



SACRAMENTO
STATE

Course Change Proposal Form A



Academic Group (College): Engineering and Computer Science	Academic Organization (Department): Computer Science	Date: December 20, 2007
Type of Course Proposal: New <input checked="" type="checkbox"/> Change <input type="checkbox"/> Deletion <input type="checkbox"/>	Department Chair: Du Zhang	Submitted by: Du Zhang
Does this course fulfill a requirement for single-subject or multiple subject credential students? Yes <input type="checkbox"/> No <input checked="" type="checkbox"/>	For Catalog Copy: Yes <input checked="" type="checkbox"/> No <input type="checkbox"/> CCE (Extension): Yes <input type="checkbox"/> No <input checked="" type="checkbox"/>	Semester Effective: Fall <input checked="" type="checkbox"/> Spring <input type="checkbox"/> , 2008

This course replaces experimental course Subject Area (prefix) and Catalog Nbr (course number):

Change from:

Subject Area (prefix) & Catalog Nbr (course no.):	Title:	Units:
--	---------------	---------------

Change to:

Subject Area (prefix) & Catalog Nbr (course no.): CSC 152	Title: Cryptography	Units: 3
---	----------------------------	-----------------

JUSTIFICATION:

This new course is integral to the creation of the Undergraduate Certificate Program in Information Assurance and Security (Form B in process) and will help Sac State to renew its designation by the NSA/DHS as a Center of Academic Excellence in Information Assurance Education.

NEW COURSE DESCRIPTION: (Not to exceed 80 words, and language should conform to catalog copy. See <http://www.csus.edu/acaf/univmanual/crspsl.htm> - Guidelines for Catalog Course Description)

Introduction to design and analysis of cryptographic systems. Symmetric cryptography: Block ciphers and secure hash functions. Asymmetric cryptography: Key exchange and public-key systems. Authentication and encryption in an adversarial model. Simple cryptanalysis. Protocol design and analysis.

Note:

Prerequisite: STAT 50, CSC 60, CSC 130
Enforced at Registration: Yes No

Corequisite:
Enforced at Registration: Yes No

CAN (California Articulation Number):

Graded: Letter Credit/No Credit **Instructor Approval Required?** Yes No

Course Classification (e.g., lecture, lab, seminar, discussion):
C 04 **Title for CMS (not more than 30 characters):**
CRYPTOGRAPHY

Cross Listed? Yes No **If yes, do they meet together and fulfill the same requirement, and what is the other course.**

How Many Times Can This Course be Taken for Credit? Once

Can the course be taken for Credit more than once during the same term? Yes No

FOR NEW COURSE PROPOSALS OR SUBSTANTIVE CHANGES ONLY:

Description of the Expected Learning Outcomes: Describe outcomes using the following format: "Students will be able to: 1), 2), etc."
See the example at <http://www.csus.edu/acaf/example.htm>

Students will have a...

Thorough understanding of:

- Adversarial model in cryptography.
- Properties of cryptographic primitives: block ciphers, secure hash functions, universal hash functions, public-key cryptosystems.
- Encryption modes-of-operation.
- Key-exchange and distribution goals, assumptions and protocols.
- Cryptographic programming using a cryptographic toolkit.

Basic Understanding of:

- Confusion/diffusion techniques.
- Use of reductions to establish security guarantees.
- Mathematics of public-key cryptography.
- RSA and Diffie-Hellman problems.
- Certificates and the public-key infrastructure.
- A major network-security protocol (e.g. SSL/TLS or IPSec).

Exposure to:

- Goals and techniques of cryptanalysis.
- Programming to avoid security vulnerabilities.

**Attach a list of the required/recommended course readings and activities [Note: it is understood that these are updated and modified as needed by the instructor(s).] This attachment should be forwarded only to your Dean's office, not Academic Affairs.

Assessment Strategies: A description of the assessment strategies (e.g., portfolios, examinations, performances, pre-and post-tests, conferences with students, student papers) which will be used by the instructor to determine the extent to which students have achieved the learning outcomes noted above:

Laboratory projects and examinations.

For whom is this course being developed?

Majors in the Dept Majors of other Depts Minors in the Dept General Education Other

Is this course required in a degree program (major, minor, graduate degree, certificate)? Yes No

If yes, identify program(s): **Proposed Undergraduate Certificate Program in Information Assurance and Security.**

Does the proposed change or addition cause a significant increase in the use of College or University resources (lab room, computer facilities, faculty, etc.)? Yes No

If yes, attach a description of resources needed and verify that resources are available.

Indicate which department or programs will be affected by the proposed course (if any). Criminal Justice (possibly)

The Department Chair's signature below indicates that affected programs have been sent a copy of this proposal form.

Approvals: If proposed change, new course or deletion is approved, sign and date below. If not approved, forward without signing to the next reviewing authority, and attach an explanatory memorandum to the original copy.

Signatures:

	Date
Department Chair:	12/21/2007
College Dean or Associate Dean:	2/22/08
CPSP (for school personnel courses ONLY)	
Associate Vice President and Dean for Academic Programs	

Distribution: Academic Affairs (original), Department Chair and College Dean. Dean's office to send original after approval to Academic Affairs, at mail zip 6016. An electronic copy must also be sent.

