A POLYTECHNIC INSTITUTION

Course Outline

School of Computing and Academic Studies
Program: Computer Systems Technology
Option: **Bachelor of Technology, Computer Systems**

Course Number: **COMP 8505**
Course Name: **Selected Topics in Network Security Development**

Start Date:	September 3, 2018	**End Date:** December 13, 2018	
Total Hours:	**Total Weeks:**	**Term/Level: 1**	**Course Credits: 3**
Hours/Week: 3	**Lecture: 1** **Lab: 2**		

Prerequisites: Dip. Of Tech in Computer Systems (or equivalent) or Permission of instructor and Program Head

Course No.	**Course Name**	**Course No.**	**Course Name**
COMP 8005	Network and Security Applications Development		

- **Course Description:**

This course is a study of topics of particular interest to advanced B. Tech students in the area of Network software development and Network security. The focus will be on the research and application of advanced implementation of security tools such as Covert Channels, Steganography, Stealth Backdoors, Trojans, Packet crafting and sniffing, and security tools proof-of-concept and prototyping. Also covered will be Malware analysis and Reverse Engineering techniques.

- **Course Goals**

- The primary goal of this course is to get students to apply all the knowledge and skills acquired in Comp 7005 and Comp 8005 in a practical, hands-on manner.
- Conduct an in-depth study of specific and highly specialized areas in Network Programming and secure coding practices.
- Understand the role of security applications, both as exploit and penetration tools in networks.
- Design and implement a variety of security tools using languages such as C, C++, Java, Ruby, and Python.
- Use advanced raw socket and kernel-level programming techniques to implement security applications.
- Implement a substantial project in a selected area of security, and produce a complete application upon completion.

- **Evaluation**

Final Exam:	30%
Assignments:	45%
Project:	25%
TOTAL	100%

Comments: The breakdown of evaluation components will be discussed in class.

- **Course Learning Outcomes/Competencies**

Upon successful completion, the student will be able to:

1. Design and implement a variety of covert channels and network surveillance software.
2. Conduct advanced research into specific and specialized areas in Data Communications such as network security and securing applications.
3. Develop strategies for imaginative and practical solutions for software development problems using custom designed code.
4. Apply and understand the usefulness (i.e., non-malicious use) of applications such as sniffers, backdoors, port knocking and stealth communications.
5. Understand and implement rootkits in the Linux environment.

- **Verification**

I verify that the content of this course outline is current.

Aman Abdulla	August 31, 2018
Authoring Instructor	Date

I verify that this course outline has been reviewed.

Program Head/Chief Instructor	Date

I verify that this course outline complies with BCIT policy.

Dean/Associate Dean	Date

Note: Should changes be required to the content of this course outline, students will be given reasonable notice.

- **Instructor(s)**

Office Location:	SW2-323	Office Phone:	604-432-8837
Office Hrs.:		E-mail Address:	aabdulla@milliways.bcit.ca

- **Learning Resources**

***Required*:**

Students will be required to conduct extensive research on the topics introduced in class using the Internet and other reference material.

Recommended:

The Shellcoders Handbook – Koziol et al
Wiley Publishers

The Rootkit Arsenal – Blunden
Wordware Publishing

Sockets, Shellcode, Porting & Coding – Foster & Price
Syngress

- **Information for Students**

By attending this course and receiving this course outline, you have been made aware of the following policies. Please follow the links provided as each student is responsible for reading and complying with these policies.

The following statements are in accordance with the *BCIT Student Regulations Policy 5002*. To review the full policy, please refer to http://www.bcit.ca/files/pdf/policies/5002.pdf.

Attendance/Illness:
In case of illness or other unavoidable cause of absence, the student must communicate as soon as possible with his/her instructor or Program Head or Chief Instructor, indicating the reason for the absence. Prolonged illness of three or more consecutive days must have a BCIT medical certificate sent to the department. Excessive absence may result in failure or immediate withdrawal from the course or program.

Academic Misconduct:
Violations of academic integrity, including dishonesty in assignments, examinations, or other academic performances are prohibited and will be handled in accordance with the *Violations of Standards of Conduct* section of Policy 5002.

The School of Computing and Academic Studies expects the highest level of professional conduct and ethical behaviour from all students enrolled in part time studies courses and programs. All students are reminded of the BCIT policy related to the *Responsible Use of Information Technology*. Read the full policy here: http://www.bcit.ca/files/pdf/policies/3501.pdf.

The Computing and IT knowledge and skills acquired by students in the course of their studies confers upon them, as with all professionals, a special responsibility to use their knowledge in a responsible, professional and ethical manner. Further, given that misuse of computer facilities at BCIT can have significant legal and/or economic impacts, upon evidence of any such misconduct, the School may recommend immediate suspension, even for first offences.

Attempts:
Students must successfully complete a course within a maximum of three attempts at the course. Students with two attempts in a single course will be allowed to repeat the course only upon special written permission from the Associate Dean. Students who have not successfully completed a course within three attempts will not be eligible to graduate from their respective program.

Accommodation:
Any student who may require accommodation from BCIT because of a physical or mental disability should refer to BCIT's Policy on Accommodation for Students with Disabilities (http://www.bcit.ca/files/pdf/policies/4501.pdf), and contact BCIT's Disability Resource Centre (SW1-2300, 604-451-6963, http://www.bcit.ca/drc/) at the earliest possible time. Requests for accommodation must be made to the Disability Resource Centre, and should not be made to a course instructor or Program area.

Any student who needs special assistance in the event of a medical emergency or building evacuation (either because of a disability or for any other reason) should also promptly inform their course instructor(s) and the Disability Resource Centre of their personal circumstances.

Course Notes:

- Notes will be posted on my Web server which you may access using the following URL:

  http://milliways.bcit.ca/c8505/

- **Assignment Details :**

  - There will be several assignments throughout the course and a final project (to be discussed in class).

Schedule

Module Number	Outcome/Material Covered	Approximate Duration
1	**Covert Channels**: • Covert TCP/IP Channels. • Examples of Covert Channels	**(1 Week)**
2	**Stealth Backdoors**: • Linux Backdoors • Packet handling using libpcap	**(1 Week)**
3	**Concealing Services & Backdoors**: • Port knocking • Firewall rule manipulation • Port knocking examples	**(1 Week)**
4	**Process and File Monitoring**: • Using inotify to develop system monitoring tools • Process and file monitoring Code examples	**(1 Week)**
5	**Raw Socket Application Development** • Raw socket programming • Code examples	**(1 Week)**

Module Number	Outcome/Material Covered	Approximate Duration
6	**Shellcode:** • Writing shellcode • Basic Examples • Binding shells to ports	**(1 Week)**
7	**Steganography** • Principles of Steganography • Types of Steganography • Techniques for concealing data • Code examples	**(1 Week)**
8	**Network Tools – Proof-Of-Concept & Prototyping** • Writing a simple packet sniffer in Ruby • Writing an ARP Poisoning Tool in Ruby • Manipulating DNS Traffic • Code examples	**(1 Week)**
9	**Network Tools – Packet Crafting** • Python packet crafting libraries • Packet Crafting & Fuzzing using Scapy	**(1 Week)**
10	**Malware Analysis & Reverse Engineering Techniques** • Analyzing applications with OllyDbg • Reverse engineering with OllyDbg • Malware analysis	**(1 Week)**
11	**Miscellaneous Topics:** • As required during the course of the term.	

***Topics may be omitted, replaced or added at the discretion of the instructor.**