CIS 6614 - Advanced Software Systems Security, Fall 2022

Home Page of CIS 6614

This is the home page of the course Advanced Software Systems Security (CIS 6614) at UCF, as taught by Gary T. Leavens.

The web pages for this class are currently under construction, so the information should not yet be trusted.

Get the Latest News for CIS 6614

The latest news about CIS 6614 is available as announcements (and discussions) on webcourses.

Questions and Answers

We plan to use in-class discussion, email, and the discussion feature of Webcourses as our primary means of communication. So please check your email, and the Webcourses site for this course often. It's probably best to ask questions by email, but general discussions can be started on Webcourses, and we will make other student questions and answers available on Webcourses. So please check Webcourses to see if others have asked your question already.

Prompt, frequent and constructive feedback is essential to success in any endeavor. For this reason we strive to grade your work in a timely manner. We would also appreciate your feedback on the course. We hope that you can do this (even anonymously) using Webcourses, but if you want to be truly anonymous, please write it down on paper and leave it in the third floor mailroom for Gary. We'll also ask for feedback in class. Thanks for all your feedback.

Last modified Tuesday, August 16, 2022.

CIS 6614 - Advanced Software Systems Security, Fall 2022

Course Staff

The instructor for CIS 6614, is professor Gary T. Leavens.

Please don't hesitate to ask questions. The fastest way to receive an answer to a question is to email Gary Leavens at Leavens@ucf.edu or phone him at (407)-823-4758. You can also send messages in Webcourses@UCF, but those may not get an answer as quickly.

During the semester, we will hold "office hours" at the times listed below, and by appointment. Please let us know if none of these times work for you and we will consider revising the office hours schedule.

Staff Contact Information

Name	E-mail	Office	Office Hours	Phone
Gary T. Leavens	Leavens@ucf.edu	329 HEC (Bldg. 116), see also Webcourses@UCF	Tues. & Thurs. 10:30-11:30am and 4:30-5:00pm, + appts.	+1-407-823-4758

Suggestions Welcome

We welcome your suggestions on the course and how well (or poorly) we, the course staff, are serving you. You are welcome to make suggestions during or after class or to visit us during office hours or by appointment. You may also send us an email or call if you want to discuss something about the course in person.

Return to top

Last modified Tuesday, August 16, 2022.

CIS 6614 - Advanced Software Systems Security, Fall 2022

About CIS 6614

This page provides general information about CIS 6614 (Advanced Software Systems Security) at the University of Central Florida. The course's home page is www.cs.ucf.edu/~leavens/CIS6614/.

This page is organized as follows:

The course grading policy and syllabus are on separate web pages. Also on a separate page is our contact information.

Meetings

For class meetings, the time is as follows: **Tuesdays and Thursdays from 9:00 AM to 10:15 AM**. The meetings are in BA1, room O216.

Return to top

Instructional Modes

This course will be taught in several modes: in-person (P), Limited Attendance (RS), and Video (V), depending on the section you signed up for. (Section 0002 is in-person, section 0R01 is limited attendance, and section 0V91 is video mode.) According to UCF's Webcourses@UCF Support page these modalities are characterized as follows:

In Person (P)
: "Courses have required classroom attendance and meet on a regularly scheduled basis in-person. Students may encounter online, video, or adaptive elements as part of the instruction, thus requiring a computer."

Limited Attendance (RS)
: "Courses are primarily online in a blended format combining required in-person and online elements. In-person classroom activities may use up to 20% of the instructional time during the semester."

Video (V)
: "Courses are online with extensive use of digital video, which may be supplemented by additional online activity, projects, or exams."

If you are attending remotely, then internet access, a browser, email, and a microphone is required. A webcam is highly desirable.

Testing for V Mode Students

You will take tests in this class based on the section you are enrolled in. (For example, students in the face-to-face or reduced seat time sections are required to take tests in class during class time.)

