Lecture 1: Introduction, and Private Key Crypto -- I

CS 436/636/736
Spring 2015

Nitesh Saxena

Today’s informative/fun bit -- CAPTCHA
Outline

- Administrative Stuff
- Introductory Technical Stuff

Some Important Pointers

- Instructor: Nitesh Saxena
  - Office: CH 133
  - Email: saxena@cis.uab.edu (best way to reach me!)
  - Phone No: 205-975-3432
  - Office Hours: Wednesdays 3:30-4:30pm (or by appointment)
- Course Web Page (also accessible through my web-page) http://www.cis.uab.edu/saxena/teaching/cs36-s15/
- TA/Grader
  - Dibya Mukhopadhyay
    - PhD student – research in Security
    - Office hrs: 2-3pm Tuesdays, and 2-3pm Thursdays - Ugrad lab (CH 154)
- Canvas: http://www.uab.edu/online/canvas
About the Instructor

• Associate Professor, CIS
• Research in computer and network security, and applied cryptography
• Web site: http://cis.uab.edu/saxena
• Group web site: http://spies.cis.uab.edu

Prerequisites

• The official prerequisites for the undergraduate (436) students are:
  – Algorithms and Data Structures (CS 303 or equivalent)
  – Computer Organization and Assembly Language Programming (CS 330 or equivalent)
• A minimum grade of C is required in the prerequisite courses
• At a higher level, the course requires the students to have good mathematical background, programming knowledge, and familiarity with algorithms and data structures
What to expect

• The course would be quite involved
  – Lot of math
  – Some programming
  – Reasonable workload
• The grading will be curved
  – I would love to give A’s but I won’t mind giving P’s when deserved 😊
• I might/will make mistakes
  – Please point them out
  – Talk to me if you have any complaints (or send me an anonymous email 😊)
• I guarantee that
  – I will encourage you to do your best
  – You’ll have fun
  – I’ll help you learn as much as I can – don’t hesitate to ask for help whenever needed
  – Although you won’t become experts, you will learn enough to move on!
  – You’ll hopefully get motivated to pursue research in this area, ultimately
• This class is not
  – Going to be be an easy substitute for something (believe me it will not be!)
  – About “hacking”

1/8/2015 Lecture 1 - Introduction

What I expect of you

• Please do attend lectures
• Review lecture slides after each lecture
• Solve text book exercises as you read through the chapters
• Ask questions in the class
• Ask questions over email
• Attend office hours
• Try to start early on homework assignments
  – Don’t wait until the very last minute!
• Follow the instructions and submit assignments on time

1/8/2015 Lecture 1 - Introduction
Course References

- No mandatory textbook
  - Lecture slides are your primary reference

- Recommended text:

- Wikipedia can be a good reference at times (but need to be careful in using/trusting the information provided)
- Other references to be provided as we proceed

Grading

- 50% - Homework Assignments (some may involve programming)
- 25% - 1 Midterm Exam
- 25% - 1 Final Exam
Policies Against Cheating or Misconduct

• You are not allowed to collaborate with any other student, in any form, while doing your homeworks, *unless stated otherwise*; perpetrators will at least fail the course or disciplinary action may be taken.
• No collaboration of any form is allowed on exams.
• You can definitely refer to online materials and other textbooks; but whenever you do, you should cite so in your homeworks. This is a rule of thumb.
• Also check: [https://www.uab.edu/students/academics/honor-code](https://www.uab.edu/students/academics/honor-code)

Late Homework Policy

• None – *no late homeworks are allowed*
• Either you submit on time and your homework will be graded OR you submit late and the homework is NOT graded.
• You should stick to deadlines.
• Exception will be made ONLY under genuine circumstances.
Tentative Course Schedule

- Cryptography: first several lectures
- Some Network Security: Protocol Design
- Privacy and Anonymity
- Security Design Principles
- Threat Modeling
- Software Vulnerabilities (Buffer Overflows)
- Usable Security

Scheduled Travel

- Usually conference and invited talks travel
- Usually no class the week of travel
- However, this will not affect our overall course schedule and topic coverage (perhaps a guest lecturer will cover on my behalf)

- Information about any travel will be provided as it becomes available
Instructions

• HW submissions
  – Name your files “Lastname_Firstname_HW#”
  – Submit it on Canvas
    • Please make sure that you have correctly submitted/uploaded the files
      (simply “saving” them may not be sufficient)
  – PDF format only
• Check the course web-site regularly
  – I am posting lecture slides and homeworks there
• Check your UAB email regularly
  – I am sending out announcements there
    • e.g., when I post homeworks
• Only use your UAB email to communicate with me and the TA/grader
• NO EXCUSES for not following instructions
Computer Security: Why it is important?

