Institute of Service Science
National Tsing Hua University

Service Security

Spring 2019

<u>Time:</u>
Friday: 9:00am – 12:00pm

<u>Instructor:</u> Soumya Ray
soumya.ray@iss.nthu.edu.tw
406 TSMC Bldg

<u>TA:</u> Boaz Shmueli (on Slack)

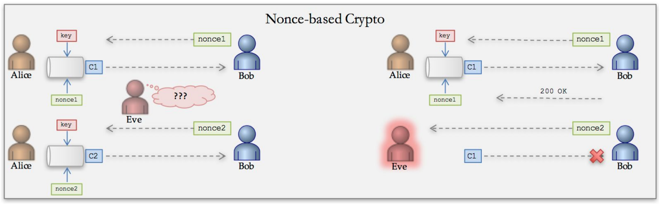
Commercial, financial, governmental, and educational institutions are rushing to deliver new and innovative services over the Internet. Security is a major concern in these initiatives: secure technologies can *protect the privacy* of users and can even be a *competitive advantage* when offered as a core feature.

This course will show students how to *develop highly secure online services* and lead organizations to adopt a *security-conscious culture*. This class focuses on the use of *cryptographic technologies* to create innovative online services for e-commerce, fintech, IoT, and other hi-tech initiatives.

Prerequisites: *This course is intended for students with strong programming experience*.
You should already be comfortable with:

- Coding in an object-oriented language (Java, Javascript, C#, C++, Python, PHP, Ruby, etc.)
- Basic web page design (HTML, CSS)
- Basic database design (relational design, ERD/SQL)
- Unix based systems (Linux, MacOS, etc.)

Objectives: This course will enable students to: design better security practices; develop secure software; interact with security practitioners; engage in security research; and digest the latest technical information about security as it emerges everyday.



Grading:

Grades will be based on individual and team performance. Individual grades come from class participation, contribution to our online discussion, and from individual assignments. Students will also work on a semester project to create an innovative online security system. However, students on a team do not receive the same grade for the team project: each student is evaluated on their individual coding contribution to the project.

Individual Coding: 70%   Class and Online Participation – 10%   Individual Homework – 20%

Topics	**Activities**
Introduction to Information Security	
1. **Service Security**: Services & Security Meet Your Tools	Linux Tutorial 1 Ruby Tutorial 1 CODE: Setup Linux & Ruby
2. **Information Theory** (reschedule) Information Entropy Binary Numbers Binary Operations	Linux Tutorial 2 Ruby Tutorial 2 CODE: XOR Defender
Cryptography: applying cryptographic principles	
3. **Data Encoding and Error** Testing Information Encoding (UTF-8) Error checking	Linux Tutorial 3 Ruby Tutorial 3 CODE: Credit Card Example - Luhn
4. **Cryptography Beginnings** Serializing versus Marshalling Data Simple Symmetric-Key Crypto Modern Perspectives	Git Tutorials 1 CODE: Credit Card Example - Crypto
5. **Symmetric Key Cryptography** Stream & Block Ciphers Brittleness/Complexity of SK Crypto Cryptographic Hashing	Git Tutorials 2 CODE: Credit Card Example - Ciphers READING: Block-chain and Fintech
6. **Public Key Cryptography** Asymmetric Cryptography Complexity Theory Generating Public/Private Keypairs	PROPOSALS: Team Projects
7. (no class)	
Web Services: secure storage	
8. **Web Security** PK Crypto & Web-of-Trust Internet Infrastructure Web Services	
9. **Databases and Web Testing** Environments & Utilities Simple Databases: Sqlite Web Testing	

10. **Database Hardening**

- Threat Model and Matrix
- Database Vulnerabilities
- Securing Database Columns

Users: Accounts and Authentication

11. **User Accounts**

- Protecting Passwords
- Account Infrastructure
- Cloud Deployment

12. **Interface and Authentication**

- API Deployment
- Interface Client
- Cookies, TLS/SSL

13. **Secure Sessions**

- Hardened Cookies and Sessions
- Secure Messaging
- Account Registration

Security Policy: Authorization and Validation

14. **Token-based Authorization**

- JWT controversy
- Discretionary Access Control
- Authorization Tokens

15. **Policies and Validation**

- Distributed Security Policy
- Policy Objects
- Form Validation

16. **Authorization Protocols** (reschedule)

- OAuth Flow
- Distributed OAuth
- Single-Table Inheritance

17. **Client-Side Security**

- Authenticating API Clients
- XSS / CSRF
- Browser Defense: Headers, CSP, Integrity

18. Final Team Presentations