SEC660

GXPN Certification

Exploit Researcher and Advanced Penetration Tester

www.giac.org/gxpn

Advanced Penetration Testing, Exploit Writing, and Ethical Hacking

Six-Day Program
46 CPEs
Laptop Required
This course has evening Bootcamp Sessions

Who Should Attend

- Network and systems penetration testers
- Incident handlers
- Application developers
- IDS engineers

You Will Be Able To

- Perform fuzz testing to enhance your company's SDL process
- Exploit network devices and assess network application protocols
- Escape from restricted environments on Linux and Windows
- Test cryptographic implementations
- Model the techniques used by attackers to perform 0-day vulnerability discovery and exploit development
- Develop more accurate quantitative and qualitative risk assessments through validation
- Demonstrate the needs and effects of leveraging modern exploit mitigation controls
- Reverse-engineer vulnerable code to write custom exploits

> "As a security engineer, SEC660 is very relevant to my job when I am testing internal products."
>
> -Tom Nicholls, Thales

> "The quality of the labs and coursework in SEC660 showcases the value SANS training has over other providers. It was an excellent, challenging, and rewarding course."
>
> -Michael Robinson, UPS

This course is designed as a logical progression point for those who have completed **SEC560: Network Penetration Testing and Ethical Hacking**, or for those with existing penetration testing experience. Students with the prerequisite knowledge to take this course will walk through dozens of real-world attacks used by the most seasoned penetration testers. **The methodology of a given attack is discussed, followed by exercises in a real-world lab environment to solidify advanced concepts and allow for the immediate application of techniques in the workplace.** Each day includes a two-hour evening bootcamp to allow for additional mastery of the techniques discussed and even more hands-on exercises. A sample of topics covered includes weaponizing Python for penetration testers, attacks against network access control (NAC) and VLAN manipulation, network device exploitation, breaking out of Linux and Windows restricted environments, IPv6, Linux privilege escalation and exploit-writing, testing cryptographic implementations, fuzzing, defeating modern OS controls such as ASLR and DEP, return-oriented programming (ROP), Windows exploit-writing, and much more!

Attackers are becoming more clever and their attacks more complex. In order to keep up with the latest attack methods, you need a strong desire to learn, the support of others, and the opportunity to practice and build experience. **SEC660 provides attendees with in-depth knowledge of the most prominent and powerful attack vectors and an environment to perform these attacks in numerous hands-on scenarios.** This course goes far beyond simple scanning for low-hanging fruit, and shows penetration testers how to model the abilities of an advanced attacker to find significant flaws in a target environment and demonstrate the business risk associated with these flaws.

SEC660 starts off by introducing the advanced penetration concept, and provides an overview to help prepare students for what lies ahead. The focus of day one is on network attacks, an area often left untouched by testers. Topics include accessing, manipulating, and exploiting the network. Attacks are performed against NAC, VLANs, OSPF, 802.1X, CDP, IPv6, VOIP, SSL, ARP, SNMP, and others. Day two starts off with a technical module on performing penetration testing against various cryptographic implementations. The rest of the day is spent on network booting attacks, escaping Linux restricted environments such as chroot, and escaping Windows restricted desktop environments. Day three jumps into an introduction of Python for penetration testing, Scapy for packet crafting, product security testing, network and application fuzzing, and code coverage techniques. Days four and five are spent exploiting programs on the Linux and Windows operating systems. You will learn to identify privileged programs, redirect the execution of code, reverse-engineer programs to locate vulnerable code, obtain code execution for administrative shell access, and defeat modern operating system controls such as ASLR, canaries, and DEP using ROP and other techniques. Local and remote exploits, as well as client-side exploitation techniques, are covered. **The final course day is dedicated to numerous penetration testing challenges requiring you to solve complex problems and capture flags.**

SANS
www.sans.org/SEC660

SANS Technology Institute
www.sans.edu

sapere aude
www.sans.org/cyber-guardian

BUNDLE ONDEMAND
WITH THIS COURSE
www.sans.org/ondemand

Course Day Descriptions

660.1 HANDS ON: Network Attacks for Penetration Testers

Day one serves as an advanced network attack module, building on knowledge gained from SEC560. The focus will be on obtaining access to the network; manipulating the network to gain an attack position for eavesdropping and attacks, and for exploiting network devices; leveraging weaknesses in network infrastructure; and taking advantage of client frailty.