• The numbers speak for themselves.
  – CERT Statistics
• Our computer systems are quite vulnerable
  – Poor design or after the fact design
  – Lack of awareness and education
  – Weak threat model and under-estimation of attacker capabilities
  – Buggy software

Threats, Vulnerabilities and Attacks

• A threat to a system is any potential occurrence, malicious or otherwise, that can have an adverse effect on the assets and resources associated with the system
• A vulnerability of a system is some characteristic that makes it possible for a threat to occur
• An attack on a system is some action that involves exploitation of some vulnerability in order to cause an existing threat to occur
Types of Threats

• Can be classified into four broad categories
  – Disclosure - unauthorized access to information
  – Deception - acceptance of false data
  – Disruption - interruption or prevention of correct operation
  – Usurpation - unauthorized control of some part of a system

• Examples include – snooping, sniffing, spoofing, delaying, denial of service, malware, theft of computational resources...

Primary Issues

• Confidentiality: prevention of unauthorized disclosure of information

• Integrity: prevention of unauthorized modification of information

• Availability: ability to withstand unauthorized withholding of information or resources
Computer Security – other issues

- There are other issues that arise in the design of secure systems besides confidentiality, availability and integrity:
  - Accountability
  - Reliability
  - Access Control
  - Authentication
  - Non-repudiation
  - Privacy and anonymity

Policy and Mechanism

- A security policy is a statement of what is, and is not, allowed
  - Expressed mathematically
  - List of allowed and disallowed actions
- A security mechanism is a procedure, tool, or method of enforcing security policy
Elements of a Security Policy

- A security policy considers all relevant aspects of confidentiality, integrity and availability
  - Confidentiality policy: Identifies information leakage and controls information flow
  - Integrity Policy: Identifies authorized ways in which information may be altered. Enforces separation of duties
  - Availability policy: Describes what services must be provided: example – a browser may download pages but no Java applets

UAB Data Protection and Security Policy

Security Mechanism

• A security mechanism is a procedure that enforces some part of a security policy
• We will learn many cryptographic and non-cryptographic mechanisms

Goals of Security Mechanism

• Given a policy that specifies what is “secure” and what is “non-secure” goal of security is to put in place mechanisms that provide:
  — Prevention
    • Involves implementing mechanisms that users cannot override and are trusted to be implemented in correct and unalterable ways
  — Detection
    • Goal is to determine that an attack is underway, or has occurred and report it
  — Recovery
    • Resuming correct operation either after an attack or even while an attack is underway
Types of Security Mechanisms/controls

• Cryptography and cryptographic protocols
• Software controls
• Hardware controls
• Physical controls

Trust

• Security policies and mechanisms are based on assumptions and one trusts that these assumptions hold.
• Aspirin from drugstore is considered trustworthy. The basis of this trust is:
  – Testing and certification by FDA.
  – Manufacturing standard of company and regulatory mechanisms that ensure it.
  – Safety seal on the bottle.
• Similarly, for a secure system to achieve trust, specific steps need to be taken.
Trust

- Trusting the mechanism requires us to assume:
  - Each mechanism designed to implement part of policy
  - Union of mechanisms implement all aspects of policy
  - Implemented correctly
  - Installed and administered correctly

Operational Issues in Security

- Risk Analysis or Assessment
- Cost-Benefit Analysis
- Laws and Regulations
- Human Issues: usability
Security Life Cycle

- Threats
- Policy
- Specification
- Design
- Implementation
- Operation and Maintenance

Some Questions
- I access internet via your wireless access point: this is an example of what?
- Alice knows that Bob bought a herpes drug today: what’s violated?
- I use an open smtp server to send an email using your email address: I what-ed you?
- My photos are residing on facebook’s server: what do I trust?
- Microsoft keeps issuing security patches for their softwares: what stages in security life cycles are involved?
Some Questions

• University policy disallows cheating – copying another student’s homework assignment. Student A has her homework file world-readable. Student B copies it. Has B violated the policy?
• You do not make your age public on Facebook. An attacker can however infer your age from the ages of your friends. What has been violated?
• Eve jams the wireless signal in the CS: what does she achieve?
• INS officials identify immigrants using their fingerprints: what is the primary assumption in play?
• US law allows sharing copyright movies using for example BitTorrent: true or false?
• Alice is dead. Could Alice’s mother get access to her late daughter’s emails residing on Yahoo’s server? See: http://www.cnn.com/2009/TECH/05/18/death.online/index.html

