FAU

FLORIDA
ATLANTIC
UNIVERSITY

NEW COURSE PROPOSAL
Undergraduate Programs

Department Electrical Engineering and Computer Science

College Engineering and Computer Science
(*To obtain a course number, contact* ***erudolph@fau.edu***)

UUPC Approval ____________
UFS Approval ____________
SCNS Submittal ____________
Confirmed ____________
Banner Posted ____________
Catalog ____________

Prefix CAI

Number 4800

(*L = Lab Course; C = Combined Lecture/Lab; add if appropriate*)

Lab Code

Type of Course Lecture

Course Title Security Engineering with Generative AI

Credits (*See Definition of a Credit Hour*) 3

Effective Date (*TERM & YEAR*) Fall 2026

Grading (*Select One Option*)

Regular ●

Sat/UnSat ○

Course Description (*Syllabus must be attached; see Template and Guidelines*)
Students design and build security tools powered by generative AI. The course covers programming with large language models, building AI-powered applications and autonomous agents, and securing AI systems against adversarial attacks. Students then apply generative AI to cybersecurity problems including code generation, vulnerability discovery, reverse engineering, exploitation, threat intelligence, incident response, and social engineering defense. Students may not enroll in CAI 4800 if they have already taken CAI 5804.

Prerequisites, with minimum grade* (COP 3530C or COP 3410C) and CNT 4403

Corequisites

Registration Controls (*Major, College, Level*) Not available to students who have completed CAI 5804.

Default minimum passing grade is D-. ***Prereqs., Coreqs. & Reg. Controls are enforced for all sections of course***

WAC/Gordon Rule Course

□ **Yes** ☑ **No**

WAC/Gordon Rule criteria must be indicated in syllabus and approval attached to proposal. See WAC Guidelines.

Intellectual Foundations Program (General Education) Requirement
(*Select One Option*)

None

General Education criteria must be indicated in the syllabus and approval attached to the proposal. See Intellectual Foundations Guidelines.

Minimum qualifications to teach course

PhD in EECS disciplines, MS in related discipline and professional experience relevant to the course

Faculty Contact/Email/Phone

Hari Kalva, hkalva@fau.edu

List/Attach comments from departments affected by new course

Contacted Math and ITOM

Approved by		***Date***
Department Chair	Hari Kalva	3/16/26
College Curriculum Chair	Yalan Liu	3/17/26
College Dean		3/02/26
UUPC Chair		
Undergraduate Studies Dean		
UFS President		
Provost		

Email this form and syllabus to mjenning@fau.edu seven business days before the UUPC meeting.

FLORIDA ATLANTIC UNIVERSITY

Department of Electrical Engineering and Computer Science

College of Engineering and Computer Science

CAI 4800

Security Engineering with Generative AI

3 Credits | Fall 2026

Instructor Information

TBD

TA Information

TBD

Course Description

Students design and build security tools powered by generative AI. The course covers programming with large language models, building AI-powered applications and autonomous agents, and securing AI systems against adversarial attacks. Students then apply generative AI to cybersecurity problems including code generation, vulnerability discovery, reverse engineering, exploitation, threat intelligence, incident response, and social engineering defense. Students may not enroll in CAI 4800 if they have already taken CAI 5804.

Prerequisites

(COP 3530C or COP 3410C) and CNT 4403

Instructional Method

In-person and/or hybrid

Required Texts/Materials

None. All materials needed will be provided and freely available to students.

Course Objectives / Student Learning Outcomes

Upon completing this course, students will be able to:

1. Design and build AI-powered security applications using large language models, autonomous agents, and tool integration frameworks.

2. Secure AI-powered applications by identifying and exploiting vulnerabilities (prompt injection, jailbreaking, data exfiltration) and implementing defenses against them.

3. Apply generative AI tools to cybersecurity problems including code generation, vulnerability discovery, reverse engineering, threat intelligence, incident response, and social engineering defense.

4. Evaluate the reliability, limitations, and failure modes of AI-generated outputs in security contexts.

5. Design, implement, and present an original AI-powered security tool that addresses a cybersecurity problem of the student's choosing.

Topical Outline

The course includes eight hands-on labs spread across the semester. Each lab spans one to two weeks depending on depth.

