TITLE

"Hands-on Introduction to Internet Security"

or, if you prefer a more academic, less catchy title

"Understanding and Designing Trustworthy Communication Systems for the Internet"

LESSON ABSTRACT

This lesson plan presents a series of 3 1-hour activities that provide a hands-on introduction to internet security for undergraduates with no computer science or programming experience. Through group design challenges mediated by the instructor, students will design their own procedures for ensuring privacy and authentication goals based on public key encryption technology. They will build up conceptual knowledge from basic encryption to digital certificates and apply these ideas to practical, every-day browsing via exercises with Firefox. Two assessment opportunities are also provided to evaluate conceptual and practical understanding.

The lessons were designed for this audience based on principles of constructivist education, situated learning, and group learning. A major innovation is the use of custom Java applets designed to make encryption, decryption, key generation, and key hacking accessible operations for novice security students, so they may focus on security concepts rather than technical details of encryption.

TARGET AUDIENCE

This lesson intends to provide an introduction to internet security systems. This could fit either within a first or second year survey course of computer science topics or as a stand-alone mini-course.

Undergraduate students with basic computing skills are the primary audience. CS majors and non-majors are equally welcome. Absolutely no programming, computer science, or software engineering experience is necessary. Familiarity with some high-school mathematics like prime numbers and algebraic factorization is helpful, but not necessary for all components.

My basic assumptions about students' existing knowledge are as follows:

- Students have abundant experience using the internet for e-commerce, e-mail, and other secured applications
- Students have some familiarity with internet security concerns and risks (why a password is important, etc.)
- Students have some familiarity with the definition of a prime number and the process of factorization

LESSON REQUIREMENTS: TIME, SPACE, AND PEOPLE

This lesson is designed to occur over the course of three (3) instructional periods (or "days") lasting about 60 minutes each. Each day contains multiple activities, including whole class demonstrations, group design challenges, pair case studies, and group presentations to the

class. It may be possible to use one or a few activities in isolation with some fine-tuning. However, the days are designed to flow well together and most of the documentation and advice given in this plan will assume all three periods will be used (so that a later activity can rely on knowledge gained in an early activity).

In terms of space and setting, the lesson should take place in a computer lab or studio space with plenty of room for students to work in pairs at a computer terminal. It is advantageous to have desktop computer stations (one per pair of students) already set up so that the Java applets can be pre-installed and ready to go. However, it could be possible to use student laptops and ask the students to pre-install the applets. Also, having access to a projector or large screen for group viewing is helpful for some demonstrations.

The lesson's primary mode of instruction is a series of group design challenges that progressively build up knowledge of encryption and security processes that enable trustworthy communication between a "bank" and a "customer". The lesson assumes that at least six and at most roughly twenty students are available, as the team-based exercises require substantial interaction with peers and the ability of the instructor to wander between groups and answer student-generated questions effectively. The instructor should be willing to allow students take charge of learning and feel comfortable answering detailed questions about internet security systems. For example, a capable instructor should be able to answer "what is a digital certificate?" and explain and justify all fields in a modern certificate.

INSTRUCTIONAL GOALS

The intention of this lesson is to elucidate the technical and logistical principles that enable *trustworthy* communications over the internet, with a concentration on *identity verification* (who am I exchanging data with?) and *privacy* (can anyone else see my data?). I hope to help students understand the *why* and *how*, with an emphasis on design considerations and practical applications. A major goal is to avoid getting bogged down in the implementation details (how encryption happens in the digital world) and focus on the reasoning behind the system (e.g. what assumptions are made about the keys that can unlock this system?).

A secondary goal of this lesson is to motivate students to take interest in security and computing. It is hoped that the active learning methodologies employed make the concepts accessible and welcoming to a broader audience than computing usually reaches, especially women and under-represented minorities.