Topics: Bypassing Network Admission Control; Impersonating Devices with Admission Control Policy Exceptions; Exploiting EAP-MD5 Authentication; Custom Network Protocol Manipulation with Ettercap and Custom Filters; Multiple Techniques for Gaining Man-in-the-Middle Network Access; Exploiting OSPF Authentication to Inject Malicious Routing Updates; Using Evilgrade to Attack Software Updates; Overcoming SSL Transport Encryption Security with Sslstrip; Remote Cisco Router Configuration File Retrieval; IPv6 for Penetration Testers

660.2 HANDS ON: Crypto, Network Booting Attacks, and Escaping Restricted Environments

Day two starts by taking a tactical look at techniques penetration testers can use to investigate and exploit common cryptography mistakes. We finish the module with lab exercises that allow you to practice your new-found crypto attack skill set against reproduced real-world application vulnerabilities.

Topics: Pen Testing Cryptographic Implementations; Exploiting CBC Bit Flipping Vulnerabilities; Exploiting Hash Length Extension Vulnerabilities; Delivering Malicious Operating Systems to Devices Using Network Booting and PXE; PowerShell Essentials; Enterprise PowerShell; Post-Exploitation with PowerShell and Metasploit; Escaping Software Restrictions; Two-hour Evening Capture-the-Flag Exercise Using PXE, Network Attacks, and Local Privilege Escalation

660.3 HANDS ON: Python, Scapy, and Fuzzing

Day three starts with a focus on how to leverage Python as a penetration tester. It is designed to help people unfamiliar with Python start modifying scripts to add their own functionality while helping seasoned Python scripters improve their skills. Once we leverage the Python skills in creative lab exercises, we move on to leveraging Scapy for custom network targeting and protocol manipulation. Using Scapy, we examine techniques for transmitting and receiving network traffic beyond what canned tools can accomplish, including IPv6.

Topics: Becoming Familiar with Python Types; Leveraging Python Modules for Real-World Pen Tester Tasks; Manipulating Stateful Protocols with Scapy; Using Scapy to Create a Custom Wireless Data Leakage Tool; Product Security Testing; Using Taof for Quick Protocol Mutation Fuzzing; Optimizing Your Fuzzing Time with Smart Target Selection; Automating Target Monitoring While Fuzzing with Sulley; Leveraging Microsoft Word Macros for Fuzzing .docx files; Block-Based Code Coverage Techniques Using Paimei

SANS

SEC660 Training Formats

(subject to change)

Live Training
www.sans.org/security-training/by-location/all

Summit Events
www.sans.org/summit

Mentor Training
www.sans.org/mentor

Private Training
www.sans.org/onsite

vLive
www.sans.org/vlive

Simulcast
www.sans.org/simulcast

OnDemand
www.sans.org/ondemand

SelfStudy
www.sans.org/selfstudy

660.4 HANDS ON: Exploiting Linux for Penetration Testers

Day four begins by walking through memory from an exploitation perspective as well as introducing x86 assembler and linking and loading. Processor registers are directly manipulated by testers and must be intimately understood. Disassembly is a critical piece of testing and will be used throughout the remainder of the course. We will take a look at the Linux OS from an exploitation perspective and discuss the topic of privilege escalation.

Topics: Stack and Dynamic Memory Management and Allocation on the Linux OS; Disassembling a Binary and Analyzing x86 Assembly Code; Performing Symbol Resolution on the Linux OS; Identifying Vulnerable Programs; Code Execution Redirection and Memory Leaks; Return-Oriented Programming (ROP); Identifying and Analyzing Stack-Based Overflows on the Linux OS; Performing Return-to-libc (ret2libc) Attacks on the Stack; Defeating Stack Protection on the Linux OS; Defeating ASLR on the Linux OS

660.5 HANDS ON: Exploiting Windows for Penetration Testers

On day five we start with covering the OS security features (ALSR, DEP, etc.) added to the Windows OS over the years, as well as Windows-specific constructs, such as the process environment block (PEB), structured exception handling (SEH), thread information block (TIB), and the Windows API. Differences between Linux and Windows will be covered. These topics are critical in assessing Windows-based applications. We then focus on stack-based attacks against programs running on the Windows OS.

Topics: The State of Windows OS Protections on Windows 7, 8, 10, Server 2008 and 2012; Understanding Common Windows Constructs; Stack Exploitation on Windows; Defeating OS Protections Added to Windows; Creating a Metasploit Module; Advanced Stack-Smashing on Windows; Using ROP; Building ROP Chains to Defeat DEP and Bypass ASLR; Windows 7 and 8; Porting Metasploit Modules; Client-side Exploitation; Windows Shellcode

660.6 HANDS ON: Capture the Flag Challenge

This day will serve as a real-world challenge for students by requiring them to utilize skills they have learned throughout the course, think outside the box, and solve a range of problems from simple to complex. A web server scoring system and Capture-the-Flag engine will be provided to score students as they capture flags. More difficult challenges will be worth more points. In this offensive exercise, challenges range from local privilege escalation to remote exploitation on both Linux and Windows systems, as well as networking attacks and other challenges related to the course material.