Students in the Video (V) section are required to take their tests with an approved in-person proctor. It is the students' responsibility to find and secure a proctor. If you are in the V section, you should have filled out the COVE Form found here: https://tinyurl.com/cove-form and have given your proctor information to Sarah Moore, who is the testing coordinator for the College of Engineering. If you have not yet given her your proctor information, she will need it no later than 2 weeks before your first test. If you have questions regarding proctoring or who qualifies as a proctor, please email her at sarah.moore2@ucf.edu. She will be distributing the test materials to the appropriate proctors.

If you are registered in the wrong section, you should correct that by the add/drop deadline.

Return to top

COVID-19 and Illness Notification

(The following is mostly quoted from the faculty center for teaching and learning's web site.)

Students who believe they may have a COVID-19 diagnosis should contact UCF Student Health Services (407-823-2509) so proper contact tracing procedures can take place.

Students should not come to campus if they are ill, are experiencing any symptoms of COVID-19, have tested positive for COVID, or if anyone living in their residence has tested positive or is sick with COVID-19 symptoms. See the CDC guidance for COVID-19 symptoms.

Students should contact their instructor(s) as soon as possible if they miss class for any illness reason to discuss reasonable adjustments that might need to be made. When possible, students should contact their instructor(s) before missing class.

In Case of Faculty Illness

If the instructor falls ill during the semester, there may be changes to this course, including having a backup instructor take over the course. Please look for announcements or mail in Webcourses@UCF or Knights email for any alterations to this course.

Return to top

Course Textbooks

There are no required textbooks for this course. However...

Recommended Texts

The following books are recommended.

- Matt Bishop. *Computer Security: Art and Science*. Addison-Wesley Professional, 2002.
- Michael Howard, David LeBlanc, and John Viega. *24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them*. McGraw-Hill, 2010. ISBN: 978-0-07-162676-7.

We may use other material as described in the syllabus's bibliography.

Return to top

Accessibility

We are happy to help with accessibility issues. The procedure is outlined in the following statement (modified from the faculty center for teaching and learning's web site):

The University of Central Florida is committed to providing access and inclusion for all persons. Students who have accessibility issues "due to course design limitations should contact the professor as soon as possible. Students should also connect with Student Accessibility Services (SAS) (Ferrell Commons 185, sas@ucf.edu, phone 407-823-2371). For students connected with SAS, a Course Accessibility Letter may be created and sent to professors, which informs faculty of potential course access and accommodations that might be necessary and reasonable. Determining reasonable access and accommodations requires consideration of the course design, course learning objectives and the individual academic and course barriers experienced by the student. Further conversation with SAS, faculty and the student may be warranted to ensure an accessible course experience."

Return to top

Deployed Active Duty Military Students

Quoting from the faculty center for teaching and learning's web site:

"Students who are deployed active duty military and/or National Guard personnel and require accommodation should contact their instructors as soon as possible after the semester begins and/or after they receive notification of deployment to make related arrangements."

Return to top

Make-Up Assignments for Authorized University Events or Co-curricular Activities

Quoting from the faculty center for teaching and learning's web site:

"Students who represent the university in an authorized event or activity (for example, student-athletes) and who are unable to meet a course deadline due to a conflict with that event must provide the instructor with documentation in advance to arrange a make-up. No penalty will be applied. For more information, see UCF policy 4-401."

Return to top

Religious Observances

Quoting from the faculty center for teaching and learning's web site:

"Students must notify their instructor in advance if they intend to miss class for a religious observance. For more information, see UCF regulation 5.020."

Return to top

Course Description and Credit Hours

CIS 6614 is a 3 credit course entitled "Advanced Software Systems Security."

From the University of Central Florida Catalog: "CIS 6614 ECS-CS 3(3, 0) Advanced Software Systems Security: PR: CIS 4615. This course will cover various advanced topics on software threat modeling, secure software development life cycle, common security issues, and mitigations in modern software operation. Odd Fall, Even Fall"

Explanation

Software is said to be *secure* when it can only be used as intended, and cannot be used to extract confidential information, undermine integrity, or facilitate unauthorized access. A secure software development process helps to ensure the security of software products. In particular it avoids known problems that could be used to attack a software product.