LEARNING OBJECTIVES

Cognitive Objectives

1) **Articulate risks of online information transactions, including**
    a. **Privacy risks (e.g. credit card sniffing)**
    b. **Authentication risks (e.g. providing password to false site)**

2) **Explain public key encryption and decryption processes, including**
    a. **Necessary component definitions (public key, private key)**

i. Keys are mathematically related

b. Practical explanation of how reasonable guarantees of security are made

i. Key length as vulnerability parameter

c. External assumptions necessary to assume system integrity

i. Secrecy of the private key

ii. Publicity of public key

3) Discuss a "digital signature" and articulate its utility in practice, including

a. How to create and validate a digital signature via public key systems

b. Explain how this signature can be a "guarantee" of data authenticity

4) Reason about "digital certificate" communications, including

a. How a certificate binds a public key to an entity

i. Why this enables secure online transactions with strangers

b. Assumptions about the Issuer necessary for trust

i. Private key secrecy

ii. Vetting of Subject identity/site ownership

c. Design considerations for the following parameters

i. Key size

ii. Validity period

d. Revocation considerations

i. Explain why revocation is a concern in e-commerce

e. Relevant signatures

i. Which party signs where, and why

Practical Objectives:

1) Identify a site as secured or unsecured while browsing in Firefox

2) Examine a security certificate for a given website using Firefox and identify component parts, including

a. Issuer

b. Issue Public Key

c. Subject

d. Subject Public Key

e. Validity Period

3) Explain Firefox's mechanism for trusting a certificate, such as

a. Stored list of trusted authorities that user can enable/disable

b. Automatic validation of sites providing known root certificates

c. Generation of security exceptions when faced with unknowns

RATIONALE

The overall goals of this lesson are to introduce essential concepts and design principles for secure communication, connect this knowledge to everyday browsing experiences, and (hopefully) help students gain confidence and interest for further experiences in computing. To this end, the primary instructional mode is a series of open-ended group design challenges that allow hands-on work in understanding and assessing communication protocols. In the next few paragraphs I argue that this design challenge approach is well-suited to these three objectives based on evidence from education research and experience teaching this lesson. A more thorough rationale is provided in the Reflection document attached as part of this lesson.

The bulk of this lesson attempts to build an *active* learning environment. This choice is deliberate, as a student can hardly learn how to design and diagnose communication protocols just by watching someone else do it. Importantly, the solutions to the case study and the design challenges are open-ended. There is no one right answer. Instead, a variety of approaches are possible, and students must learn to balance contextual factors (how rigorous should a bank's verification procedure be? my customer will have to do it!) and invent novel approaches. These opportunities for creativity and reasoned justification will solidify conceptual knowledge and help students make the material personally relevant.

In pursuit of the connection to real-world technology and practice, this lesson strives to provide authentic contexts for students to engage with the material. Educational researchers emphasize that students gain knowledge that is applicable to problems outside the classroom when they are asked to adopt not just the tools but the culture and mindset of the practioner[1]. Toward this end, the case study asks students to stand in the shoes of a banking company and make meaningful decisions about how potential account openers must provide documentation of their identity. The design challenge asks students to be online merchants and consumers as well as attackers, so they can approach problems from those perspectives. Finally, the take-home assessment asks students to narrate out-loud why they trust a particular bank's website using the security tools provided by Firefox. Each of these opportunities is reasonably authentic and helps students make connections to why this knowledge is important and how it can be deployed.

The final objective of inspiring curiosity and welcoming diverse students into computing is perhaps the most challenging. To reach this goal, the team-based component of the lesson is perhaps most important. A team setting can provide students ready outlets when they "get stuck" and make the experience fun and enjoyable, especially since the competition between attackers and banks can build morale and team unity. Of course, the team exercises must be well-managed by the instructor to make sure conflicts, gender roles, and other mishaps do not impede learning and inclusion. However, overall the chance to engage with others in authentic exercises should help motivate continued interest in the subject after this lesson ends.

The author personally taught most of the design challenge material to a group of six undergraduate students (four of whom were female). Students remarked afterward that they

[1] A. S. Brown, A. Collins, & P. Duguid (1989), "Situated cognition and the culture of learning," *Educational Researcher* 18(1): 32-41.

"enjoyed" the lesson and were never "bored." Many developed creative, novel solutions to the challenge problems and demonstrated advanced knowledge of the subject (such as seeing encryption as a composable function). Some requested information about additional resources for further learning. This practice experience provides perhaps the most convincing evidence that this approach can be successful in reaching novices and welcoming them into the field of security and systems design.

PROCEDURES

This 3 day lesson is outlined below. Detailed plans can be found in the Instructor Guide.

