

PowerShell for Security Analysts

SEC205

The PowerShell a security analyst uses: triage a Windows host, investigate processes, files, the registry and event logs, read attacker PowerShell, hunt across many hosts, and build the tools that make it repeatable. Taught on a real intrusion's evidence, with every command run and every answer checked.

Modules	22
Lessons	139
Phases	5
Free preview	PowerShell in Security Operations (8 lessons)

The method

Every result is real. Each lesson runs its commands against evidence files from Northgate's hosts and tenant, and every output in the course came from running that command. Download a module's evidence and the same commands give the same answers on any machine with PowerShell 7.4 or later.

Windows PowerShell and PowerShell 7. The course is written for PowerShell 7.6 and teaches Windows PowerShell 5.1 wherever it behaves differently, because it is on every Windows machine you will investigate.

Check every answer. An empty result, a total, an unsigned file and an AI-written script are all things the course teaches you to verify before anyone relies on them.

Full syllabus

Orientation

PowerShell in Security Operations FREE PREVIEW

Module 0 of PowerShell for Security Analysts: why defenders use PowerShell, Windows PowerShell 5.1 and PowerShell 7, how PowerShell is built, the console, Terminal and VS Code, cmdlets, functions, scripts and modules, finding your way around, an administrator's tool and an attacker's tool, and...

1. 0.1 Why Defenders Use PowerShell
2. 0.2 Windows PowerShell 5.1 and PowerShell 7
3. 0.3 How PowerShell Is Built
4. 0.4 Console, Terminal and VS Code
5. 0.5 Cmdlets, Functions, Scripts and Modules
6. 0.6 Finding Your Way Around
7. 0.7 An Administrator's Tool and an Attacker's Tool
8. 0.8 Building Your Analyst Workstation

Foundations

The Syntax Analysts Need

Module 1 of PowerShell for Security Analysts: objects and the pipeline, filtering and comparison, logic, wildcards and regular expressions, variables and collections, loops, functions and error handling, and reducing endpoint data to an answer.

1. 1.1 Objects, Properties and Methods
2. 1.2 The Pipeline
3. 1.3 Where-Object and Comparison Operators
4. 1.4 Logical Operators, Wildcards and Regex
5. 1.5 Variables, Arrays and Hashtables
6. 1.6 Loops and Conditions
7. 1.7 Functions and Error Handling
8. 1.8 Reducing Endpoint Data to an Answer
9. Module Summary and Knowledge Check

Working With Security Data

Module 2 of PowerShell for Security Analysts: CSV, JSON and REST responses, XML, plain text and Select-String, CliXml, sorting and grouping, calculated properties, and comparing datasets.

1. 2.1 CSV
2. 2.2 JSON and REST Responses
3. 2.3 XML
4. 2.4 Text and Select-String
5. 2.5 CliXml and PowerShell Objects
6. 2.6 Sorting, Grouping and Unique Values
7. 2.7 Calculated Properties
8. 2.8 Comparing Datasets
9. Module Summary and Knowledge Check

Endpoint Investigation

Host Triage and Accounts

Module 3 of PowerShell for Security Analysts: what the machine is, its patches, boot time and uptime, installed software, services and drivers, local users, groups and administrators, recently changed accounts, and Active Directory accounts.

1. 3.1 What Is This Machine
2. 3.2 Patches, Boot Time and Uptime
3. 3.3 Installed Software
4. 3.4 Services and Drivers
5. 3.5 Local Users and Groups
6. 3.6 Local Administrators and Privileged Accounts
7. 3.7 Recently Created and Re-enabled Accounts
8. 3.8 Active Directory Accounts
9. Module Summary and Knowledge Check

Process and Network Investigation

Module 4 of PowerShell for Security Analysts: Get-Process and Win32_Process, parent, path, command line and user, process trees, LOLBins and suspicious chains, connections and listeners, the DNS cache, and correlating processes with the addresses they reach.

