

# Set environment variables

export AZURE\_CLIENT\_ID="your-client-id"

export AZURE\_CLIENT\_SECRET="your-client-secret"

export AZURE\_TENANT\_ID="your-tenant-id"

export AZURE\_SUBSCRIPTION\_ID="your-subscription-id"

**8.3 GCP Authentication**

**Option 1: gcloud CLI**

# Install gcloud CLI

curl https://sdk.cloud.google.com | bash

exec -l $SHELL

# Authenticate

gcloud auth login

gcloud auth application-default login

# Set default project

gcloud config set project your-project-id

**Option 2: Service Account**

# Set service account credentials

export GOOGLE\_APPLICATION\_CREDENTIALS="/path/to/service-account.json"

**8.4 Kubernetes Authentication**

**Powerpipe/Steampipe Kubernetes Authentication**

# Using default kubeconfig

steampipe plugin install kubernetes

# Using specific kubeconfig file

export KUBECONFIG=/path/to/specific/kubeconfig

steampipe plugin install kubernetes

# Multiple cluster configurations

cat > ~/.steampipe/config/kubernetes.spc << EOF

connection "k8s\_prod" {

plugin = "kubernetes"

config\_path = "/path/to/prod-kubeconfig"

config\_context = "prod-cluster"

}

connection "k8s\_dev" {

plugin = "kubernetes"

config\_path = "/path/to/dev-kubeconfig"

config\_context = "dev-cluster"

}

EOF

# Test connection

steampipe query "select name, namespace from kubernetes\_pod limit 5" --connection k8s\_prod

**Prowler Kubernetes Authentication**

# Using default kubeconfig

prowler kubernetes

# Using specific kubeconfig file

prowler kubernetes --kubeconfig-file /path/to/kubeconfig

# Using specific context

export KUBECONFIG=/path/to/kubeconfig

kubectl config use-context target-cluster

prowler kubernetes

# In-cluster assessment (from pod with service account)

prowler kubernetes --in-cluster

**Option 1: kubectl Configuration**

# Install kubectl

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

# Configure cluster access

kubectl config set-cluster my-cluster --server=https://k8s-cluster:6443

kubectl config set-credentials my-user --token=bearer-token

kubectl config set-context my-context --cluster=my-cluster --user=my-user

kubectl config use-context my-context

# Verify access

kubectl cluster-info

kubectl auth can-i --list

**Option 2: Service Account Token**

# Create service account for assessment

kubectl create serviceaccount security-assessor

kubectl create clusterrolebinding security-assessor --clusterrole=cluster-reader --serviceaccount=default:security-assessor

# Get service account token

kubectl get secret $(kubectl get serviceaccount security-assessor -o jsonpath='{.secrets[0].name}') -o jsonpath='{.data.token}' | base64 --decode

**Option 3: Certificate-based Authentication**

# Generate client certificate

openssl genrsa -out assessment.key 2048

openssl req -new -key assessment.key -out assessment.csr -subj "/CN=security-assessor/O=security"

# Configure kubeconfig with certificate

kubectl config set-credentials security-assessor --client-certificate=assessment.crt --client-key=assessment.key

**8.5 Microsoft 365 Authentication**

**Powerpipe/Steampipe M365 Authentication**

**Azure AD Configuration:**

# Create connection configuration

cat > ~/.steampipe/config/azuread.spc << EOF

connection "azuread" {

plugin = "azuread"

tenant\_id = "your-tenant-id"

client\_id = "your-client-id"

client\_secret = "your-client-secret"

# Alternative: certificate-based authentication

# certificate\_path = "/path/to/certificate.pem"

# certificate\_password = "cert-password"

}

EOF

# Test Azure AD connection

steampipe query "select display\_name, user\_principal\_name from azuread\_user limit 5"

**Microsoft 365 Configuration:**

# Create M365 connection configuration

cat > ~/.steampipe/config/microsoft365.spc << EOF

connection "microsoft365" {

plugin = "microsoft365"

tenant\_id = "your-tenant-id"

client\_id = "your-client-id"

client\_secret = "your-client-secret"

# Additional scopes for comprehensive access

scopes = [

"https://graph.microsoft.com/.default",

"https://outlook.office365.com/.default"

]

}

EOF

# Test M365 connection

steampipe query "select id, display\_name from microsoft365\_security\_score\_control\_profile limit 5"

