SEC505: **Securing Windows and PowerShell Automation**

GCWN
Windows Security Administrator
giac.org/gcwn

6 Day Program | 36 CPEs | Laptop Required

You Will Be Able To

- Write PowerShell scripts for security automation of Windows and Active Directory
- Execute PowerShell scripts on remote systems using SSH or TLS
- Install and use OpenSSH for Windows
- Use generative AI services to help write and optimize PowerShell scripts
- Harden PowerShell itself against abuse and enable the logging of PowerShell commands for your SIEM
- Use PowerShell to access the WMI service for remote command execution, searching event logs, reconnaissance, and more
- Block the lateral movement of hackers and ransomware using Windows Firewall, admin credential protections, and more
- Manage Windows Firewall rules with PowerShell
- Prevent exploitation using AppLocker and other Windows OS hardening techniques in a scalable way with PowerShell
- Configure the Just Enough Admin (JEA) feature of PowerShell to create a Windows version of Linux sudo and setuid root
- Configure mitigations against pass-the-hash attacks, Kerberos Golden Tickets, Remote Desktop Protocol (RDP) hijacking, Security Access Token abuse, and other attacks discussed in SEC504 and other SANS hacking courses
- Install and manage a full Windows Public Key Infrastructure (PKI) with PowerShell, such as for smart cards, certificate auto-enrollment, Online Certificate Status Protocol (OCSP), and the detection of spoofed root Certificate Authentications (CAs)
- Harden essential protocols against exploitation, such as SSL/TLS, SSH, RDP, SMB, and PowerShell Remoting

WINDOWS SECURITY AUTOMATION MEANS POWERSHELL

In this course you will learn how to:

- Write PowerShell scripts for Windows and Active Directory security automation
- Use generative Artificial Intelligence (AI) to help write and optimize your scripts
- Run PowerShell scripts on thousands of hosts over the network with SSH or TLS
- Defend against PowerShell malware such as ransomware
- Harden Windows Server and Windows 11 against skilled attackers

You will leave this course ready to start writing your own PowerShell scripts to help secure your Windows environment. It's easy to find Windows security checklists, but how do you automate those changes across thousands of machines? How do you safely run scripts on many remote boxes? In this course you will learn not just Windows and Active Directory security, but how to automate security using PowerShell.

FOR GOV/MIL ATTENDEES, LEARN HOW TO LEVERAGE POWERSHELL AS A FORCE MULTIPLIER FOR WINDOWS SECURITY

PowerShell is popular because it's fun! You will be surprised at how much you can accomplish with PowerShell in a short period of time, especially when using generative AI to help write your scripts.

Learning PowerShell is also useful for another kind of security: job security. Employers are looking for IT people with scripting and automation skills. You don't have to know any PowerShell to attend this course, we will learn PowerShell together during the hands-on labs.

WE WILL WRITE A POWERSHELL RANSOMWARE SCRIPT AND DEFEND AGAINST IT

Unfortunately, PowerShell is being abused by attackers, so in the capstone lab of the course we will write a ransomware script to practice our new coding skills and then use that script to discuss defenses against malware. If you're new to PowerShell, don't worry, there are lots of hints to guide you through this capstone lab, and you will be encouraged to use generative AI to help you write your ransomware script.

ARTIFICIAL INTELLIGENCE HAS REVOLUTIONIZED YOUR PROFESSION

The future of software development is to use generative Artificial Intelligence (AI) to help write, test and optimize source code. Prepare to be amazed at what you can do with AI and PowerShell today! We will use a free AI coding service in a lab to get experience, discuss how to use AI safely, and install Microsoft Visual Studio Code. Visual Studio Code is free and the most popular tool for writing and debugging PowerShell.

The course author, Jason Fossen, is a SANS Institute Fellow and has been writing and teaching for SANS since 1998. Jason Fossen's career focus has always been Microsoft Windows security, especially for high security GOV/MIL on-premises environments. SEC505 has included PowerShell for more than 15 years.

"In SEC505, real-life solutions are offered by someone who understands the roadblocks in the way. This is information I could implement tomorrow and make my network more secure."

— Mary Becken, **Egan Company**

Section Descriptions

SECTION 1: Learn PowerShell Scripting for Security

This section of the course covers what you need to know to get started using PowerShell. You do not need to have any prior scripting or programming experience. We have PowerShell labs throughout the course, so this section is not the only PowerShell material. We start with the essentials, then go more in depth as the course progresses. Do not worry, you will not be left behind, the PowerShell labs walk you through every step. If you already have PowerShell experience, then there will be intermediate topics for you too.

TOPICS: Writing Your Own Scripts, Functions and Modules; Up and Running Quickly with PowerShell; Piping Objects Instead of Text; Flow Control; PowerShell Core; Capturing the Output of Commands; Variables, Arrays and Blocks

SECTION 2: You Don't Know THE POWER!

How can we run PowerShell scripts on thousands of remote systems with just a few lines of code? This section is about remote command execution. We will use built-in PowerShell Remoting, install OpenSSH for Windows, configure the Task Scheduler service to run scripts hands-free, and use Group Policy to run scripts at boot up, log on, log off, and shutdown. To do it more safely, we will see exactly how to configure the Just Enough Admin (JEA) for PowerShell Remoting and how to harden OpenSSH for Windows.