1. 4.1 Get-Process and Win32_Process
2. 4.2 Parent, Path, Command Line and User
3. 4.3 Process Trees
4. 4.4 LOLBins and Suspicious Chains
5. 4.5 Connections and Listeners
6. 4.6 DNS Cache and Resolution
7. 4.7 From PID to Remote Address
8. 4.8 A Process and Network Correlator
9. Module Summary and Knowledge Check

Files and the Registry

Module 5 of PowerShell for Security Analysts: finding files, timestamps, hidden and recent files, hashing, signatures and PE files, alternate data streams and the mark of the web, the registry provider, persistence and configuration keys, and USB, RDP and recent-activity keys.

1. 5.1 Finding Files
2. 5.2 Timestamps, Hidden and Recent Files
3. 5.3 Hashing
4. 5.4 Signatures and PE Files
5. 5.5 Alternate Data Streams and the Mark of the Web
6. 5.6 The Registry Provider
7. 5.7 Persistence and Configuration Keys
8. 5.8 USB, RDP and Recent-Activity Keys
9. Module Summary and Knowledge Check

Logs and Detection

Windows Event Logs

Module 6 of PowerShell for Security Analysts: logs, providers and event IDs, Get-WinEvent and FilterHashtable, XPath and event XML, logon, process and privilege events, services, tasks and account changes, the Sysmon, Defender, RDP and WMI logs, and a security timeline tool.

1. 6.1 Logs, Providers and Event IDs
2. 6.2 Get-WinEvent and FilterHashtable
3. 6.3 XPath and Event XML
4. 6.4 Logon Events
5. 6.5 Process and Privilege Events
6. 6.6 Services, Tasks and Account Changes
7. 6.7 Sysmon, Defender, RDP and WMI Logs
8. 6.8 A Security Timeline Tool
9. Module Summary and Knowledge Check

PowerShell Logging and PowerShell Attacks

Module 7 of PowerShell for Security Analysts: script block, module and transcription logging, events 4103, 4104, 400 and 800, PSReadLine history, encoded commands, download cradles, obfuscation and safe deobfuscation, AMSI tampering as evidence, and execution policy and Constrained Language Mode.

1. 7.1 PowerShell's Own Logs
2. 7.2 Reading Script Block Logs
3. 7.3 PSReadLine History and Transcripts
4. 7.4 Encoded Commands
5. 7.5 Invoke-Expression and Download Cradles
6. 7.6 Obfuscation and Safe Deobfuscation
7. 7.7 AMSI and Its Bypasses, as Evidence
8. 7.8 Execution Policy and Constrained Language Mode
9. Module Summary and Knowledge Check

Persistence and Defender

Module 8 of PowerShell for Security Analysts: scheduled tasks, services and Run keys across a fleet, startup folders and PowerShell profiles, WMI event subscriptions, DLL search-order abuse, Defender status and signatures, threats, exclusions and tampering, and a persistence hunter.

1. 8.1 Scheduled Tasks
2. 8.2 Services and Run Keys Across the Fleet
3. 8.3 Startup Folders and PowerShell Profiles
4. 8.4 WMI Event Subscriptions
5. 8.5 DLL Search-Order Abuse
6. 8.6 Defender Status and Signatures
7. 8.7 Threats, Exclusions and Tampering
8. 8.8 A Persistence Hunter
9. Module Summary and Knowledge Check

Response, Hunting and Automation

Triage, Collection and IOC Hunting

Module 9 of PowerShell for Security Analysts: a repeatable triage method, the host triage collector, evidence integrity with hashing and case metadata, the incident evidence collector, hunting one indicator and many with an IOC scanner, and remote collection with WinRM, credentials and JEA.

1. 9.1 A Repeatable Triage Method
2. 9.2 The Host Triage Collector
3. 9.3 Evidence Integrity, Hashing and Case Metadata
4. 9.4 The Incident Evidence Collector
5. 9.5 Hunting One Indicator
6. 9.6 Multi-Indicator Hunting and the IOC Scanner
7. 9.7 Remoting, WinRM and Credentials
8. 9.8 At Scale, with JEA
9. Module Summary and Knowledge Check

APIs and Threat Intelligence

Module 10 of PowerShell for Security Analysts: REST calls with headers and authentication, JSON paging and rate limits, reputation lookups, API keys as secrets, Microsoft Graph PowerShell, Entra ID sign-in data, the Defender XDR and Sentinel APIs, and an enriched report from an alert.