**Prowler M365 Authentication**

**Service Principal Authentication:**

# Set environment variables for Prowler

export AZURE\_CLIENT\_ID="your-client-id"

export AZURE\_CLIENT\_SECRET="your-client-secret"

export AZURE\_TENANT\_ID="your-tenant-id"

# Run Prowler with service principal

prowler m365 --sp-env-auth

**Browser Authentication:**

# Interactive browser authentication

prowler m365 --browser-auth --tenant-id "your-tenant-id"

# With specific credentials file

prowler m365 --credentials-file /path/to/credentials.json

**Option 1: Azure AD Application Registration**

# Create app registration for assessment

$app = New-AzADApplication -DisplayName "Security Assessment App" -ReplyUrls "http://localhost"

$sp = New-AzADServicePrincipal -ApplicationId $app.ApplicationId

# Required API permissions for comprehensive assessment

$requiredPermissions = @(

"Directory.Read.All",

"AuditLog.Read.All",

"SecurityEvents.Read.All",

"Policy.Read.All",

"Application.Read.All",

"User.Read.All",

"Group.Read.All"

)

# Set API permissions (requires admin consent)

foreach ($permission in $requiredPermissions) {

Add-AzADAppPermission -ObjectId $app.ObjectId -ApiId "00000003-0000-0000-c000-000000000000" -PermissionId $permission

}

**Option 2: PowerShell Modules Authentication**

# Install required modules for Powerpipe/Prowler integration

Install-Module -Name ExchangeOnlineManagement -Force

Install-Module -Name Microsoft.Online.SharePoint.PowerShell -Force

Install-Module -Name MSOnline -Force

Install-Module -Name AzureAD -Force

Install-Module -Name Microsoft.Graph -Force

# Connect to services

Connect-ExchangeOnline -UserPrincipalName admin@domain.com

Connect-SPOService -Url https://domain-admin.sharepoint.com

Connect-MsolService

Connect-AzureAD

Connect-MgGraph -Scopes "Directory.Read.All","AuditLog.Read.All"

# Export credentials for tool consumption

$creds = Get-Credential

$creds | Export-Clixml -Path "m365-creds.xml"

**Option 3: Modern Authentication with Device Code**

# Using Azure CLI for token acquisition

az login --use-device-code --tenant "your-tenant-id"

# Get access tokens for different services

az account get-access-token --resource https://graph.microsoft.com --query accessToken -o tsv

az account get-access-token --resource https://outlook.office365.com --query accessToken -o tsv

# Set tokens for tool consumption

export GRAPH\_ACCESS\_TOKEN=$(az account get-access-token --resource https://graph.microsoft.com --query accessToken -o tsv)

export EXCHANGE\_ACCESS\_TOKEN=$(az account get-access-token --resource https://outlook.office365.com --query accessToken -o tsv)

**Option 4: Certificate-based Authentication**

# Create self-signed certificate for app authentication

$cert = New-SelfSignedCertificate -Subject "CN=M365Assessment" -CertStoreLocation "Cert:\CurrentUser\My" -KeySpec KeyExchange -KeyExportPolicy Exportable

# Export certificate for Powerpipe/Prowler

Export-Certificate -Cert $cert -FilePath "M365Assessment.cer"

Export-PfxCertificate -Cert $cert -FilePath "M365Assessment.pfx" -Password (ConvertTo-SecureString -String "password" -AsPlainText -Force)

# Configure certificate-based authentication

cat > ~/.steampipe/config/azuread-cert.spc << EOF

connection "azuread\_cert" {

plugin = "azuread"

tenant\_id = "your-tenant-id"

client\_id = "your-client-id"

certificate\_path = "/path/to/M365Assessment.pfx"

certificate\_password = "password"

}

EOF

**Option 5: Managed Identity (Azure VM)**

# For assessments running on Azure VMs with managed identity

export AZURE\_USE\_MSI=true

# Run Prowler with managed identity

prowler m365 --msi

# Configure Steampipe for managed identity

cat > ~/.steampipe/config/azuread-msi.spc << EOF

connection "azuread\_msi" {

plugin = "azuread"