Further Reading

• Must read “What is There to Worry About? An Introduction to the Computer Security Problem” by Brinkley and Schell
• Optional read “Concepts and Terminology for Computer Security” by Brinkley and Schell
Private Key Cryptography -- I

Outline

- Cryptography Overview
- Private Key Cryptography: Encryption
- Classical Ciphers
Cryptography

- Etymology: Secret (Crypt) Writing (Graphy)
- Study of mathematical techniques to achieve various goals in information security, such as confidentiality, authentication, integrity, non-repudiation, etc.
- Not the only means of providing information security, rather a subset of techniques.
- Quite an old field!

Cryptography: Cast of Characters

- Alice (A) and Bob (B): communicating parties
- Eve (E): Eavesdropping (or passive) adversary
- Mallory (M): Man-in-the-Middle (or active adversary)
- Trent (T): a trusted third party (TTP)
Today’s Focus

• How to achieve confidentiality by means of cryptography?

Private Key/Public Key Cryptography

• **Private Key**: Sender and receiver share a common (private) key
  – Encryption and Decryption is done using the private key
  – Also called conventional/shared-key/single-key/symmetric-key cryptography

• **Public Key**: Every user has a private key and a public key
  – Encryption is done using the public key and Decryption using private key
  – Also called two-key/asymmetric-key cryptography
Common Terminologies

- Plaintext
- Key
- Encrypt (encipher)
- Ciphertext
- Decrypt (decipher)
- Cipher
- Cryptosystem
- Cryptanalysis (codebreaking)
- Cryptology: Cryptography + Cryptanalysis

Private key model
Open vs Closed Design

- Closed Design (as was followed in military communication during the World Wars)
  - Keep the cipher secret
  - Also sometimes referred to as the “proprietary design”
  - Bad practice! (why?)

- Open Design (*Kerckhoffs’ principle*)
  - Keep everything public, except the key
  - Good practice – this is what we focus upon!

Private Key Encryption: main functions

1. KeyGen: $K = \text{KeyGen}(l)$ ($l$ is a security parameter)

2. Enc: $C = \text{Enc}(K, M)$

3. Dec: $M = \text{Dec}(K, C)$
Goals of the Attacker

• Learn the plaintext corresponding to a given ciphertext -- **One-Way Security**
• Extract the key – **Key Recovery Security**
• Learn some information about the plaintext corresponding to a given ciphertext – **Semantic Security**
• *Key recovery security and one-way security are a must for an encryption scheme. Semantic Security is ideal.*

Capabilities of the Attacker

1. **No Information** (besides the algorithm)
2. **Ciphertext only**
   – Adversary knows only the ciphertext(s)
3. **Known plaintext**
   – Adversary knows a set of plaintext-ciphertext pairs
4. **Chosen (and adaptively chosen) plaintext** (CPA attack)
   – Adversary chooses a number of plaintexts and obtains the corresponding ciphertexts
5. **Chosen (and adaptively chosen) ciphertext attack** (CCA attack)
   – Adversary chooses a number of ciphertexts and obtains the corresponding plaintexts
Security Model

<table>
<thead>
<tr>
<th>least attacker capability</th>
<th>most attacker capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>weakest cryptosystem</td>
<td>strongest cryptosystem</td>
</tr>
</tbody>
</table>

- 1 is the hardest and 5 is the easiest attack to perform
- A cryptosystem secure against 5 is the strongest, and secure against 1 is the weakest
- A cryptosystem secure against 5 is automatically secure against 4, 3, 2 and 1

Brute Force Attacks: Key Recovery

- Since the key space is finite, given a pair (or more) of plaintext and ciphertext, a cryptanalyst can try and check all possible keys.
- For above to be not feasible, key space should be large!!
  - How large?
  - Large enough to make it impractical for an adversary. But what is impractical today, may not be so tomorrow. At least $2^{80}$ – see this paper on “selecting cryptographic key sizes”
Ciphers We Will Study

- Classical ones
  - Substitution Ciphers
    - Caesar’s Cipher
    - Monoalphabetic
    - Polyalphabetic
  - Transposition Ciphers

- Modern ones
  - DES/AES
  - Others...