TOPICS: PowerShell Remoting; WSMAN with TLS vs. SSH; Installing and Securing OpenSSH on Windows; PowerShell Just Enough Admin (JEA) sandboxes; Using the Task Scheduler to Run Scripts Hands-Free; Group Policy Management of PowerShell Settings

SECTION 3: WMI and Active Directory Scripting

PowerShell is deeply integrated into the Windows Management Instrumentation (WMI) service. Many PowerShell commands are just wrappers for WMI functions. Hackers love the WMI service too, but for the wrong reasons. The WMI service is enabled by default and accessible over the network. With our PowerShell WMI scripts we can remotely execute commands, reboot machines, forcibly log users off, kill processes, and much more. WMI scripting is a bit difficult, but we'll go through all the strange syntax together. In this section we will also use PowerShell to search, manage, and secure Active Directory. With PowerShell we can, for example, find abandoned user accounts and disable them, enforce proper membership in the Domain Admins group, and reset the passwords of thousands of user accounts. Because malicious insiders can use PowerShell too, we will see how to restrict what users can see or change in Active Directory using PowerShell, and how to enable Active Directory logging.

TOPICS: Why Hackers Love WMI; PowerShell for WMI; WMI Namespaces and Classes; Remote Access to the WMI Service; PowerShell Scripting of Active Directory; Active Directory Permissions and Auditing

SECTION 4: PowerShell DevOps and Artificial Intelligence Generated Code

Generative AI exists today for PowerShell and it is amazing! AI is not just fun, it can cut in half the amount of time it takes to write a long script. You are unlikely to lose your job to AI, but you might lose your job to someone who knows how to use AI better than you. We will discuss AI service providers, how to use AI assistance when writing scripts, and how to use AI safely. You will also install Microsoft Visual Studio Code to handle errors gracefully. PowerShell has great error handling features, but they can be a bit complex. For DevOps hardening of Windows Server, we will also cover the orchestration of multiple scripts to automate tedious tasks. Server hardening includes role management, disabling unnecessary services, setting registry values, locking down privileges, and configuring firewall rules. After configuring Windows Firewall rules with PowerShell, you might find it easier to script these rules than to use the traditional graphical tool for it.

TOPICS: Generative AI for PowerShell; ChatGPT and Microsoft Copilot; Microsoft Visual Studio Code; PowerShell Error Handling; DevOps Orchestration of Multiple Scripts; Server Hardening Automation; Windows Firewall Scripting

SECTION 5: Certificates and Multifactor Authentication

Smart cards and smart tokens, such as YubiKeys, are the gold standard for multifactor authentication (MFA). We also need PKI for TLS encryption and code-signing certificates. We will use PowerShell to install a certificate server that can be used to deploy smart cards, smart USB tokens, and TLS certificates. Smart cards and tokens can be used for PowerShell Remoting, signing PowerShell scripts, Remote Desktop Protocol (RDP) logons, User Account Control (UAC), ASP.NET web application logons, and more. We will use PowerShell to request new certificates, hash files, encrypt files, harden SSL/TLS, and manage a computer's list of trusted root Certification Authorities.

TOPICS: Installing PKI with PowerShell; Certificate Authentication and TLS Encryption for PowerShell Remoting; How to Deploy Smart Cards/Tokens; TPM Virtual Smart Cards; Hashing; File Encryption; Hardening PKI

SECTION 6: PowerShell Ransomware and Security

In a capstone lab, you will write a PowerShell ransomware script and unleash it inside your training virtual machines. The purpose of this ethical hacking is to practice your new PowerShell coding skills, practice using AI help, and to discuss defenses against this kind of PowerShell abuse. How can we secure PowerShell itself? There is no magic patch to make PowerShell "secure", but there are many defensive techniques to reduce future compromises, limit the harm we suffer after a compromise, and to gain visibility into PowerShell malicious activity for the sake of forensics, incident response, and threat hunting.

TOPICS: PowerShell Ransomware; Anti-Exploitation Defenses for PowerShell; PowerShell Logging for Visibility and Detection; Final Capstone Lab for the Course

Who Should Attend

- Anyone who wants to learn PowerShell automation
- Windows administrators
- Anyone implementing the CIS Critical Security Controls
- Anyone implementing the MITRE ATT&CK mitigations
- Blue Team defenders who were terrified by SEC504 and other hacking courses at SANS

"This class provided real-world examples and sample scripts to make a Windows-centric environment fundamentally more secure."

— Nick Boardman, **HRSD**

GCWN
Windows Security Administrator
giac.org/gcwn

GIAC Certified Windows Security Administrator

The GIAC Certified Windows System Administrator (GCWN) certification validates a practitioner's ability to secure Microsoft Windows clients and servers. GCWN certification holders have the knowledge and skills needed to configure and manage the security of Microsoft operating systems and services, including Active Directory, PKI, Group Policy, PowerShell, OpenSSH, WMI and WSMAN for hardening Windows against malware and advanced persistent adversaries.

- Defensible networking
- Endpoint protection
- Operating system and application hardening
- PKI management
- Restricting administrative compromise
- Securing PowerShell