ARISHTI

CONSOLIDATED

Training program:

Making Exploit Development Fun

Unlock the Mysteries of Cybersecurity: Master Exploit Development

Contents

Course Overview 3
Course Structure: 4
Module 1: Overview and Background Knowledge 4
Module 2: Stack-based Buffer Overflow 4
Module 3: Shellcode Development 4
Module 4: Format String Vulnerabilities 4
Module 5: Heap-based Buffer Overflow 4
Module 6: Integer Overflow 4
Module 7: Return-Oriented Programming (ROP) 4
Module 8: Advanced Vulnerability Analysis Techniques 4
Module 9: Hands-on: Exploit Development for Open-Source Software 5
Teaching Methodology: 5
Materials Needed: 5
Thank You. 6

Course Overview

This training program covers everything you need to know about developing exploits, starting from the basics and moving to advanced methods. You will learn about various types of vulnerabilities, especially focusing on memory corruption issues, how to create shellcode (the small piece of code that is used as the payload for exploiting a vulnerability), and how to analyze vulnerabilities in real-world software applications. The course is taught by an experienced researcher from Palo Alto Networks, who has a strong background in securing Internet of Things (IoT) devices and control systems used in industries (SCADA systems). This is a great opportunity for anyone looking to expand their knowledge in cybersecurity and exploitation techniques, whether you're starting out or looking to enhance your skills further.

Course Description: Unlock the secrets of exploit development, one of cybersecurity's most intriguing disciplines. This course bridges the gap between software proliferation and the skills needed to identify and exploit vulnerabilities. From foundational concepts to advanced techniques, participants will explore the essentials of exploit development, focusing on SCADA and IoT environments. Hands-on practice in real-world scenarios ensures a comprehensive learning experience.

Course Objectives:

- Grasp the foundational concepts of exploit development.
- Identify and exploit common vulnerabilities such as buffer overflows and format string issues.
- Develop practical skills in shellcode writing and advanced exploitation techniques.
- Gain insights into vulnerability research, particularly in critical infrastructure settings.

Course Structure:

Module 1: Overview and Background Knowledge

- Understanding Compilers, Linkers, and Loaders
- Deep Dive into x86 and x86-64 Architectures
- Linux File Permissions and Set-UID Programs
- Memory Mapping in Linux Processes
- System Calls and Shell Variables
- Introduction to Basic Reverse Engineering
- ELF File Format

Module 2: Stack-based Buffer Overflow

- What is a Stack-based Buffer Overflow?
- Practical Exploitation Techniques
- Defenses and Countermeasures

Module 3: Shellcode Development

- Writing Shellcode from Scratch
- Types of Shellcodes: Bind, Reverse, etc.
- Shellcode Evasion Techniques

Module 4: Format String Vulnerabilities

- Understanding Format String Bugs
- Exploiting Format Strings for Memory Leaks and Code Execution

Module 5: Heap-based Buffer Overflow

- Heap Management Basics
- Exploiting Heap Overflows

Module 6: Integer Overflow

- Integer Vulnerabilities Overview
- Practical Exploitation of Integer Overflows

Module 7: Return-Oriented Programming (ROP)

- ROP Basics
- Advanced ROP Chains
- Bypassing DEP and ASLR

Module 8: Advanced Vulnerability Analysis Techniques

- Fuzzing Methodologies
- Introduction to Symbolic Execution
- Tools and Techniques for Vulnerability Discovery

Module 9: Hands-on: Exploit Development for Open-Source Software

- Guided Exploit Development Workshops
- Case Studies on Real Vulnerabilities in Open-Source Projects
- Live Debugging and Exploitation Practice

Teaching Methodology:

- Lectures: Cover theoretical aspects with real-world examples from SCADA and IoT.
- Interactive Sessions: Q&A sessions after each module.
- Workshops: Hands-on labs with tools like GDB, IDA, and custom environments.
- Guest Lectures: Insights from industry experts.
- Project: Students work on a mini-project to find and exploit a vulnerability in open-source software under supervision.

Materials Needed:

- Virtual machines pre-configured with necessary tools
- Access to a lab environment (cloud-based or local setup)
- Course slides, practical guides, and a reference list of tools and further readings

ARISHTI
CONSOLIDATED

Thank You.

We at Arishti Consolidated appreciate your participation in our training program.

Regards,
Animesh Roy
CEO, Arishti Consolidated.
https://arishtisecurity.com