

CS-6035 Intro to Information Security Prerequisites

CS6035 assumes classical computer science (CS) background, preferably from an accredited undergraduate CS program. A wide variety of CS and programming knowledge is required to be successful in project coursework. These are the prerequisites:

1. Computer Organization and Architecture
 - a. Be able to describe what a stack, heap (high-level memory organization not data structure), registers, virtual memory, and processes are.
 - b. Understanding of the stack's role in program execution.
 - c. How memory is allocated both statically and dynamically.
 - d. Be able to follow basic assembly language as it applies to compiled code (C).
2. Programming
 - a. Experience in programming a high-level programming language. CS6035 projects specifically use C, Python 3.x, Java, HTML/JS/PHP, and SQL in various contexts. The ability to pick up programming languages on the fly will be positive to learning outcomes.
 - b. Experience with algorithms, data structures, and basic time complexity
 - c. The majority of advanced debugging will involve the execution analysis of C programs, with GNU DeBugger (GDB) used as a debugging platform. Knowing how to use GDB to dump registers/stack memory, set breakpoints to follow the execution path, and to analyze compiled assembly language will be very helpful.
 - d. The ability to translate pseudocode into applied, functioning code - even in a programming language that you might not be familiar with and actively learning.
3. Mathematics
 - a. Understanding of public key cryptography. basic hashing mechanism, discrete mathematics, number system foundations, and modular math can be very helpful to learning outcomes.
4. Misc Computer Skills
 - a. Working knowledge of Virtualbox VMs and their intermediate management.
 - b. Basic understanding of how to navigate using a Linux shell and executing basic command line tools.
 - c. How to compile C programs, write and compile Java classes, interpret Python code, use built-in browser tools to analyze Web development interactions between integrated technologies (e.g., HTML and JS), and understanding basic SQL queries will be useful.
 - d. Basic understanding of Linux network protocols and command line tools (nslookup, dig, DNS). In addition knowledge of how to use wireshark is helpful.
 - e. Familiarity with the Microsoft Windows registry, its function, how to change, and consequences of such changes.
5. Understanding that gaps in one's overall skill set and predicate knowledge may require a significant amount of time and effort to overcome when considering the amount of time

required to study course specific materials. Due to the scale of the class and program, students will be expected to fill in the gaps in their own knowledge in these areas, instructors will not have the resources in order to provide such remedial assistance. These hardships are not insurmountable, even without a CS background, if sufficient effort is applied toward learning outcomes.

6. Useful resources for skills mentioned in 2.
 - a. gdb-example-ncurses.html (GDB)
 - b. <http://phrack.org/issues/49/14.html> (Buffer Overflow Concepts)
 - c. <https://www.cprogramming.com/gdb.html> (GDB)
 - d. <https://docs.python.org/3/tutorial/>
 - e. <https://www.hacksplaining.com> (For SQL injection / Javascript / XSS/ CSRF Vulnerabilities)