The Hacktivists

Hacking | Forensic | Security

SYLLABUS

AAAE

ADVANCED ANDROID APPLICATION EXPLOITATION VERSION 1

The most practical and extensive training course on android application pentesting

IT Security Professionals have chosen "THE HACKTIVISTS" as their best cybersecurity training provider.
We have trained professionals who are working in Fortune 500 companies and Best organization across 100+ countries around the globe.

wipro hp accenture Infosys paytm verizon EY Cognizant

HCL amazon Honeywell PhonePe Mindtree Mphasis Deloitte. pwc

INTRODUCTION

2020

COURSE DESCRIPTION

The Advanced Android Application Exploitation (AAAE) is an online training program that provides all the high-level skills required for professional penetration test against modern android applications. AAAE is a unique training which covers security and exploitation of android applications, and it is different from traditional androidapplication penetration testing approach.

This training includes the most advanced android application attacks, exploitation and pentesting techniques. This training, although based on the offensive approach, provides the most excellent exercises to solve modern android application security issues discovered during bug bounty hunting and penetration testing.

PRE-REQUISITES

AAAE is advanced training that requires the following pre-requisites:

- Basic knowledge of programming fundamentals.
- Basic knowledge of programming such as Java and Objective-C
- One year in an information security role or equivalent experience is recommended.
- Ability to read and understand android application code will help, although it is not mandatory.

The Hacktivists AAAE training provides most of the above pre requisites.

WHO SHOULD TAKE THIS COURSE?

Advanced Android Application Exploitation (AAAE) training is beneficial for:

- Bug Bounty Hunters
- Penetration Testers
- Application Developers
- Mobile Security Enthusiasts
- IT Security professionals with a technical background

INTRODUCTION

2020

WILL I GET A CERTIFICATE ?

Once you satisfy the requirements of the final practical certification test, you will be awarded an "Advanced Android Application Exploitation" certificate and will hold the AAAE certification.

DETAILED COURSE CONTENT

AAAE training based on a live android application testing where we cover all the practical skills essential to understanding the technical threats and attack vectors targeting android applications.

AAAE training covers how to code Android applications to build real-world POCs and exploits. These skills will be required to understand android application security completely.

Module 1 : Android Bug Bounty Hunting Approach - Introduction

Module 2 : Android Application Security Testing Lab Environment

Module 3 : Primary Stage to Security Analysis of AndroidApp

Module 4 : Code Quality and Build Settings of Android Application

Module 5 : Tampering Android Application and Security Analysis

Module 6 : Security Analysis of Android Source Code

Module 7 : Insufficient Transport Layer Protection

INTRODUCTION

2020

Module 8 : Insecure Connection and Untrusted Connections

Module 9 : Insecure Logging Security Issues

Module 10 : Insecure Sensitive Hardcoding Issues

Module 11 : Confidential Information Exposure By Design (Side Channel Data Leakage)

Module 12 : Security Issues in OAuth Implementations

Module 13 : Insecure Cryptographic Storage

Module 14 : Unprotected Application Components

Module 15 : Private File Access Security Issues

Module 16 : Testing Code Quality and Injection Flaws

Module 17 : Security Analysis of API Endpoints with Telerik Fiddler

Module 18 : Insufficient Anti Automation

Module 19 : Insecure Authentication and Authorization

Module 20 : Improper Access Control

Module 21 : Server Side Vulnerabilities

Module 22 : Android Application Older Vulnerabilities

MODULES

2020

MODULE 1 Android Bug Bounty Hunting Approach - Introduction

- Android Penetration Testing Methodologies - Detailed Explanation
- Android Bug Bounty Methodology according to Bug Hunting Platforms
- Differences between Android Pentesting & Bug Bounty Approach
- Traditional Android Penetration Testing Report - Test Cases
- Traditional Android Penetration Testing Approach and Guidelines
- Android Application Attack Surface - Client Side Vulnerabilities
- Android Application Attack Surface - Server Side vulnerabilities
- Android Application Attack Surface - Logical Security Threats

MODULE 2 Android Application Security Testing Lab Environment

- Install Android Pentest Operating System
- Genymotion Android Emulator Installation
- Installing Android App components (GSuite)
- Installing Android App components ARM Translator
- An Overview of the Android Architecture
- An Overview of the Application Framework
- An Overview of the Android Permissions Model

MODULE 3 Primary Stage to Security Analysis of AndroidApp

- Setup Android Debug Bridge Utility (adb)
- Android Debug Bridge (adb) Pentester Utilities
- Vulnerable Android Application Source Code Analysis
- Understanding Source Code Compilation Process
- Structure of an Android Application Package (APK)
- Unzipping and Unpacking Android Applications
- Reversing an Android application using dex2jar
- Reversing an Android application using apktools

MODULES

2020

MODULE 4 Code Quality and Build Settings of AndroidApp

- Android Application Manifest Overview
- Security Review of Manifest Elements
- Security Analysis of Manifest Elements

MODULE 5 Tampering Android Application and Security Analysis

- Signing an Android Applications Manually
- Code Obfuscation and Code Protection
- Adding Malicious Code to Android Apps
- Debugging Detection
- Root Detection
- VM Detection

MODULE 6 Security Analysis of Android Source Code

- Steps for Static Source Code Analysis
- Searching Vulnerable Functions in Source Code
- Steps for Dynamic Security Analysis of Application
- Dynamic Security Analysis using Drozer Security Testing Framework

MODULE 7 Insufficient Transport Layer Protection

- Dynamic Security Analysis using BurpSuite
- An Introduction and Installation of Xposed Framework
- Android SSL Verification and Certificate Pinning
- Bypass SSL Pinning to Perform Active Man-in-the-Middle