Day One (60 min): Encryption Basics

1) Motivation Demo and Discussion (5 min)
2) Background Video and Discussion (5 min)
3) Crypto Applet Demo (5 min)
4) Design Challenge A : Encryption and Decryption with Key Exchange (15 min)
5) Design Challenge B : Key Vulnerability (15 min)
6) Design Challenge C : Encryption without face-to-face key exchange (15 min)

First, the instructor gives a motivating demonstration to provide context for the lesson's activities. The script is provided in the Content section below, but the primary goal is to highlight the risks of identity verification for e-commerce. This is done by replaying a popular scam that enticed WellsFargo bank account holders to login at http://www.vvellsfargo.com (note the two v's instead of a w). This provides a good segue into the instructional mission: understand and design process to verify digital identity and privacy. The instructor should make clear that the digital verification problem has much in common with the general verification problem, so the day's activities will include many non-digital situations to increase accessibility.

Second, a brief background video will bring students up to speed on how information travels across the internet. The goal here is to explain roughly how the address www.bankofamerica.com gets send out into the network can comes back with an HTML page. Emphasis should be placed on the many different servers (all owned by different people/corporations) that information travels through. This exposes the risks of eavesdropping or deliberate manipulation of data along the way, as well as the tenuous notion of "ownership" of a URL/domain name. Reiterate the activities focus on understanding and designing ways to authenticate *identity* and *privacy*.

Next, the instructor provides a brief demonstration of the Java applets that will be used to provide accessible, experimentable encryption and decryption services. This demonstration should explain the concepts **public key** and **private key** and highlight how these are used to encrypt and decrypt a real message. Hinting that these keys are mathematically related is a good idea. Also, demonstrating to students that small errors in the message or key produce very different encryption/decryption results is important.

Finally, students will embark on a series of open-ended design challenges in which Bank teams

need to pass a URL link to Customer teams in a secure fashion to facilitate logging in to a new account. The teams are given access to the Java applets and allowed to plan out their process for a brief time before executing it. Once the basics are mastered, new challenges are thrown in like an intruder that can hack the public key given enough time and the inability for face-to-face communication.

Day Two (75 min): Digital Certificate Basics

1) Group Discussion : Recap Day One (5 min)
2) Case Study: Assumptions and risks in identity documentation (10 min)
3) Practical Browsing: What does a digital certificate look like in firefox? (10 min)
4) Design Challenge D : Requesting and Using Digital Certificate (20 min)
5) Design Challenge E : Intruder-proof Certificate-based Communication (20 min)

After recapping the major lessons of Day One, the instructor can introduce students to the idea of digital certificates first through an analogy to driver's licenses and other identity documentation. A brief case student allows students in pairs to reason about why some documentation is trustworthy and the assumptions underlying this reasoning. In this case study, they will pretend to manage new account openings for an online investment bank. The case study will present a series of "applications" for new accounts, and the students' role is to determine whether to validate each application based on whether the opener's identity credentials appear sufficient. In closing, the instructor should emphasize where the analogy breaks down to avoid misconceptions[2].

Next, a practical exercise has the pairs open up Firefox and examine a real digital certificate. The goal here is to familiarize students with what goes on "under the hood" and connect their concepts to everyday e-commerce actions. Students fill out a short list of questions that encourages them to apply the reasoning in the case study above to the digital certificate domain. For example, "what assumptions do I make about the Issuer if I trust this certificate?"

Finally, a series of design challenges asks students to first invent a scheme for communicating via digital certificates and then revise this scheme to make it intruder-proof within certain time limits. If possible, several student teams should be placed alternately in the intruder role to allow many different, creative attacks. Again, the Bank and Customer context applies to these challenges.

Day Three (75 min): Certificate Issuer Policies and Risks

1) Group Discussion : Recap Day Two (5 min)
2) Design Challenge F: Policy for a Trustworthy Certificate Authority (30 min)
3) Group Presentations : Justifying Issuer Policy Decisions (5 min / team of 6)

After recapping the definition and purpose of a digital certificate, the group immediately jumps into the capstone exercise of this lesson series: a design challenge to invent and justify a policy

[2] Glynn, S.M. (1995) Using analogies to explain scientific concepts. *The Science Teacher*. 62 (9).

for a certificate issuer. This exercise will utilize all knowledge build up to this point (e.g. how big should keys be connects to Day One, what does a certificate look like connects to Day Two) and also extends knowledge in important, practical ways (e.g. thinking about revocation). The result of this design exercise is a presentation to the class about the particular scheme invented by the group and justification of strengths and weaknesses. These presentations are formally assessed by the instructor to determine group-wide comprehension.