Topic	What Students Do
Environment setup, API access, model comparison, LLM programming	Set up cloud and LLM accounts. Deploy an Ollama server for testing open models locally. Access multiple LLMs programmatically from Python. Write scripts that send prompts, parse outputs, and chain calls. Compare model responses to security-related prompts and benchmark performance across a suite of tasks.
LangChain fundamentals, document loading, RAG, agents, tool use	Build applications using LangChain prompt templates, chains, and output parsers. Load and chunk security documents. Build a retrieval-augmented generation (RAG) application that answers questions from a security knowledge base. Then build autonomous agents that select and use tools to accomplish tasks. Create a custom LangChain agent that performs a security-related function of the student's design.
Model Context Protocol, OWASP LLM Top 10, prompt injection, jailbreaking, defensive techniques	Build MCP-based agent applications that connect LLMs to external tools and data sources. Create a custom MCP agent. Then attack it: perform prompt injection, jailbreaking, and data exfiltration against the application. Walk through the OWASP Top 10 for LLM applications and security-test the agent against each category. Apply defenses including input validation, output filtering, and system prompt hardening. Document findings and remediation steps.
Code generation, coding agents, generated code review	Use LLMs for code generation, editing, and translation. Use AI-powered coding agents to build applications. Generate a security-relevant application with a coding agent, then evaluate the security of the generated code. Identify common weaknesses in AI-generated code.

Topic	What Students Do
Command generation, configuration generation, deployment automation, output validation	Use LLMs to generate shell commands, firewall rules, and infrastructure configurations. Build an agentic application that generates configurations and deploys them. Evaluate the correctness and security of generated outputs. Test whether the AI introduces misconfigurations or insecure defaults.
Code documentation, reverse engineering, static analysis, vulnerability scanning	Use LLMs to document, summarize, and analyze code. Apply LLMs to reverse engineering tasks including decompiled code analysis. Build an LLM application to aid in analyzing an unfamiliar codebase. Then use LLMs to identify vulnerabilities in source code and running applications. Use LLM-integrated vulnerability scanning tools. Compare LLM-assisted discovery to manual methods.
Exploit generation, patch generation, log analysis, alert triage, incident reporting	Use LLMs to generate exploits for known vulnerabilities and to produce patches. Build an LLM application that aids in exploiting a vulnerable target and generating a fix. Then apply LLMs to incident response: parse and summarize security logs, triage alerts, and generate incident reports. Build a pipeline that ingests log data and produces actionable summaries.
Threat intelligence, IOC extraction, phishing generation, social engineering detection	Use LLMs to analyze threat intelligence reports, extract indicators of compromise (IOCs), and correlate threat data across sources. Then examine both sides of social engineering: use LLMs to generate phishing emails and pretexting scripts, and build detection tools that identify AI-generated social engineering attempts. Evaluate offensive and defensive applications.

Course Evaluation Method

Component	Weight
Attendance and participation	10%
Lab notebooks	25%
Homework screencasts	35%
Final project	30%

Lab Notebooks

Each lab has an accompanying notebook. Students complete the lab exercises and maintain a lab notebook in a Google Doc that documents their progress, including responses to embedded prompts. Screenshots must include the student's FAU ID. Notebooks are exported as PDF with a generated table of contents and submitted via the student's private git repository. Graded on neatness, completeness, and inclusion of FAU ID in screenshots.

Homework Screencasts

For selected labs, students build their own application that applies generative AI to a security problem and record a screencast walkthrough. Screencasts capture the student's authentic engineering workflow: how they approach the problem, how they interact with AI tools, how they test and debug, and how they evaluate the results. This format verifies that students can independently design, build, and reason about their work. Instructions and rubrics are provided with each assignment.

Final Project

Students design and implement an AI-powered security application that addresses a cybersecurity problem of their choosing. The project draws on the frameworks, techniques, and tools from the course. Students present their tool and demonstrate it in a final screencast. Details are provided in the second half of the semester.

Due Dates and Late Submissions

Submit what you have on time. Late work receives a 20% deduction and must be placed in the late directory of the student's git repository. Late work is graded at the end of the last week of class and will not be accepted after that point.