use\_msi = true

}

EOF

**9. Assessment Execution**

**9.1 Complete Assessment Workflow**

**Multi-Tool AWS Assessment**

# Run comprehensive AWS assessment

prowler aws --compliance cis\_2.0\_aws

scout aws --profile default

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140

**Multi-Tool Azure Assessment**

# Run comprehensive Azure assessment

prowler azure --az-cli-auth --compliance cis\_2.0\_azure

scout azure --cli

powerpipe benchmark run azure\_compliance.benchmark.cis\_v140

**Multi-Tool GCP Assessment**

# Run comprehensive GCP assessment

prowler gcp --compliance cis\_1.2.0\_gcp

scout gcp --project-id your-project

powerpipe benchmark run gcp\_compliance.benchmark.cis\_v120

**Kubernetes Security Assessment**

# Comprehensive Kubernetes assessment with multiple tools

powerpipe benchmark run kubernetes\_compliance.benchmark.cis\_v120

prowler kubernetes --compliance cis\_1.6.0\_kubernetes

kube-bench run --json > kube-bench-results.json

kube-hunter --remote cluster-endpoint.com --report json

**Microsoft 365 Security Assessment**

# Comprehensive M365 assessment

powerpipe benchmark run microsoft365\_compliance.benchmark.cis\_v300

powerpipe benchmark run azuread\_compliance.benchmark.cis\_v100

prowler m365 --compliance cis\_3.0.0\_microsoft365

# PowerShell-based M365 assessment

Start-HawkTenantInvestigation

Get-OrganizationConfig | Export-Csv org-config.csv

Get-DlpPolicy | Export-Csv dlp-policies.csv

Get-ConditionalAccessPolicy | Export-Csv ca-policies.csv

**Multi-Platform Assessment Automation**

#!/bin/bash

# Comprehensive multi-platform security assessment script

echo "Starting comprehensive security assessment..."

# Create results directories

mkdir -p ./results/{aws,azure,gcp,kubernetes,m365}

# Cloud platform assessments

echo "Assessing AWS..."

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140 --output json > ./results/aws/powerpipe-cis.json

prowler aws --compliance cis\_2.0\_aws --output-modes json --output-directory ./results/aws/

echo "Assessing Azure..."

powerpipe benchmark run azure\_compliance.benchmark.cis\_v140 --output json > ./results/azure/powerpipe-cis.json

prowler azure --az-cli-auth --compliance cis\_2.0\_azure --output-modes json --output-directory ./results/azure/

echo "Assessing GCP..."

powerpipe benchmark run gcp\_compliance.benchmark.cis\_v120 --output json > ./results/gcp/powerpipe-cis.json

prowler gcp --compliance cis\_1.2.0\_gcp --output-modes json --output-directory ./results/gcp/

echo "Assessing Kubernetes..."

powerpipe benchmark run kubernetes\_compliance.benchmark.cis\_v120 --output json > ./results/kubernetes/powerpipe-cis.json

prowler kubernetes --compliance cis\_1.6.0\_kubernetes --output-modes json --output-directory ./results/kubernetes/

kube-bench run --json > ./results/kubernetes/kube-bench.json

echo "Assessing Microsoft 365..."

powerpipe benchmark run microsoft365\_compliance.benchmark.cis\_v300 --output json > ./results/m365/powerpipe-cis.json

prowler m365 --compliance cis\_3.0.0\_microsoft365 --output-modes json --output-directory ./results/m365/

echo "Assessment complete. Results saved to ./results/"

echo "Generating consolidated report..."

# Generate consolidated dashboard

powerpipe server --port 9033 &

echo "Dashboard available at http://localhost:9033"

**9.2 Compliance Framework Testing**

**CIS Benchmarks**

# AWS CIS v2.0

prowler aws --compliance cis\_2.0\_aws

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140

# Azure CIS v2.0

prowler azure --compliance cis\_2.0\_azure

powerpipe benchmark run azure\_compliance.benchmark.cis\_v140

# GCP CIS v1.2.0

prowler gcp --compliance cis\_1.2.0\_gcp

powerpipe benchmark run gcp\_compliance.benchmark.cis\_v120

**NIST Frameworks**

# NIST 800-53 Rev 5

prowler aws --compliance nist\_800\_53\_revision\_5\_aws

prowler azure --compliance nist\_cybersecurity\_framework\_v1.1\_azure

# NIST CSF

prowler aws --compliance nist\_csf\_1.1\_aws

**9.2 Compliance Framework Testing**

**CIS Benchmarks**

# Cloud Platform CIS Benchmarks

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140 --output json

prowler aws --compliance cis\_2.0\_aws

powerpipe benchmark run azure\_compliance.benchmark.cis\_v140 --output json

prowler azure --compliance cis\_2.0\_azure

powerpipe benchmark run gcp\_compliance.benchmark.cis\_v120 --output json

prowler gcp --compliance cis\_1.2.0\_gcp

# Kubernetes CIS Benchmarks

powerpipe benchmark run kubernetes\_compliance.benchmark.cis\_v120 --output json

prowler kubernetes --compliance cis\_1.6.0\_kubernetes

kube-bench run --benchmark cis-1.6

# Microsoft 365 CIS Benchmarks

powerpipe benchmark run microsoft365\_compliance.benchmark.cis\_v300 --output json