Take-home Assessment

1) Practical Browsing: Why can I trust this banking website? (30 min estimated)

In a take-home reinforcement exercise, students are asked to record an audio/video screencast of a visit to a secured website of their choice. They should narrate the process of dissecting the site's certificate and explaining why this document makes the site trustworthy at a detailed level including concepts like key size, revocation, and validity period. This screencast video is then assessed by the instructor to determine individual comprehension of material.

LESSON MATERIALS

Four attached documents provide all written material necessary for enacting this lesson.

- The InstructorGuide document provides a field guide for giving this lesson to students. Included are a detailed summary of all activities day-by-day and hints and tips as well as recap boxes to help the instructor summarize knowledge at the end of each activity.

- The StudentPacket document provides a ready-to-print packet to distribute to each student for the entire lesson. This includes instructions and workspace for all major activities of the lesson series.

- The Assessment document provides the instructions and rubrics for each assessment exercise.

- The ReflectiveEssay document provides a 3000 word essay written by the author explaining the detailed justification for this lesson.

Software is provided as the file EasyRSA.zip, which includes the following

- EasyRSA.jar - a java archive executable that launches all four applets
  - Encryption
    - Encrypts plain text ASCII message with given private key
    - Supports multi-line messages
    - Produces a very long decimal number
  - Decryption
    - Decrypts cipher text (decimal number) given public key
    - Supports multi-line messages
  - Key Generator
    - Generates key of given number of decimal digits (min=6,max=16)
  - Key Hacker

    - Factorizes given public key using pre-generated lists of primes
    - Provides real-time clock to observe how long factorization takes
- primes*.txt - series of text files containing prime numbers (used for KeyHacker)

Instructions for installing and operating the Java applets are in the appendices of this document.

ASSESSMENT

1) *Day Three Design Challenge: Policies for a Trustworthy Certificate Authority*

Form teams of four to six students each. Each team is given the prompt and a series of questions they must answer. They are encouraged to talk to other teams and draw/sketch out flowcharts for several processes. They are given 20 minutes to come up with creative, consistent answers to all questions. After 20 minutes, teams must stop their ideation and focus on presentation planning. After 10 minutes of preparation, groups must give a presentation to the group lasting no more than 10 minutes. They are encouraged to identity one primary advantage and one disadvantage of their scheme as part of the goal of the presentation.

As feedback, the instructor will provide his/her marked up copy of the evaluation sheet to all team members. The instructor will also give brief oral comments at the end of each presentation and encourage classmates to ask questions as well.

2) *Individual Take-Home Screencast: Trusting Certificates with Modern Browsing Technology*

Each student is given a prompt and a series of questions. They must open up a browsing session, navigate to a provided URL, and then think through each question out loud while manipulating the certificate or authority policies of the browser. They are encouraged to rehearse their process a few times before recording a five minute screencast (a video of their on-screen actions with audio narrative). The narrative should both answer the questions and outline things they still wonder about.

As feedback, the instructor will provide a copy of the marked up evaluation sheet to all team members. Additionally, instructor should make herself available for drop-in office hours to answer questions about the assessment after completion.

APPENDICES

A) Installing and running the Java applets
    a. Pre-requisites
        i. Since Java is architecturally independent, these applets can run on any desktop OS (Windows, Mac, or Linux)
        ii. However, the necessary processing power probably requires a computer with at least 500 MHz (1 GHz or more is better)
    b. Pre-installation

i. The applets require a JRE (java runtime environment) to be installed on the system.
ii. This can be installed following the instructions at http://www.java.com/en/download/help/download_options.xml

c. Installation
   i. Unzip the EasyRSA folder into any directory on your computer
   ii. Make sure the *.jar file as well as the prime*.txt files all end up in the same directory
d. Run
   i. Double-click on the *.jar file
   ii. If this fails,
      1. Open a command window and cd into the installation directory
      2. Type "java –jar *.jar"