Web Application Programming and Hacking

(EECE8095-001 & EECE4005-01)

Course Syllabus and Schedule, Summer 2024

Latest update: May 3, 2024. This document is live on: https://bit.ly/waph-sm24

General Information

This class is asynchronous; lectures will be recorded and released around 5:00 PM or earlier on Tuesdays and Thursdays. Detailed class schedule here->.
Lecture slides and video recordings for each lecture will be posted twice a week on Canvas. Students are required to watch video recordings to engage with class activities.
All course materials, such as the syllabus, major assignments, and handouts, can be found on the course page located on Canvas at https://uc.instructure.com/courses/1692942
(Canvas -> Courses -> (24US-Full) SPECIAL STUDIES (001); EECE4005-001, cross-listed with EECE8095-001).

Instructor: Dr. Phu Phung, Visiting Scholar, Department of Electrical and Computer Engineering, homepage: https://researchdirectory.uc.edu/p/1697637

Email: phungph@ucmail.uc.edu

Office Hours: 2:00-4:00 pm Tuesdays and Thursdays on Teams. You must book a 10-minute slot in advance (link->). Additional slots might be available upon request.

Teaching Assistant: Sohan Chidvilas Bodapati (bodapass@mail.uc.edu)

Office Hours: 11:30 am-12:30 pm and 5:30-7:30 pm MWF on Teams. Additional slots might be available upon request.

Course Description

In this course, students will study basic web application development with front-end (HTML5, JavaScript, CSS) and back-end (PHP/MySQL). Web application vulnerabilities and attacks will be introduced and explored with hands-on exercises on the range. Secure programming principles and practices will be introduced to avoid potential web application vulnerabilities and attacks.

(Recommended) Prerequisites: Intermediate programming skills in any language.

Topics (Tentative):

- The Web and Web Applications
- Client-side Web Development with HTML5, CSS, and JavaScript
- Server-side Web Application Development with PHP and Database (MySQL)
- Version control (git)
- Web Application Security: Vulnerabilities and Attacks with Hands-on Exercises

- Secure Web Programming Principles and Practices

Course Goals and Objectives:

Students will have the ability to:

- obtain the ability and skills to effectively use state-of-the-art techniques in web application development and security tools for the analysis, design, and implementation of secure web systems
- enhance students' programming skills to develop secure, robust, sustainable software products
- establish security analysis skills to identify and avoid potential web vulnerabilities
- identify the types of security threats and perform attacks in web applications and adopt corresponding countermeasures to prevent such attacks
- obtain the ability and skills to apply practical knowledge of security in designing and implementing web applications with contemporary programming technologies.

Course Outcomes:

Upon successful completion of this course, students will be able to:

- design, implement, and evaluate a web-based solution to meet a given set of computing requirements in the context of the program's discipline.
- understand the principle aspects of concepts of web application security
- identify and discuss the types of security threats and attacks in web applications and practical experience with hands-on exercises
- adopt corresponding countermeasures to identified threats in web applications and argue their effectiveness
- apply practical knowledge of security in designing and implementing web applications with contemporary technologies

Textbook

There is no required textbook for this course. Lecture handouts are provided for every class.
A good reference book (Free PDF online): *"Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites,"* Robin Nixon, O'Reilly Media, 6th Edition (2021).

Additional readings will be posted correspondingly before lectures.

Technology Requirements

You are required to have a laptop/computer to run a Linux virtual machine for hands-on exercises, e.g., labs, individual projects, and a team project. You need to install VirtualBox (https://www.virtualbox.org/) and install an Ubuntu 22.04 virtual machine in your laptop/computer. Alternatively, you will be provided with an online virtual machine to use during the semester. Detailed instructions will be provided in the second class.

The Shell and git video tutorials

This class heavily uses Shell and git. If you are not familiar with the Shell, we strongly recommend you do some hands-on exercises on the first topic, "The Shell": https://missing.csail.mit.edu/2020/course-shell/ from a free MIT course "The Missing Semester of Your CS Education."

We will provide hands-on exercises for git during the labs. However, if you have never used git before, it is strongly recommended you practice the git lecture https://missing.csail.mit.edu/2020/version-control/ in the above course to avoid any git issues during the code development.

Classroom Protocol and Requirements

Video recordings will be released around 5:00 PM or earlier on Tuesdays and Thursdays. To help you keep up with the class pace, you are required to watch the videos within 24 hours on Canvas. There will be a survey randomly placed in a video you must respond within 24 hours to earn 5% participation grade.

Multiple-choice quizzes in lectures

Each class will have instant multiple-choice quizzes, discussed and explained with answer keys in class and recorded in the videos that you should note for later references. These quizzes will be posted on Canvas, and it should take less than a few minutes to answer if you have attended the class or watched the video. To ensure you keep up with the class pace, you need to submit your answers by the end of each week. In total, you will earn 5% for these quizzes toward your total final grade percentage. These quiz questions will be randomly selected for the final exam online.

Advice and class communication

It is important for you to know that skipping any class/lecture video will not only prevent you from learning the class's topics and concepts but also put you behind the assignments and the next classes. If you must miss a class, e.g., due to illness, please inform the instructor and ensure that you will learn the materials and watch the video within an extended time granted by the instructor.

No email communication is encouraged in this class. Instead, please use the Discussions board on Canvas to ask questions or discuss with your peers. You are encouraged to ask questions on public channels so that your classmates may benefit from the answers. Everyone is encouraged to answer public questions from their classmates. You will earn a 1% bonus towards your final grade with active participation on the Discussion board.