Caesar Cipher (or Shift Cipher)

- Substitution cipher
- Let messages be all lower case from a through z (no spaces or punctuation).
- Represent letters by numbers from 0 to 25.
- Encryption function
  \[ C_i = E(P_i) = P_i + K \pmod{26} \]
  where \( K \) is secret key
- Decryption is
  \[ P_i = D(C_i) = C_i - K \pmod{26} \]
Security of Caesar Cipher

- Easy to brute force: size of key-space is 26
  - Not secure against even ciphertext-only attack (the one where adversary had the least capability)

Monoalphabetic Substitution

<table>
<thead>
<tr>
<th>P</th>
<th>O</th>
<th>K</th>
<th>E</th>
<th>M</th>
<th>O</th>
<th>N</th>
<th>M</th>
<th>A</th>
<th>S</th>
<th>T</th>
<th>E</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td>K</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>N</td>
<td>O</td>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
</tr>
<tr>
<td>P</td>
<td>O</td>
<td>L</td>
<td>Y</td>
<td>T</td>
<td>E</td>
<td>C</td>
<td>H</td>
<td>N</td>
<td>I</td>
<td>U</td>
<td>V</td>
<td>R</td>
</tr>
<tr>
<td>S</td>
<td>B</td>
<td>K</td>
<td>W</td>
<td>A</td>
<td>D</td>
<td>F</td>
<td>G</td>
<td>J</td>
<td>M</td>
<td>Q</td>
<td>X</td>
<td>Z</td>
</tr>
<tr>
<td>K</td>
<td>B</td>
<td>U</td>
<td>T</td>
<td>R</td>
<td>B</td>
<td>S</td>
<td>R</td>
<td>P</td>
<td>D</td>
<td>F</td>
<td>T</td>
<td>A</td>
</tr>
</tbody>
</table>
Monoalphabetic Substitution

- Key space is large $26! = 4 \times 10^{26}$
  - Quite large, however,
  - Can be broken (not secure against ciphertext-only) using language characteristics!

Polyalphabetic Substitution – Vigenere Cipher

- Use K mono-alphabetic ciphers – $E_1$, $E_2$, …, $E_k$.
- In position $i$, of plaintext, use cipher $E_i$.
- Example using Caesar ciphers …

| Plaintext:  helloiloveyouwontyoutellmeyourname |
| Key:       polytechnicpolytechnicpolytechnicpoly |
| Ciphertext: wswijhmnv................................. |

- A little harder to break but frequency analysis is possible
- Some well known techniques for determining key length – we will not cover (see text for Kasiski method)
One time Pad or Vernam Cipher: Best Possible Cipher

- If we use Vigenere with key length as long as plaintext, then cryptanalysis will be difficult!
- If we change key every time we encrypt then cryptanalyst’s job becomes even more difficult. One-time pad or Vernam Cipher.
- How do we get such long keys?
- Such a cipher is difficult to break but not very practical.

Binary Vernam

- plaintext is binary string and key is binary string of equal length, then encryption can be done by a simple XOR operation.
  - Plaintext: 01010000010001010011
  - Key: 11010101001001100111
  - Ciphertext: 10000101011000110100
- If the key is random and is not re-used, then such a system offers unconditional security – perfect secrecy!
- Intuitively perfect secrecy can be seen from the fact that given any plaintext and ciphertext, there is a key which maps the selected plaintext to the selected ciphertext. So given a ciphertext, we get no information whatsoever on what key or plaintext could have been used.
- How do we obtain “random” bit-strings for shared secret keys as long as the messages, and never re-use them?
- Again system is not practical.
Transposition

- Harder to break than substitution ciphers
- Still susceptible to frequency analysis

Product Ciphers

- Substitution and transposition ciphers are not secure due to language characteristics
- What about using two or more of these ciphers in a serial fashion
  - Two or more substitutions
  - Two or more Transpositions
  - A few substitutions and a few transposition
    ➔ Transition from classical to modern ciphers
Some Questions

- Enigma is an example of what type of design?
- Encryption can provide confidentiality, but not integrity: true or false?
- World's best cipher is what?
- I give you a ciphertext, and ask you to give me the corresponding plaintext. What attack is this? How does it compare to the known plaintext attack?
- All classical ciphers are based on either what type of design?
- What's the problem in choosing a long key? It should give you a lot of security, no?

Some Questions

- An encryption scheme is said to be deterministic if encrypting the same plaintext twice yields the same ciphertext. (otherwise it is said to be randomized).
  - Is a deterministic scheme a good scheme in terms of security?
Further Reading

- Stallings (edition 5) – Chapter 2.1 to 2.3
- HAC – Chapter 1 and 7