Course Grading Scale

Letter Grade	Percentage
A	94 – 100%
A-	90 – 93%
B+	87 – 89%
B	83 – 86%
B-	80 – 82%
C+	77 – 79%
C	73 – 76%
C-	70 – 72%
D+	67 – 69%
D	63 – 66%
D-	60 – 62%
F	Below 60%

Course Policies

Attendance Policy

Students are expected to attend all of their scheduled University classes and to satisfy all academic objectives as outlined by the instructor. The effect of absences upon grades is determined by the instructor, and the University reserves the right to deal at any time with individual cases of non-attendance. Students are responsible for arranging to make up work missed because of legitimate class absence, such as illness, family emergencies, military obligation, court-imposed legal obligations or participation in University-approved activities. Examples of University-approved reasons for absences include participating on an athletic or scholastic team, musical and theatrical performances and debate activities. It is the student's responsibility to give the instructor notice prior to any anticipated absences and within a reasonable amount of time after an unanticipated absence, ordinarily by the next scheduled class meeting. Instructors must allow each student who is absent for a University-approved reason the opportunity to make up work missed without any reduction in the student's final course grade as a direct result of such absence.

Code of Academic Integrity

Students at Florida Atlantic University are expected to maintain the highest ethical standards. Academic dishonesty is considered a serious breach of these ethical standards, because it interferes with the university mission to provide a high quality education in which no student enjoys an unfair advantage over any other. Academic dishonesty is also destructive of the university community, which is grounded in a system of mutual trust and places high value on personal integrity and individual responsibility. Harsh penalties are associated with academic dishonesty. For more information, see University Regulation 4.001.

Disability Policy

In compliance with the Americans with Disabilities Act Amendments Act (ADAAA), students who require reasonable accommodations due to a disability to properly execute coursework must register with Student Accessibility Services (SAS) and follow all SAS procedures. SAS has offices across three of FAU's campuses – Boca Raton, Davie and Jupiter – however disability services are available for students on all campuses. For more information, please visit the SAS website at www.fau.edu/sas/.

Religious Accommodation

In accordance with Florida law, students have the right to reasonable accommodations for religious observances. See University Regulation 2.007.

Policy on Recording of Lectures

Students may record class lectures for personal educational use. Recording of student presentations, class discussions, labs, and private conversations is prohibited. Recordings may not be published or shared without written consent.

Grade Appeal Process

Students may request a review of a final course grade if there was a computational or recording error, the grading used non-academic criteria, or there was a gross violation of the instructor's grading system. See University Regulation 4.002.

Counseling and Psychological Services (CAPS)

Life as a university student can be challenging physically, mentally and emotionally. Students who find stress negatively affecting their ability to achieve academic or personal goals may wish to consider utilizing FAU's Counseling and Psychological Services (CAPS) Center. CAPS provides FAU students a range of services – individual counseling, support meetings, and psychiatric services, to name a few – offered to help improve and maintain emotional well-being. For more information, go to http://www.fau.edu/counseling/

Artificial Intelligence Preamble

FAU recognizes the value of generative AI in facilitating learning. However, output generated by artificial intelligence (AI), such as written words, computations, code, artwork, images, music, etc., for example, is drawn from previously published materials and is not your own original work. FAU students are not permitted to use AI for any course work unless explicitly allowed to do so by the instructor of the class for a specific assignment. [Policy 12.16 Artificial Intelligence]. Class policies related to AI use are decided by the individual faculty. Some faculty may permit the use of AI in some assignments but not others, and some faculty may prohibit the use of AI in their course entirely. In the case that an instructor permits the use of AI for some assignments, the assignment instructions will indicate when and how the use of AI is permitted in that specific assignment. It is the student's responsibility to comply with the instructor's expectations for each assignment in each course. When AI is authorized, the student is also responsible and accountable for the content of the work. AI may generate inaccurate, false, or exaggerated information. Users should approach any generated content with skepticism and review any information generated by AI before using generated content as-is.

If you are unclear about whether or not the use of AI is permitted, ask your instructor before starting the assignment.

Failure to comply with the requirements related to the use of AI may constitute a violation of the Florida Atlantic Code of Academic Integrity, Regulation 4.001.

Proper Citation: If the use of AI is permitted for a specific assignment, then use of the AI tool must be properly documented and cited. For more information on how to properly cite the use of AI tools, visit https://fau.edu/ai/citation

Artificial Intelligence Policy Specific to this Course

This course requires extensive use of generative AI tools as part of the coursework. AI-generated output is not the student's original work. Students must document all AI tool usage, including prompts and outputs. Students are responsible and accountable for the content of all submitted work, including any AI-generated content. AI may produce inaccurate, false, or insecure output; students must review and verify all generated content before use. See FAU Policy 10.16.

Title IX Statement

In cases involving allegations of sexual misconduct, students are encouraged to report the matter to the University Title IX Coordinator in the Office of Civil Rights and Title IX (OCR9). Faculty who become aware of such allegations are expected to report them. More information is available at www.fau.edu/ocr9/title-ix/.