Advanced Programming & Cyber Security Syllabus

October 2019 Cohort

Summary

**Written and approved by ITC Cyber Security Tech Lead and Fellows Program Director

1. Programming and Fundamental Skills

The gist: programming skills as well as fundamental skills relevant to development in general and in cyber security in particular.

Syllabus

Tools and Frameworks

This section of the course includes core components that are essential to developing in the industry, from **good coding practices** as expected in the industry, to working in **version control** in a **Linux** environment.

In the **docker** and **virtual environments** modules, students learn how to create reproducible development environments via virtual environments, and how to create and edit docker environments for an even more robust managing of their packages and other software.

Python

Python is very commonly used both in Cyber research and development. Following this subject, the students should be able to use Python for quickly implementing various solution, and in order to research Cyber-related topics.

Emphasis will be placed on high quality coding, testing, acquaintance with useful modules, and Pythonic writing.

C

C is a powerful language, widely used in the industry and crucial for understanding underlying mechanisms of software. Students will be proficient in writing high quality C code, as well as understand the small details about the operation of the stack, compiling and linking.

Knowing C is crucial for understanding Operating Systems, as well as Reverse Engineering - in later modules.

2. Research Skills

The gist: getting a variety of Reverse Engineering skills

Syllabus

Assembly

This chapter provides the students with an overview of Assembly language on the Intel x64 architecture, as a preparation for more advanced training in Reverse Engineering.

The students will learn the structure of the **CPU**, common assembly **instructions**, control flow, the stack, function calls, and system calls.

Reverse Engineering

Reverse engineering is a crucial research tool used to investigate existing code. Specifically, reverse engineering can be used to understand complicated algorithms and functioning of Malware.

In this module the students will be acquainted with the main tools used in the reverse engineering process, using both **static analysis** and **dynamic analysis** methods.

3. Core Subjects

The gist: Computer Networks and Operating Systems - from development as well as research point of view

Computer Networks

In this module we cover hands on tools such as **Wireshark** as well as Python libraries like **scapy** or **sockets**, to gain a deep understanding of how networks operate.

Students will layer about the **five layers model**, get to know **network devices** and various **protocols** such as Ethernet, ARP, IP, UDP, TCP, DNS, HTTP and others.

This subject will provide the students with skills to research specific protocols, as well as a whole network, in addition to development skills that will allow them to implement network related applications.

Windows Internals

During this subject, the students will gain a comprehensive understanding of Windows operating system in particular, and knowledge of operating system concepts in general.

We will cover everything from the Windows Registry, through Win32API, Objects, Memory Management, Processes & Threads, Synchronization & IPC, to Hooking and Injection.

Hands on practice will include implementing Windows applications in C, and researching the underlying mechanisms of the operating system using WinDBG.

Linux Internals

We will start from simple bash commands, through process management, practice forks and pipes, and go all the way to implement basic device drivers.

Via hands on practice, we will understand the entire process that happens under the hood when performing various operations.

4. Endpoint Security

The gist: from finding vulnerabilities, researching malwares and developing mitigations such as anti viruses, this chapter covers various aspects of endpoint security.

Malware Research

We will start with an overview of the malwares world. The students will learn about types of Malwares and get acquainted with familiar **APTs**. They will understand common **Malware operations** (such has persistence and lateral movement) and get to know techniques used by **real malwares**.

We will focus on actual **analysis** of real malwares. Students will use both static and dynamic analysis methods learnt in Reverse Engineering, as well as knowledge from Computer Networks, Windows Internals and Linux Internals modules in order to investigate real malwares operation.

The students will get to know **sandboxes**, learn about **packers** and practice unpacking, try **signing** malware and deal with **obfuscation**.

Vulnerabilities, Exploits and Exploit Kits

In this module, the students will get to know vulnerabilities such as **buffer overflows**, as well as known exploit kits such as **Metasploit**. We will also learn about **mitigation techniques** (ASLR etc).

5. Network Security

The gist: security issues concerning the enterprise network and the web

Practical Cryptography

This short module provides an introduction to the cryptography world, teaching basic terms and implementing algorithms. The students will learn concepts such as Symmetric and Asymmetric encryption, CA and PKI.

Enterprise Security Architecture

The students will learn about the structure of modern enterprise networks, as well as security devices such as VPNs, Firewalls and Intrusion Detection Systems.

Web Application Security

This subject familiarizes the students with different types of web attacks, from info gathering and authentication vulnerabilities, through input validation, SQL injection, XSS and more.

6. Projects hosted by companies

The 5-weeks projects hosted by the companies usually focus on a proof of concept for an idea that the company wanted to check, or a small tool the company wants to develop. The projects are diverse and rely on different skills the Fellows acquire during the program. They serve as the biggest “hands-on” experience in the course and as an opportunity for the students to demonstrate what they have learned so far and gain additional knowledge and skills. Here is an example for a project hosted by PayPal from previous cohorts.