1. 10.1 REST, Headers and Authentication
2. 10.2 JSON, Paging and Rate Limits
3. 10.3 Reputation Lookups
4. 10.4 Secrets for API Keys
5. 10.5 Microsoft Graph PowerShell
6. 10.6 Entra ID and Sign-in Data
7. 10.7 Defender XDR and Sentinel APIs
8. 10.8 From Alert to Enriched Report
9. Module Summary and Knowledge Check

Building Analyst Tools

Module 11 of PowerShell for Security Analysts: from one-liners to functions, parameters and validation, logging and help, configuration and credentials, risk scoring, modules and the analyst toolkit, signing, JEA, WDAC and AppLocker, and checking a script you did not write.

1. 11.1 From One-liners to Functions
2. 11.2 Parameters, Validation and Pipeline Input
3. 11.3 Logging, Verbose Output and Help
4. 11.4 Configuration and Credentials
5. 11.5 Risk Scoring
6. 11.6 Modules and the Analyst Toolkit
7. 11.7 Signing, JEA, WDAC and AppLocker
8. 11.8 Checking a Script You Did Not Write
9. Module Summary and Knowledge Check

Resources

Analyst Cheatsheets

One sheet per module of PowerShell for Security Analysts, each entry a question, a command that answers it against the course evidence with its real output, what to read and what the answer does not prove.

1. PowerShell in Security Operations
2. Syntax Analysts Need
3. Working with Security Data
4. Host Triage and Accounts
5. Process and Network
6. Files and Registry
7. Windows Event Logs
8. PowerShell Logging and Attacks
9. Persistence and Defender
10. Triage, Collection and Hunting
11. APIs and Threat Intelligence
12. Building Analyst Tools

Investigation Cookbooks

Procedures for the three problems every PowerShell analyst meets: a command that returns nothing, a collection that is incomplete, and an evidence file nobody has read before.

1. A Command Returns Nothing
2. A Collection Is Incomplete
3. Reading an Unfamiliar Evidence File

Windows Lab Setup

The course evidence, which runs every lesson on any machine with PowerShell 7, and a Windows virtual machine of your own for the live commands, with the logging and remoting the course relies on.

1. The Course Evidence
2. A Windows Lab of Your Own

Case Walkthroughs

Six worked investigations on the course evidence, each from the question handed to you to the finding written honestly, with every command run and every wrong turn shown.

1. From Word to the DLL
2. The Count That Missed Two Hosts
3. The Filter That Found Microsoft
4. The Toolkit That Answered Differently
5. The Account Nobody Created
6. One Timeline from Two Logs

Hunting Playbooks

Six hunting patterns in PowerShell, each run on the course evidence: rare across the fleet, rare configuration, new since the baseline, Office starting a shell, paired events in sequence, and start then connect.

1. Rare Across the Fleet
2. Rare Configuration
3. New Since the Baseline
4. Office Starting a Shell
5. Paired Events in Sequence
6. Start Then Connect

Evidence Playground

Places to practice PowerShell for security work and things to keep open while you do: the course evidence, the detection library, the SOC simulator, the playbook suite, an open-source DFIR toolkit, and reference pages.

Operational Reference

Every command, check and trap from PowerShell for Security Analysts in one place, organized by the task in front of you and linked back to the lesson that explains it.

1. Operational Reference

References & Further Reading

The documentation behind PowerShell for Security Analysts, arranged by the question it answers: the language and its editions, logs and evidence, PowerShell's own security, remoting and APIs, tools and review, and incident response practice.

Project

Project: One Compromised Endpoint

The PowerShell for Security Analysts project: a brief for investigating one compromised laptop from its incident package alone, with the course's PowerShell, from first alert to a PowerShell-generated investigation report.

1. The Brief

Course Completion

Course Exam

PowerShell for Security Analysts end-of-course exam: twenty-two tasks, two per teaching module, each answered by writing PowerShell against an evidence set the course did not investigate, and checked on the server.

1. Course Completion. PowerShell for Security Analysts

Generated 2026-10-08 from the published course. The current syllabus is always at ridgelinecyber.com/training/courses/powershell-security-operations/