Threat modeling is the process of assessing what the most likely and important threats are to a computer system. This involves assessing what is important to clients of the system and what are the likely attacks (usually based on prior, known attacks).

Motivation for the Course Objectives

Software controls much of our modern world and impacts nearly all aspects of our lives. More and more physical devices that we depend on (such as automobiles and cell phones) are controlled by complex software systems. These software systems may allow attackers many different ways to undermine the system's security in ways that would benefit them and/or adversely affects the system's legitimate users. When such devices and information that they store become more important, then it becomes more important to secure them against attacks.

Software can be vulnerable to attacks either because it uses inherently insecure library functions (such as `gets` in C) or operating system calls, or because it has bugs. Analysis of the software before it is run (called *static analysis*) or monitoring of a system's execution while it is running (called *dynamic analysis*) can reveal insecurities before an attacker has a chance to cause (a great amount of) damage. A secure software development process often emphasizes static analysis (e.g., with code reviews) and static analysis, but can also use dynamic analysis as a kind of testing strategy. All of these techniques will be subjects for our study in this class.

Motivation for the Course Plan

To secure software systems, professionals will need to understand both static and dynamic analysis techniques and be able to put together a strategy to use these techniques in a cost-effective manner to secure a software system. Both research and practice in this area involve building tools to help with either threat modeling or mitigation.

Therefore, an important part of this course will be building tools to aid either threat modeling or static or dynamic analysis of software. Thus a major component of this course will be a team-based effort to build such tools.

Return to top

Course Learning Objectives

The objectives for this course are divided into two parts: a set of essential objectives, and a set of enrichment objectives. The essential objectives will be helpful for your career; hence they lead to the course's essential outcomes that we want to help you master. The enrichment objectives are less important for the course, but lead to enrichment outcomes that you are encouraged to explore both for their own sake and because learning more about those will help deepen your understanding of the essential objectives. The enrichment outcomes may also lead to avenues for research in software systems security.

Essential Objectives

In one sentence, this course's main objective is you will be able to supervise an enterprise's software system safety.

In more detail the essential objectives for this course are that you will be able to:

- **[Strategize]** plan a strategy to assure that an enterprise's software systems and/or products are secure from likely and important threats.
- **[Design]** Design a set of mitigations to the likely and important threats to software security and an architecture for tools to support that design.
- **[Implement]** Efficiently and correctly implement a tool to support a software system's secure development process.
- **[Evaluate]** Evaluate the adequacy of a threat model and mitigations to protect against those threats.

Enrichment Objectives

Enrichment objectives could be multiplied without limit, but the following seem most important, especially in relation to research in Computer Science and the Computer Science graduate program.

The course's enrichment objectives are that you will be able to:

- **[Teamwork]** Effectively participate in a team that can assure the security of an enterprise's software systems.
- **[Writing]** Convincingly and clearly write about the strategies, designs, architectures, and tool implementations in a way that could be published.

Course Learning Outcomes

This course's learning outcomes are divided into two parts: a set of essential outcomes, and a set of enrichment outcomes. The essential outcomes are designed to support this course's essential learning objectives, and thus to be helpful for your career as a computer scientist or software engineer; hence we want to help you to master them. They also form the basis for grading and assessment of your learning. The enrichment outcomes are not used directly for assessment. However, you are encouraged to explore topics related to the enrichment outcomes both for their own sake and because learning more about those will help your performance relative to the essential outcomes.

This course's outcomes are linked to this course's objectives (above). The links to this course's objectives are shown in references that look like this: [Strategize].

