Cryptography/Secure Coding Theory and Application

2023-2024 Catalog

[ARCHIVED CATALOG]

CSIA 245 - Cryptography/Secure Coding Theory and Application

PREREQUISITES/COREQUISITE: SDEV 140 - Introduction to Software Development.

PROGRAM: Cyber Security/Information Assurance
CREDIT HOURS MIN: 3
LECTURE HOURS MIN: 3
DATE OF LAST REVISION: Fall, 2014

Students will learn about cryptography as an indispensable resource for implementing strong security in real-world applications. Students will learn why conventional crypto schemes, protocols, and systems are vulnerable. The foundations of cryptography using simple mathematical terms including probability, information theory, computational complexity, number theory, and algebraic approaches will be covered. The students will assess the strength, security, and efficiency of encryption standards and use formal methods to assess their levels of security and efficiency. Discussions on application of security measures and the challenges associated with each will be covered. Part of a layered security approach begins with implementing good coding practices. Students will cover the steps for writing, testing, and deploying robust and security-enhanced code. Subjects cover include threat modeling, secure code lifecycle, current tools used in the industry, and software maintenance and incident preparedness.

MAJOR COURSE LEARNING OBJECTIVES: Upon successful completion of this course the student will be expected to:

1. Discuss and demonstrate classical encryption techniques and ciphers.
2. Discuss the basic concepts of probability, random variables and their probability distribution, information theory, and redundancy in natural languages.
3. Discuss and demonstrate different types of algorithms used in coding.
4. Discuss and demonstrate the use of various private and public key technologies.
5. Discuss everyday uses of encryption.
6. Demonstrate authentication systems.
7. Discuss various types of Cryptographic Attacks.
8. Understand the importance of information security in software development.
9. Discuss current industry standards, tools, and security practices in software development.
10. Examine the principles and goals of secure and quality coding.
11. Discuss and design an application guide.
12. Understand, analyze, and interpret customer requirements.
13. Create design diagrams or artifacts based off of customer requirements.
14. Analyze design diagrams and artifacts for weaknesses and apply appropriate security measures.
15. Apply proper secure coding and testing techniques.
16. Discuss software maintenance and incident preparedness.

COURSE CONTENT: Topical areas of study include -

- Classical encryption techniques
- Types of ciphers
- Logons, logins and passwords
- Probability, random variables, redundancy in natural languages
- Breaking ciphers
- Algorithms used in coding
- Public and private key systems
- Cryptographic attacks
- Authentication systems
- Current industry secure coding standards
- Application guide design
- Secure and quality coding principles
- Customer needs assessment
- Design diagrams and artifacts
- Code testing techniques
- Error handling
- Application logging
- Threat modeling
- Software maintenance
- Incident preparedness
- Source Code Control
- Kerberos, SSH, Radius and TACACS+

Course Addendum - Syllabus (Click to expand)