# Modern Security Topics
## Spring 2017

# Instructors

**Name**:  Collin Berman and Cyrus Malekpour, both 4th year CS majors in SEAS. We are also co-presidents of the [Computer and Network Security Club](#)

**Meeting Time:** Thursday 5:00PM - 5:50PM in Thornton Hall D223

**Office Hours**:  TBD

**Email**: [collin@virginia.edu](mailto:collin@virginia.edu) and [cyrus@virginia.edu](mailto:cyrus@virginia.edu)

**Professor of Contact:** If a student has a significant issue with the course, grades, or instructor, contact the course professor:

> **Name:** David Evans

> **Email:** [evans@virginia.edu](mailto:evans@virginia.edu)

> **Office:** Rice Hall, room 507

# Course Details

## *Pre-requisites*

- CS 2150
- CS 4630 (Defense against the Dark Arts) is **not** required, but would be helpful

## *Course Description*

This course will provide an overview of topics related to modern computer and network security. The focus is to understand modern exploitation and defense, including binary vulnerability attacks, web security, cryptography, provable security, and mobile security. Students will be exposed to current security research and news, and also be given hands-on exploits and attacks to implement.

## *Content Covered*

This course will cover the following areas:

- x86 and x64 assembly code
- Binary analysis tools, including static and dynamic analysis
- Binary exploitation, including manual attacks and fuzzing
- DEP/ASLR protections and bypasses
- Modern browser exploitation
- Windows exploitation and reversing
- Web security (SQLi, XSS, etc) and analysis tools
- Cryptography (primitives, popular systems, modern topics)
- Provable security

- ▪ Android and iOS security architecture
- ▪ Kernel exploitation
- ▪ Basic malware analysis

## *Required Readings*

There are no textbooks required for this class. Any papers or other readings will be posted on the course website. Useful links will also be posted on the course website.

## *Attendance Policy*

Attendance is required for this class, and composes a large amount of the course grade. Students are allowed two unexcused absences for the semester, after which you will lose 5% of their attendance grade per absence. Excused absences do not count against any of the above.

## *Grading Criteria*

Grading in this class is a pass/fail based on two criteria. In order to achieve a "pass", you must attend at least 12 lectures (not including excused absences) and complete 8 out of 12 homework assignments. Homework is due at the start of the class after it is assigned, and some assignments have extra credit which counts as an additional homework opportunity.

For each homework, students will be required to turn in their finished result (code, exploit script, test case, etc) and a brief writeup (~1 paragraph) of what they did to solve the problem.

# Course Schedule

Readings are meant to be completed by the class they are listed next to. Assignments are due before the start of the following class.

| Week # | Topic | Readings Due | Assignment |
|--------|-------|--------------|------------|
| 1 | Syllabus, Security Mindset, review of C concepts | | |
| 2 | (Re)introduction to x86 assembly | | Implement a simple assembly program |
| 3 | Debuggers and static analysis | 2150 GDB Tutorial Part I | Binary reversing assignment |
| 4 | Buffer overflows, ret2libc, ROP | Buffer overflow writeup | Implement a ROP attack |
| 5 | DEP and ASLR protections and bypasses | | ROP with DEP and ASLR implemented |
| 6 | Fuzzing & Browser Exploits | | Simple fuzzing with afl |

| | | | |
|---|---|---|---|
| 7 | Cryptographic primitives | Sections 1.1, 1.2 of [HAC](#) | Cryptopals |
| 8 | RSA, AES, ECDSA | [AES: The Making of a New Encryption Standard](#) | Forge RSA signatures |
| 9 | Modern Topics in Cryptography | Sections 1, 3 of [djb's intro to PQcrypto](#) | Research a topic |
| 10 | Provable Security | Sections 1, 3 of [arXiv:1610.08279](#) | Secure insecure software |
| 11 | Malware Analysis and Reversing | | Reverse a piece of ransomware |
| 12 | Exploitation and analysis on Windows | | Simple Windows binary exploit |
| 13 | Web security and tools | | A SQLi challenge and an XSS challenge |
| 14 | Android and iOS security architecture | Sections from [Android](#) and [iOS](#) security docs | |

## *Assignment Details (by week)*

1. (No homework)
2. Implement a program in x86 assembly for Linux that reads a file and prints it out backwards
3. Reverse a crackme binary to recover a password
4. Use a ROP attack to exploit a simple 32-bit Linux binary
5. Use a ROP attack to exploit a 32-bit Linux binary that is protected by DEP and ASLR
6. Use AFL to find a crash in a program we give you
7. Complete Cryptopals set 1, challenges 1–4. For extra credit, also complete challenge 5 and 6.
8. Carry out an existential forgery against a network service implementing a signing oracle using unpadded RSA. For extra credit, carry out a selective forgery.
9. Find a cool topic in cryptography from the past few years and write a short summary.
10. Given a piece of software along with an incorrect proof of security, identify sections of code that prevent the proof from going through and correct these sections to prove the system secure.
11. You're given a piece of ransomware! Figure out what it does and decrypt the file we give you
12. Exploit a simple vulnerable binary on Windows
13. Complete challenges from https://github.com/cnsuva/web-challenges-server
14. (No homework)