Essential Outcomes

In one sentence, this course's main expected learning outcome is that you will be able to effectively design and implement an enterprise's strategy for creating secure software systems, including building some of the necessary tools. [Strategize] [Design] [Implement]

In more detail, the essential objectives for this course are that you will be able to:

- **[Plan]** Plan a strategy for protecting a software system against important threats. [Strategize].
- **[Architect]** Create a plan for processes and tools that will protect a software system against important threats. [Design] [Evaluate].
- **[Build]** Create a tool that realizes an important part of a strategy to either reduce the possibility of important attacks or protect a software system from attacks. [Implement] [Evaluate].
- **[Judge]** Give a well-reasoned, critical judgment about the importance and implications (for security) of a strategy, architecture, or tool implementation, especially in terms of how it will the system's users and mitigate the important threats. [Evaluate].

Enrichment Outcomes

Enrichment outcomes could be multiplied without limit, but the following seem most important, especially in relation to research in software security.

The course's enrichment outcomes are that you will be able to:

- **[Prioritize]** Clearly explain which threats are most important to an enterprise (and its users) and the likelihood of the threat being realized (given other mitigations) [Strategize] [Evaluate] [Writing].
- **[Collaborate]** Work on a team to discover, evaluate, and communicate strategies, architectures, and tool designs [Teamwork].

Return to top

Prerequisites

The formal prerequisite in the University of Central Florida catalog is "CIS 4615 or C.I."

See the professor if you have questions about the prerequisites.

Return to top

Acknowledgments

Many thanks to David Mohaisen for discussions about this course.

Thanks to Curtis Clifton (now at Apple) for his initial work on the HTML for these web pages, which I have adapted from another course, and his style sheets, which I have also adapted.

Return to top

Last modified Tuesday, August 16, 2022.

CIS 6614 - Advanced Software Systems Security, Fall 2022

Grading Policy

This page defines the course's grading policy. In essence, everything to do with grading, including ethical issues. See the course's about page for a description of the course itself.

The material on this page is organized as follows:

1. Access to Grades
2. Participation
3. Late Homework Problems Policy
4. Final Grades
5. Letter Grades
6. Cheating and Plagiarism
7. Teamwork, Cooperation, and Discussions

Access to Grades

You can access your grades through Webcourses@UCF.

Late Homework Problems Policy

The late policy for homework problems is designed to encourage you to:

- hand in one "good" version of each problem,
- hand it in on-time if possible,

but to hand it in eventually in any case. To allow the staff to grade what you have handed as soon as possible, only one version of any problem, the first you hand in, will be accepted.

Like all homework, late homework problems must turned in using webcourses@UCF. Email will *not* be accepted.

Note that you can turn in just the problems that you have finished; we don't require you to turn in entire homeworks at once. The late penalties only apply to those problems you turn in late.

We do give partial credit for homework, so you will have to balance the gain from waiting to get a good version of a problem and the loss from handing the problem in late. In general, we encourage you to hand in a good version of each problem, but if you are late (and have been trying), consider that as a sign that you need help on the concepts, and get help from us!

Homework problems that are late receive points based on the following table.

When Handed In	Percentage Penalty
by 24 hours after the time the homework problem is due	5%
by 48 hours after the time the homework problem is due	10%
by 72 hours (3 days) after the time the homework problem is due	20%
by 96 hours (4 days) after the time the homework problem is due	40%
later, or during last week of classes	100%

For example, if a homework problem is due on Tuesday at 11 PM but you turn it in by Wednesday at 11 PM, you will have 5% of what would have been your score subtracted; thus if the problem was 25 points, and you earned 20 of them, your score would be recorded as 19 points (= 20 - (0.05*20)), due to the 5% penalty. If you turned the same thing in by Thursday at 11 PM, your penalty would be 10%, so your score would be recorded as 18 points.

Absolutely no credit for late homework problems will be given during the last calendar week of classes (or later!), or for homework problems turned in later than 4 days after the time the homework is due.

Final Grades

Final grades for the course will be based on your performance in the homework and project assignments during the semester. You can anticipate between 4-6 homeworks. The project have a draft and a final report. There are no exams.

Final grades are based on the following weights.