powerpipe benchmark run azuread\_compliance.benchmark.cis\_v100 --output json

prowler m365 --compliance cis\_3.0.0\_microsoft365

**NIST Frameworks**

# NIST 800-53 Rev 5 (Cloud Platforms)

powerpipe benchmark run aws\_compliance.benchmark.nist\_800\_53\_rev\_5 --output json

prowler aws --compliance nist\_800\_53\_revision\_5\_aws

prowerpipe benchmark run azure\_compliance.benchmark.nist\_800\_53\_rev\_5 --output json

prowler azure --compliance nist\_cybersecurity\_framework\_v1.1\_azure

# NIST Cybersecurity Framework

powerpipe benchmark run aws\_compliance.benchmark.nist\_csf --output json

prowler aws --compliance nist\_csf\_1.1\_aws

powerpipe benchmark run kubernetes\_compliance.benchmark.nist\_800\_53\_rev\_5 --output json

# NIST 800-171 (Controlled Unclassified Information)

prowler aws --compliance nist\_800\_171\_revision\_2\_aws

powerpipe benchmark run aws\_compliance.benchmark.nist\_800\_171\_rev\_2 --output json

# Kubernetes NIST Guidelines

powerpipe benchmark run kubernetes\_compliance.benchmark.nsa\_cisa\_v1 --output json

prowler kubernetes --compliance nsa\_cisa\_kubernetes\_hardening\_guidance\_v1.2

# Microsoft 365 NIST Alignment

powerpipe benchmark run microsoft365\_compliance.benchmark.nist\_csf --output json