MODULE 8 Insecure Connection and Untrusted Connection

- Use of Insecure Network Protocols
- Authentication over Insecure Protocols
- Data Transmission over Insecure Protocols

MODULES

2020

MODULE 9 Insecure Logging Security Issues

- Insecure Logging - Verbose Error Logging
- Insecure Logging - Authentication Token Leakage
- Insecure Logging - Sensitive Information Disclosure
- Insecure Logging - Personally identifiable information (PII)

MODULE 10 Insecure Sensitive Hardcoding Issues

- Insecure Hardcoding - API Keys Leakage
- Insecure Hardcoding - Authentication Token
- Insecure Hardcoding - Internal IP Disclosure
- Insecure Hardcoding - Git Repository Disclosure
- Insecure Hardcoding - Embedded Third-Party Secrets
- Insecure Hardcoding - Sensitive Information Disclosure

MODULE 11 Confidential Information Exposure By Design (Side Channel Data Leakage)

- Confidential Information Leakage - Insecure Backup Storage
- Confidential Info Leakage - Screen Capture on Personal Data
- Confidential Info Leakage - Application Level Denial-of Service
- Confidential Info Leak - Personal Data using Virtual Keyboard
- Confidential Info Leak - Sensitive Data Copied to Clipboard
- Confidential Info Leak - Data Disclosure Through UserInterface
- Confidential Info Leak - Sensitive data Cleartext Storage in Memory

MODULE 12 Security Issues in OAuth Implementations

- Leaking OAuth Tokens - Android logcat
- Leaking OAuth Tokens - Shared Preferences
- Leaking OAuth Tokens - OAuth HardCoded Secret Tokens
- Leaking OAuth Tokens - Inadequate transmission protection

MODULES

2020

MODULE 13 Insecure Cryptographic Storage

- Insecure Cryptographic Storage - SQLite Databases
- Insecure Cryptographic Storage - Internal Storage
- Insecure Cryptographic Storage - External Storage
- Insecure Cryptographic Storage - Shared Preferences

MODULE 14 Unprotected Application Components

- Unprotected Application Components - Unprotected Services
- Unprotected Application Components - Unprotected Activities
- Unprotected App Components - Leaking Content Providers
- Unprotected App Components - Typos in Custom Permissions
- Unprotected App Components - Implicit Broadcasts (Sending)
- Unprotected App Components - Implicit Broadcasts (Receiving)
- Unprotected App Components - Android Fragment Injection
- Unprotected App Components - Allowing Manipulation Unprotected Activities

MODULE 15 Private File Access Security Issues

- Private File Access - Local File Inclusion
- Private File Access - Remote Command Execution Vulnerability
- Private File Access - Private Data Overwrite due to Path Traversal
- Private File Access - Private Data Overwrite due to ZIP FileTraversal

MODULE 16 Testing Code Quality and Injection Flaws

- Injection Flaws - SQL Injection
- Injection Flaws - HTML Injection
- Injection Flaws - Cross Site Scripting
- Injection Flaws - Improper Markup Sanitisation
- Injection Flaws - Crash App & DoS using other app
- Injection Flaws - Insecure Deeplink leads to Sensitive Information Disclosure

MODULES

2020

MODULE 17 Security Analysis of API Endpoints with Telerik Fiddler

- Composing Application API Calls - Functional API
- Capturing Application API Calls - Functional API
- Filtering Application Request Traffic
- Analyzing the Authentication Endpoints
- Analyzing an Additional API Call
- Analyzing Sensitive Data Disclosure in API Endpoint

MODULE 18 Insufficient Anti Automation

- Insufficient Anti Automation - Registration
- Insufficient Anti Automation - Login (static)
- Insufficient Anti Automation - Password Reset Function

MODULE 19 Insecure Authentication and Authorization

- Bypass One Time Verification Codes
- OTP SMS or Voice Code Leaked in Response
- Bypass Second Factor Authentication (2FA)

MODULE 20 Improper Access Control

- Improper Access Control
- Insufficient Entropy For Random Values
- Personally Identifiable Information (PII) Disclosure

MODULE 21 Server Side Vulnerabilities

- Improper Session Handling
- Leakage of API Auth Tokens
- Improper Restriction of Misconfigured API
- Improper Restriction of Unprotected APIs Endpoint
- Transporting API Auth tokens as Cleartext Allowed

MODULES

2020

MODULE 22 Beware of Recent Android Vulnerabilities

- Tapjacking Vulnerability
- Remote Wipe Vulnerability
- AAPT Time Zone Disclosure Bug
- Android Master Key vulnerability
- Address Bar Spoofing Vulnerability

ABOUT US

Our Company The Hacktivists™ (Leading IT Security Services & Training Providing Company) offers a wide range of courses & training in Information Cyber Security. We are a Fast-growing online information security training company based out of India.

We have a large number of professional instructors who are specialized and experienced in various Information/Cyber Security domains. Our Instructors holds a wide range of accreditation like OSCP, OSEE, OSCE, eCXD, eMAPT, eWPTX, eWDP, CEH, CHFI, CISSP, CISM, CISA.

The Hacktivists™ is one of the most trusted and reliable training providers in information/cybersecurity, providing exceptional unmatched Hands-on practical training to individuals and corporates worldwide. Our goal is to train, mentor, and support your career in cybersecurity.

We emphasize more on hands-on practical training which gives our clients and candidate an edge to grow and advance professionally in their respective career(s).

Contact details:
www.thehacktivists.in
info@thehacktivists.in