Component	Weight
Homework assignments	30%
Exam(s)	30%
Project	40%

Return to top

Letter Grades

Your grade is independent of anyone else's grade in this class. That is, we do not grade on a curve, and everyone can get an *A*. Our purpose in grading is to uphold a standard of quality and to give you feedback: it is not to rank students.

Although we will not always make fine distinctions in points the nominal minimum standards are given by the following table. (We will only assign +/- grades for borderline cases.)

Percentage	Grade
90%	A
80%	B
70%	C
55%	D
less	F

Return to top

Cheating and Plagiarism

The simple rule of thumb is:

Never give or use someone else's presentation or written answers.

Such exchanges are definitely cheating and not cooperation. This includes taking answers from the web.

If you use reference materials to solve a problem, you must give a citation. Furthermore, use of more than a few words from any source (including the course texts) must be properly set off with quotation marks ("...") or in an italicized block quote and a proper citation given. This definitely includes material from the web. Not attributing material as described above is *plagiarism*, which is a form of cheating. This includes arranging

sentences from other sources without proper use of quotation marks and citations for each quote. We take plagiarism quite seriously, so note this policy well.

Here's a standard statement (from the UCF FCTL web site) about our use of turnitin.com:

"To detect cheating we may use turnitin.com, an automated system that can quickly and easily compare each student's assignment with billions of web sites, as well as an enormous database of student papers that grows with each submission. Accordingly, you may be expected to submit assignments in electronic format. After the assignment is processed, as an instructor I receive a report from turnitin.com that states if and how another author's work was used in the assignment. For a more detailed look at this process, visit http://www.turnitin.com."

If we catch you cheating on a test or exchanging code or written answers, you will get no credit for that test or homework, and you may be reported to the Director of the Office of Student Rights and Responsibilities. Read the section on academic dishonesty/cheating in the *Golden Rule*.

If you honestly believe that certain problems are too much busy work, then bring it to the instructor's attention; or failing that, only do the part of the problem that you think you need to do to learn the material and explain that to us.

Return to top

Teamwork, Cooperation, and Discussions

Some assignments in this class will be done in teams. When working with other team members, you are expected to cooperate and collaborate and that is not considered cheating.

You are encouraged to discuss this class with other students, including those outside your team. Such discussions about ideas are not cheating, whereas the exchange of finished presentations or written work with people outside your team is cheating. However, when you have more than casual discussions about assignments, you *must* cite the other person as described below.

When you cooperate on ideas or other work that goes into a final product for an assignment, you must cite the other people you worked with as follows. This must be done *for each assignment* on which you cooperate or collaborate.

- If you discussed ideas jointly, but wrote up a final product for the assignment independently, then each person should include a note with that problem's solution such as "this assignment was developed using ideas and discussions with Alyssia P. Hacker." Each person's final assignment receives a grade independently of the other's; there is no bonus or penalty for such a citation.

If you have someone else produce final products that you turn in as your own work, then that is cheating. Such cheating will be dealt with as described above. It should be clear that you will learn less by such exchanges of finished work.

Be careful, not to get involved in an unequal collaboration, where you are doing less work than someone else. Part of what you need to do to learn the material is to struggle with it; if you deny yourself that struggle, you will learn less and remember what you learned less. So beware of this trap.

Also, as a kindness to your classmates, you should terminate an unequal collaboration where you are doing more than the other person. The other person will learn the material better if you help them but don't collaborate so closely. In this case it's better to help them only by discussing problems with them, and not by jointly collaborating on solutions.

If you have questions about the details of cooperation vs. cheating, please see the professor.

Return to top

Last modified Monday, August 22, 2022.

This web page is for CIS 6614 at the University of Central Florida. The details of this course are subject to change as experience dictates. You will be informed of any changes. Please direct any comments or questions to Gary T. Leavens at Leavens@ucf.edu. Some of the policies and web pages for this course are quoted or adapted from other courses I have taught, in partciular, COP 4020.