**PCI-DSS Compliance**

# PCI-DSS v3.2.1

prowler aws --compliance pci\_3.2.1\_aws

powerpipe benchmark run aws\_compliance.benchmark.pci\_dss\_v321

# Azure PCI-DSS

prowler azure --compliance pci\_dss\_v3.2.1\_azure

**GDPR Compliance**

# GDPR assessments

prowler aws --compliance gdpr\_aws

prowler azure --compliance gdpr\_azure

# M365 GDPR compliance check

```powershell

Get-DlpPolicy | Where-Object {$\_.Name -like "\*GDPR\*"}

Get-RetentionPolicy | Where-Object {$\_.Name -like "\*GDPR\*"}

Get-InformationProtectionPolicy

**SOC 2 and Industry-Specific**

# SOC 2 Type II

prowler aws --compliance soc2\_aws

# HIPAA

prowler aws --compliance hipaa\_aws

prowler azure --compliance hipaa\_azure

# ISO 27001

prowler aws --compliance iso27001\_2013\_aws

# FedRAMP

prowler aws --compliance fedramp\_low\_revision\_4\_aws

prowler aws --compliance fedramp\_moderate\_revision\_4\_aws

**Kubernetes-Specific Compliance**

# Pod Security Standards

kubectl get psp --all-namespaces

kubectl get securitycontextconstraints (OpenShift)

# Network Policy Assessment

kubectl get networkpolicy --all-namespaces

# RBAC Assessment

kubectl get clusterroles

kubectl get clusterrolebindings

kubectl auth can-i --list --as=system:serviceaccount:kube-system:default

**Microsoft 365 Compliance Assessment**

# Data Loss Prevention

Get-DlpPolicy

Get-DlpRule

Get-DlpSensitiveInformationType

# Retention and Compliance

Get-RetentionPolicy

Get-ComplianceSearch

Get-CaseHoldPolicy

# Conditional Access and Identity

Get-ConditionalAccessPolicy

Get-AzureADDirectorySetting

Get-AzureADPasswordPolicy

# Security and Threat Protection

Get-SafeAttachmentPolicy

Get-SafeLinksPolicy

Get-AntiPhishPolicy

Get-HostedContentFilterPolicy

**9.3 Output and Reporting Options**

# Multiple output formats

prowler aws --output-modes csv json-ocsf html

# Custom output directory

prowler aws --output-directory /path/to/reports

# Integration with AWS Security Hub

prowler aws --security-hub --status FAIL

# Interactive dashboard

powerpipe server --port 9033

**10. Quick Reference Guide**

**Tool Capability Matrix**

| **Tool** | **AWS** | **Azure** | **GCP** | **Kubernetes** | **M365** | **Compliance** | **Dashboard** | **API** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Powerpipe | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Prowler | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✓ |
| ScoutSuite | ✓ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| kube-bench | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| kube-hunter | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ |
| Hawk | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ |
| ROADtools | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ |

**Common Command Patterns**

# List available options across all platforms

prowler aws --list-checks

prowler azure --list-services

prowler gcp --list-compliance

prowler kubernetes --list-checks

prowler m365 --list-services

# Powerpipe queries for different platforms

powerpipe query "select name, namespace from kubernetes\_pod where namespace = 'kube-system'"

powerpipe query "select display\_name, account\_enabled from azuread\_user limit 10"

powerpipe query "select name, state from microsoft365\_security\_score\_control\_profile"

# Generate specific outputs

scout aws --report-name "assessment-$(date +%Y%m%d)"

powerpipe benchmark run aws\_compliance.benchmark.cis\_v140 --output json

kube-bench run --json > k8s-assessment.json

prowler m365 --output-modes json html --output-directory ./m365-results/

# Filter and focus assessments

prowler aws --services s3 iam --excluded-checks s3\_bucket\_logging\_enabled

prowler kubernetes --namespaces production staging --services rbac network\_policies

prowler m365 --services azuread exchange --checks azuread\_conditional\_access\_policy\_block\_legacy\_authentication

scout azure --cli --subscription-ids specific-sub-id

kube-hunter --cidr 10.0.0.0/8 --report json

# Powerpipe multi-connection assessments

powerpipe benchmark run kubernetes\_compliance.benchmark.cis\_v120 --connection k8s\_prod

powerpipe benchmark run microsoft365\_compliance.benchmark.cis\_v300 --connection m365\_tenant

**M365 focused assessments using PowerShell:**

# PowerShell patterns for M365 integration with tools

Get-DlpPolicy | Where-Object {$\_.State -eq "Enabled"} | Export-Csv dlp-enabled.csv

Get-ConditionalAccessPolicy | Export-Csv ca-policies.csv -NoTypeInformation

Start-HawkUserInvestigation -UserPrincipalName suspicious-user@domain.com

Get-AzureADDirectorySetting | ConvertTo-Json | Out-File azuread-settings.json

**Cross-platform assessment automation:**

#!/bin/bash

# Multi-platform assessment script with Powerpipe and Prowler

echo "Starting comprehensive security assessment..."

# Function to run Powerpipe benchmarks

run\_powerpipe\_assessment() {

local platform=$1

local benchmark=$2

local output\_file=$3

echo "Running Powerpipe $platform assessment..."

powerpipe benchmark run $benchmark --output json > $output\_file

}

# Function to run Prowler assessments

run\_prowler\_assessment() {

local platform=$1

local compliance=$2

local output\_dir=$3

echo "Running Prowler $platform assessment..."

