CSE 3400: Introduction to Computer & Network Security

(or CSE 5850: Introduction to Cybersecurity)

Lecture 1
Introduction

Ghada Almashaqbeh

UConn - Spring 2022

Outline

- Course logistics and syllabus overview.
- Brief history.
- Cryptography and cybersecurity.
- Background.

History I

- Cryptology - “science of secrets”.
    - Ancient field, even before computers were invented.
    - Was merely about confidential communication.
        - Mainly about encryption; convert plaintext to ciphertext such that only the intended recipient can correctly decrypt.
    - Cryptography - “secret writing” - is a more popular term now.
- Kerckhoffs' Principle.
    - Avoid security by obscurity.
    - Instead, cryptographic/security algorithms, schemes, or mechanisms should be public.

History II

- Modern Cryptography.
  - Moved from ad hoc ancient solutions and military secret tools to science/scholarly research/industrial products/etc.
    - Public algorithms.
    - Well defined security notions.
    - Formal security proofs and/or extensive cryptanalysis.

Cryptography is only about secrecy?

- No!! It can achieve a large variety of goals, to name a few:
  - Confidentiality (or secrecy) - encryption.
  - Integrity and authenticity - message authentication codes and digital signatures.
  - Nonrepudiation - digital signatures.
  - Secret key establishment, sharing, and management.
  - Secure function evaluation over private input (two or multi party setup).
  - Computation over encrypted data.
  - etc.

Cybersecurity

- Securing the cyberspace.
  - The cyberspace is the collection of interconnected computers, devices, machines, etc., and the information flow between them.
  - More technological advances ⇒ more critical data can be exchanged ⇒ attackers are more motivated to attack the cyberspace.
  - Resulted in multiple fields, such as:
    - Computer security.
    - Software security.
    - Network security.
    - Information security.

Background - Computational Complexity I

- Mainly we deal efficient or computationally bounded adversaries.
  - The class of PPT (probabilistic polynomial time) algorithms.
  - An algorithm A is in PPT if it takes a polynomial number of steps (in the input size) to terminate.
- A scheme that is secure against PPT adversaries is ***computationally secure***.
  - A scheme is secure if a PPT attacker succeeds with negligible probability.
  - This rules out exhaustive search attacks.
    - Infeasible in practice.

Background - Computational Complexity II

- A scheme secure against unbounded attackers is ***information theoretically (or unconditionally) secure***.
  - Even if the attacker has unbounded resources (storage, time, etc.), it cannot break the security of the scheme.
- Security parameter.
  - The main factor impacting the run time of cryptographic algorithms.
  - Usually related to the key length.
  - Passed as input to algorithms in unary representation.
    - E.g., a security parameter value is integer l we pass it as 1^l.

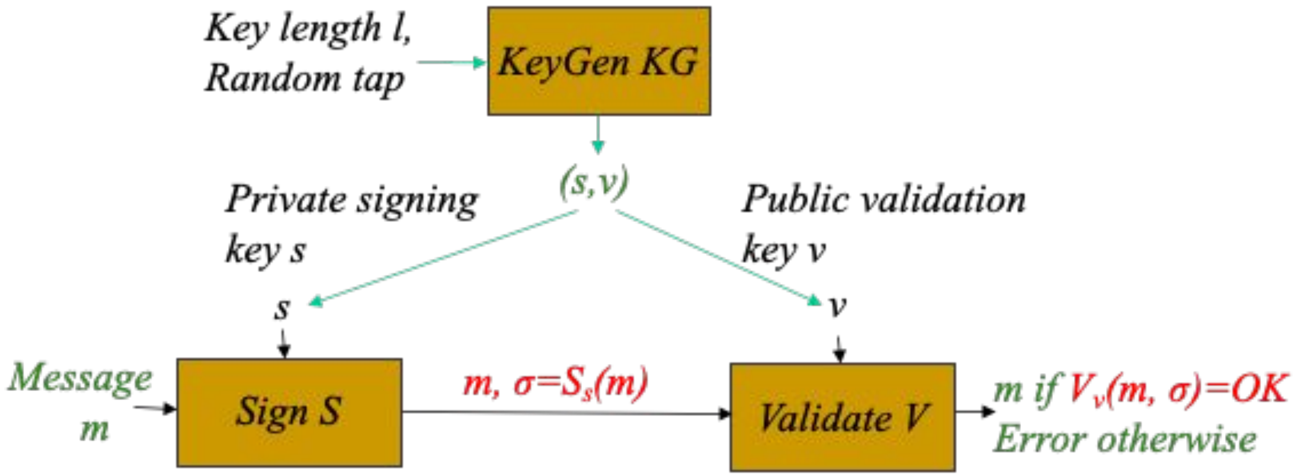
Security Goals and Definitions I

- Three principles of modern cryptography:
  - Correctness and security definitions (or notions).
    - Define how the scheme should act when used as defined (benign scenario)
    - Define the security goals/requirements/properties that when met the scheme will be secure.
      - This also prevents incorrect use of the scheme.
  - Precise assumptions.
    - Precise definition of attacker capability (but not strategies) we account for.
    - Usually this involves hardness assumpitons on which we rely to establish the security of the developed scheme.

Security Goals and Definitions II

- Three principles of modern cryptography (contd.):
    - Formal security proofs.
        - Show how the scheme satisfies the security notion under the used assumptions.
        - For involved systems/protocols, it could be hard to have completely rigorous models and proofs.

An Example - Digital Signatures



- Assumed limitations:
  - Knowledge limitations: key s is secret (unknown to attacker)
  - Resource limitations: can’t find key s by trying all keys
- Correctness: any signature produced using S will verify correctly (V will always output OK)
- Security: An attacker cannot forge signatures
  - I.e., find `signature’ σ for a new message m s.t. $V_v(m, \sigma)$=OK

* Slide adopted from textbook slides.

Concrete and Asymptotic Security

- Concrete security:
  - Measure security in terms of the adversary advantage function value.
  - So it computes a concrete probability value for specific (concrete) parameter values such as key length, number of queries an adversary can perform, etc.
- Asymptotic security:
  - It requires the advantage function to be negligible in the security parameter.
    - I.e., it converges to zero for large enough input values.
    - E.g., a polynomial p(n) is non-negligible while an inverse exponential 2^{-n} is negligible in n.
  - We use NEGL to denote the set of all negligible functions.

Notes - Chapter 1

- Self study.
  - Section 1.3.3 and Appendix A.3 to refresh your knowledge of basic probability.
- For later.
  - Sections 1.3.2 and A.2 (Background on Modular Arithmetic) will be covered when we study public key cryptography.
  - Sections 1.3.6 and 1.4 (defining security of digital signatures).
- Table 1.1 under section 1.3.5 includes most of the notations used in the textbook.
  - We will revisit them over and over again while studying the course material.

Notes - Textbook

- You may find some of the concepts mentioned in Chapter 1 hard to comprehend (like section 1.2). These will become much clearer as we progress in the course material.
    - ***So do not get discouraged!!***
- As you may have noticed, the textbook is still a draft version.
    - Prof. Herzberg is still reviewing/editing the textbook.
        - Make sure to fetch the latest version of each chapter as we move forward in the semester.
        - If you find any mistakes or typos, or you have any suggestions or comments, please share them with me to pass them along.

Covered Material From the Textbook

- Chapter 1: all except Sections 1.3.2, 1.3.6 and 1.4.
- Appendix A.3