COURSE DESCRIPTION

Dept., Number	CSC 152	Course Title	Cryptography
Semester hours	3	Course Coordinator	Ted Krovetz

Catalog Description

Introduction to design and analysis of cryptographic systems. Symmetric cryptography: Block ciphers and secure hash functions. Asymmetric cryptography: Key exchange and public-key systems. Authentication and encryption in an adversarial model. Simple cryptanalysis. Protocol design and analysis. Prerequisites: STAT 50, CSC 60, CSC 130.

Textbook Ferguson and Schneier, Practical Cryptography, Wiley, 2003.
Viega, Messier and Chandra, Network Security with OpenSSL, O'Reilly, 2002.

References

Menesez, van Oorschot and Vanstone, Handbook of Applied Cryptography, CRC 1996 (available free at <http://www.cacr.math.uwaterloo.ca/hac/>).
Nigel Smart, Cryptography, An Introduction (Second Edition), self-published, 2007 (available free at http://www.cs.bris.ac.uk/~nigel/Crypto_Book/).

Course Goals

To give students an operational understanding of basic cryptography, including:

1. The ability to safely integrate cryptographic functionality into computer programs using a cryptographic programming toolkit; and
2. The ability to discuss cogently major cryptographic primitives, their design principles, common uses, security goals and methods of analysis.

Prerequisites by Topic

Thorough understanding of:

- Programming in a high-level language.
- Elementary discrete probability: Random variables, expectation, conditional probability.

Basic Understanding of:

- Programming in C or C++, including the use of pointers.
- Proof methods, especially contradiction.
- Analysis of non-recursive algorithms.

Exposure to:

- Basic counting principles.

Major Topics Covered in the Course

1. Goals of cryptography; adversarial model; examples of achieving goals using a random function. (3 hours.)
2. Historical ciphers; confusion/diffusion primitives; cryptanalysis. (3 hours.)
3. Block ciphers; random permutations; modeling block ciphers as a random permutation; cryptanalysis of simple block-ciphers. (3 hours.)
4. Block cipher encryption; modes of operation; attacks on encryption schemes; reductionist cryptography. (6 hours.)
5. Hash functions, cryptographic and universal; properties; constructions. (6 hours.)
6. Message and password authentication; authenticated encryption. (3 hours.)
7. Mathematics of public-key cryptography: Groups, Euclid's algorithm. Complexity theoretic assumptions. (3 hours.)
8. Public-key cryptography; Diffie-Hellman and RSA; encryption and signature schemes; certificates. (6 hours.)
9. Network protocols; breaking poor protocols; proving good protocols; key distribution. (3 hours.)
10. Case studies (e.g., SSH, SSL/TLS, IPsec). (3 hours.)
11. Cryptographic programming; cryptographic toolkits; secure programming. (6 hours.)

Outcomes

Thorough understanding of:

- Adversarial model in cryptography.
- Properties of cryptographic primitives: block ciphers, secure hash functions, universal hash functions, public-key cryptosystems.
- Encryption modes-of-operation.
- Key-exchange and distribution goals, assumptions and protocols.
- Cryptographic programming using a cryptographic toolkit.

Basic Understanding of:

- Confusion/diffusion techniques.
- Use of reductions to establish security guarantees.
- Mathematics of public-key cryptography.
- RSA and Diffie-Hellman problems.
- Certificates and the public-key infrastructure.
- A major network-security protocol (e.g. SSL/TLS or IPsec).

Exposure to:

- Goals and techniques of cryptanalysis.
- Programming to avoid security vulnerabilities.

Laboratory Projects

Because the course aims to build cryptographic programming ability, it features regular programming assignments. These might include:

1. Implement and verify a block-cipher or hash function starting with a formal specification.
2. Write a network client to determine if a remote "oracle" is a random or pseudo-random function.
3. Use a multi-precision integer library to implement number-theoretical algorithms.
4. Create and interpret public-key certificates.
5. Establish secure communication using a cryptography library's high-level functionality.
6. Use cryptography library to secure an already-written insecure program.

Estimated Curriculum Category Content (Semester hours)

<i>Area</i>	<i>Core</i>	<i>Advanced</i>	<i>Area</i>	<i>Core</i>	<i>Advanced</i>
Algorithms		1.0	Data Structures		
Software Design		1.0	Prog. Languages		
Comp. Arch.					

Oral and Written Communications

Programming documentation and short essay problems in homework and exams.

Social and Ethical Issues

No significant component.

Theoretical Content

This course addresses the complexity-theoretic assumptions made by modern cryptographers, including the intractability of several problems. Also, reductions play a prominent role in several security arguments made in the class.

Problem Analysis

Analysis is a constant theme in the course. Analyzing security requirements and security properties are the primary activities of cryptography. Written homework frequently stresses such analysis.

Solution Design

Students will design solutions to problems in their programming projects. They will also design, on paper, security protocols.

Status: Approved by Curriculum Committee: 10/24/2007
Approved by Department: 12/12/2007