If you have a question that you want to ask the instructor or the TA/grader directly, please use Chat on Canvas.

Submission Policy

Each assignment/submission has a **deadline**, which must be submitted on **Canvas -> Assignments** to be graded, i.e., submissions via email or other channels will NOT be graded. You need to submit your work before the deadlines so that you can gain the expected outcomes and feedback in a timely manner. To avoid last-minute issues, you need to start working on each submission when it is released, ideally during hands-on activities while watching lecture videos. By doing this, if you face any issues,

you should be able to seek support from the instructor and the TA to complete your work on time. Waiting until a later time or close to the deadlines to start any assignment will prevent you from being successful in this class; therefore, you need to plan your time carefully. To encourage you to do and submit your work earlier, there will be 0.5% bonus every 6 hours before the original deadline (up to 3% maximum bonus for each submission).

If you missed an original deadline, although it is strongly **NOT** encouraged, you would be allowed to make late submissions until the end of the semester (Sunday, July 21, 2024, at 11:59 pm). Every 24 hours late will be deducted 1% of the grade of the submission. You will get at least 70% credit for late submissions. However, you are strongly recommended to AVOID these late submissions. They will not only give you a low grade in this course but also prevent you from learning the concepts introduced in that assignment and the next related topics/assignments. Always talk to the instructor if you fall behind in any work/concepts/lectures. Experience in the past shows that missing or late assignment submissions will result in a very low grade in this class.

Academic Integrity

You are encouraged to discuss with your classmates the readings and ideas brought up in class, especially using the Slack workspace. But in all assignments to be graded as individual work you are expected to do your own written work. In the case of teamwork, all members of a group will be held responsible for the content of work turned in to satisfy group assignments. The instructor will keep a healthy eye out for possible plagiarism when reading your work.

It is best to express the ideas you use in your own words. In the case of both individual and group work, words or ideas that come from someplace or someone else must be cited accordingly.

For specific university policies concerning academic honesty, see the University's Academic Misconduct. Any behavior violating the Student Code of Conduct will not be tolerated and will be handled accordingly.

Assessments and Grading Policy

Your performance and course outcomes will be assessed via class/video participation, multiple-choice quizzes (weekly), hands-on lab exercises (4), hackathon exercises (4), individual projects (2), a team project, and a multiple-choice final exam.

Your final grade will depend (tentative)

- 5% of multiple-choice quizzes
- 5% of participation for graded discussions
- 20% of hands-on lab exercises
- 20% of hackathon exercises
- 25% of individual projects
- 15% of the team project
- 10% of the final exam

Your final grade will be based on your total grade percentage and will be scaled to a corresponding letter grade following the University's grade schema.

Course Schedule

Latest update: May 3, 2024.

The schedule is subject to change with 24-hour notice via the Isidore announcement. UC's academic calendar is available here->.

Week	Date	Topics	Assignment
1	05-07-2024	1. Introduction to Web application programming and the Course	
	05-09-2024	2. Development Environment Setup – **Lab 0**	
2	05-14-2024	3. Foundations of the Web – **Lab 1**	Lab 0 submission
	05-16-2024	4. Front-end web development with basic HTML and JavaScript – **Lab 2.1**	
3	05-21-2024	5. Front-end development with Ajax, CSS, and jQuery – **Lab 2.2**	Lab 1 submission
	05-23-2024	6. Integration with Web services/APIs & Front-end deployment on the Cloud **Individual Project 1 – A Professional homepage on GitHub.io**	
4	05-28-2024	7. Web Applications: Client-side Security and Server-side Technologies, Security and Standards	Lab 2 submission
	05-30-2024	8. Client-side Web & JavaScript Security **Hackathon 1: Cross-site scripting attacks and defenses**	Hackathon 1 submission on Sunday
5	06-04-2024	9. Web Application Development in PHP & MySQL	
	06-06-2024	10. (Insecure) Database Programming with PHP/MySQL – **Lab 3.1**	Project 1 submission on Sunday
6	06-11-2024	11. Database security & SQL Injection Attacks	**Team formation**
	06-13-2024	12. **Hackathon 2: SQL Injection Attacks**	Hackathon 2 submission on Sunday
7	06-18-2024	13. Secure Database Programming with Prepared Statements - **Lab 3.2**	
	06-20-2024	14. Cookies, Sessions and (Insecure) Session Authentication - **Lab 4.1**	Lab 3 submission on Sunday
8	06-25-2024	15. HTTPS and Secure Session Management **Lab 4.2** **Individual Project 2 – Full-stack Web Application**	**Team confirmation**
	06-27-2024	16. **Hackathon 3: Session Hijacking Attacks** (Insecure) Database Modification	Hackathon 3 submission on Sunday

9	07-02-2024	17. Database Modification & Team project tutorial and getting started	Lab 4 submission on Wednesday 11:59 PM
	07-04-2024	No class	
10	07-09-2024	18. **Hackathon 4: Cross-site Request Forgery Attacks** Secure Database Modification: CSRF Protection	Team project Sprint 0 submission
	07-11-2024	19. Database access control Team project Sprint 1 & tutorial	Hackathon 4 submission
11	07-16-2024	20. Displaying Data & Team project Sprint 2 & tutorial	Project 2 submission
	07-18-2024	21. WebSocket and real-time web applications	Team project Sprint 1 submission on Sunday
12	07-23-2024	22. Team project Sprint 3 & tutorial	
	07-25-2024	23. Course Review and Reflections	Team project Sprint 2 submission on Sunday
13	07-30-2024	Team project presentations	Team Project Submission on Wednesday 11:59 PM
	08-01-2024	Final Exam	