prowler $platform --compliance $compliance --output-modes json html --output-directory $output\_dir

}

# Create results structure

mkdir -p ./results/{aws,azure,gcp,kubernetes,m365}/{powerpipe,prowler}

# Cloud platforms

run\_powerpipe\_assessment "AWS" "aws\_compliance.benchmark.cis\_v140" "./results/aws/powerpipe/cis-benchmark.json"

run\_prowler\_assessment "aws" "cis\_2.0\_aws" "./results/aws/prowler/"

run\_powerpipe\_assessment "Azure" "azure\_compliance.benchmark.cis\_v140" "./results/azure/powerpipe/cis-benchmark.json"

run\_prowler\_assessment "azure --az-cli-auth" "cis\_2.0\_azure" "./results/azure/prowler/"

run\_powerpipe\_assessment "GCP" "gcp\_compliance.benchmark.cis\_v120" "./results/gcp/powerpipe/cis-benchmark.json"

run\_prowler\_assessment "gcp" "cis\_1.2.0\_gcp" "./results/gcp/prowler/"

# Kubernetes

run\_powerpipe\_assessment "Kubernetes" "kubernetes\_compliance.benchmark.cis\_v120" "./results/kubernetes/powerpipe/cis-benchmark.json"

run\_prowler\_assessment "kubernetes" "cis\_1.6.0\_kubernetes" "./results/kubernetes/prowler/"

kube-bench run --json > ./results/kubernetes/kube-bench.json

# Microsoft 365

run\_powerpipe\_assessment "M365" "microsoft365\_compliance.benchmark.cis\_v300" "./results/m365/powerpipe/cis-benchmark.json"

run\_prowler\_assessment "m365 --sp-env-auth" "cis\_3.0.0\_microsoft365" "./results/m365/prowler/"

echo "Assessment complete. Results saved to ./results/"

echo "Starting Powerpipe dashboard..."

powerpipe server --port 9033 &

echo "Dashboard available at http://localhost:9033"

**11. Additional Resources**

**Documentation Links**

**Multi-Cloud Tools:**

* **Powerpipe Documentation:** <https://powerpipe.io/docs>
* **Powerpipe Hub:** [https://hub.powerpipe.io](https://hub.powerpipe.io/)
* **Prowler Documentation:** [https://docs.prowler.com](https://docs.prowler.com/)
* **Prowler GitHub:** <https://github.com/prowler-cloud/prowler>
* **ScoutSuite GitHub:** <https://github.com/nccgroup/ScoutSuite>
* **ScoutSuite Wiki:** <https://github.com/nccgroup/ScoutSuite/wiki>

**Kubernetes Security Tools:**

* **kube-bench GitHub:** <https://github.com/aquasecurity/kube-bench>
* **kube-hunter GitHub:** <https://github.com/aquasecurity/kube-hunter>
* **Falco Documentation:** <https://falco.org/docs/>
* **OPA Gatekeeper:** <https://open-policy-agent.github.io/gatekeeper/>
* **Polaris Documentation:** <https://polaris.docs.fairwinds.com/>

**Microsoft 365 Security Tools:**

* **Hawk GitHub:** <https://github.com/T0pCyber/hawk>
* **ROADtools GitHub:** <https://github.com/dirkjanm/ROADtools>
* **SPARROW GitHub:** <https://github.com/cisagov/SPARROW>
* **Microsoft 365 Security Center:** [https://security.microsoft.com](https://security.microsoft.com/)

**Framework References**

* **CIS Benchmarks:** <https://www.cisecurity.org/cis-benchmarks>
* **OWASP Cloud Top 10:** <https://owasp.org/www-project-cloud-top-10/>
* **CSA Cloud Controls Matrix:** <https://cloudsecurityalliance.org/>
* **NIST Cybersecurity Framework:** <https://www.nist.gov/cyberframework>
* **NIST 800-53:** <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf>
* **PCI Security Standards:** <https://www.pcisecuritystandards.org/>
* **GDPR Official Text:** <https://gdpr-info.eu/>

**Platform-Specific Security Resources**

**Kubernetes Security:**

* **Kubernetes Security Documentation:** <https://kubernetes.io/docs/concepts/security/>
* **Pod Security Standards:** <https://kubernetes.io/docs/concepts/security/pod-security-standards/>
* **Network Policies:** <https://kubernetes.io/docs/concepts/services-networking/network-policies/>

**Microsoft 365 Security:**

* **Microsoft 365 Security Documentation:** <https://docs.microsoft.com/en-us/microsoft-365/security/>
* **Azure AD Security:** <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/concept-fundamentals-security-defaults>
* **Microsoft Secure Score:** <https://docs.microsoft.com/en-us/microsoft-365/security/defender/microsoft-secure-score>

**12. Appendices**

**Appendix A: Troubleshooting Common Issues**

**Authentication Errors:**

* Verify credentials and permissions across all platforms
* Check region configurations and API access limits
* Validate service account tokens and certificate expiration
* Ensure proper RBAC configurations for Kubernetes
* Verify Microsoft 365 admin consent for required permissions

**Tool Installation Issues:**

* Ensure Python version compatibility (3.8+ recommended)
* Check network connectivity for downloads and API access
* Verify system dependencies (kubectl, PowerShell modules)
* Resolve Docker permission issues for containerized tools
* Check PowerShell execution policy for M365 modules

**Assessment Performance:**

* Optimize tool configurations for large environments
* Use region-specific and namespace-specific assessments
* Implement parallel execution strategies
* Configure appropriate API rate limits and timeouts
* Use service mesh policies to avoid assessment interference

**Platform-Specific Issues:**

* **Kubernetes:** Check cluster connectivity and context switching
* **M365:** Verify tenant permissions and conditional access policies
* **Multi-cloud:** Ensure proper credential isolation and profile management