Applied Cryptography and Computer Security
CSE 664 Spring 2020

Lecture 1: Basic Definitions and Concepts

Department of Computer Science and Engineering
University at Buffalo

Academic Integrity

- Your first assignment is to to read the CSE and UB academic integrity policies
- We are not talking about this for the sake of talking
  - examples of violating rules in industry include
    - a Microsoft employee shares code with another Microsoft employee and the company gets a fine for a half a billion
    - a new employee who takes code from the previous employer faces 15 years in prison
    - certain software sectors require documenting code development and "I took it from stack exhange" is not acceptable documentation

Academic Integrity

- Here are examples for your consideration
  - you work on your laptop at a library with friends and step away from your computer without locking it
  - you look at your neighbors' papers during an exam, but don't copy their answers
  - you take a piece of code from some website and give a link to the website at the end of the homework
  - you work on a homework problem with friends, type the solution at home, but it's the same as that of your friends
  - your program doesn't run, but you want to get credit and complete the assignment by reporting program execution timings

What Background is Expected?

- Mathematical maturity, including:
  - basic complexity theory
    - ability to evaluate complexity of algorithms using big-O notation
  - elementary discrete math
    - ability to work with sets, modular arithmetics
  - elementary probability theory
    - ability to compute probability of conjunction or disjunction of independent events, conditional probability
  - familiarity with mathematical proofs
    - proofs by construction, contradiction
- Programming abilities

What is Cryptography?

- Historically, the use of cryptography was to ensure secrecy of transmitting messages
- Primarily uses were by military and was perceived as an art of designing codes
- Today it evolved into a rigorous study of mathematical techniques
- Its uses significantly exceed secret communication alone

Where Do We Find Cryptography Today?

- Widely used applications of cryptography include:
  - secure communication on the web
    - secure credit card purchases, online banking, etc.
  - secure remote login and authentication
  - digital signatures and certificates
  - access control enforcement in multi-user operating systems
  - disk encryption
  - software protection
  - system, transaction, or communication integrity checking
  - trusted computing and data modification
  - secure electronic voting and elections

More Esoteric Uses of Cryptography

- Cryptography also allows us to realize:
  - secure bidding and auctions
  - e-cash
  - contract negotiation and fair contract signing
  - anonymous authentication (e.g., using hidden credentials and/or hidden policies)
  - usage of untrusted storage (e.g., searches on encrypted data) or untrusted computational power (e.g., uncheatable grid computing)
  - privacy-preserving computation and outsourcing
  - many other capabilities

What is Modern Cryptography?

- Cryptography is the scientific study of techniques for achieving security objectives
  - securing digital information, transactions, distributed communications
  - any distributed computation or interaction that may come under attack
- Cryptanalysis is the study of mathematical techniques for attempting to defeat security objectives
- Modern cryptography is formal and rigorous

Why is Rigorous Treatment Important?

- Too many proposals fail to achieve their security objectives
  - if any of them is deployed on a wide scale, consequences can be disastrous
- In modern cryptography, we
  - clearly state all assumptions
  - define the power an adversary has
  - show security of the system in the presence of such adversary under the stated assumptions
- Such design is likely to withstand the time if the underlying assumptions prove to hold

But Good Design is Not Everything

- Good design is only half of the game
  - correct implementation is no less important
  - history shows numerous examples of spectacular security failures due to improper implementation or configuration
- Common causes of implementation failure
  - improper choice of parameters
  - improperly chosen randomness
- Clear understanding of security guarantees of a cryptographic solution is important for correct use

What Security Objectives Can We Have?

- Examples of security objectives:
  - confidentiality: information is available to authorized parties only
  - integrity: any unauthorized change to the data is detected
  - availability: resources are available to authorized parties
- Cryptography is only one tool for realizing security objectives
  - others include software, hardware, physical security, etc.
- Many other security objectives can be formulated

Attacker Models

- We often refer to participants in a cryptosystem as Alice and Bob
- An adversary Eve/Carl/Mallory eavesdrops on the communication or tries to disrupt the protocol
  - passive attacker
  - active attacker
  - outsider
  - insider

Attacker's Power

- A cryptographic system often
  - precisely defines the power of an attacker
  - formally shows resilience to such adversarial behavior
- How powerful should we expect the adversary to be?
  - option 1: can assume adversary has unlimited resources
  - option 2: can assume adversary is limited by our computational abilities

What Does it Mean for a Cryptosystem to be Secure?

- Unconditional or information-theoretic security
  - the system is secure even in presence of adversary with unlimited computational resources
  - security analysis uses probability theory
  - for example, perfect secrecy in encryption schemes
- Computational security
  - relies on a hard computational problem that cannot be solved on a today's computer
  - can be broken in principle using enough computing resources
  - system stays secure as long as the underlying hard problem is believed to remain hard

Modern Cryptographic Design

- Kerckhoffs' principle
  - it states that algorithms comprising a cryptosystem should not be kept secret
  - why?

- Unfortunately, security by obscurity is still very common
  - always use a standardized construction with public design

Modern Cryptographic Design

- Principles of modern cryptography
  - formulation of rigorous and precise definition of security
    - important for design
    - important for usage
    - important for studying
  - unproven assumptions must be clearly stated
    - security cannot be proven otherwise
    - can be used for comparison of schemes (weaker assumptions are preferred)
    - facilitates studying of the assumptions

Modern Cryptographic Design

- Principles of modern cryptography (cont.)
  - proofs of security with respect to the definition and relative to the assumption
    - without proofs, security is left to intuition and is often broken shortly after
    - reductions are most common types of security proofs

      "given that assumption A holds, construction B is secure according to the given definition"
    - reduction means that breaking security of B is at least as hard as breaking A
    - proof by reduction proceeds by showing that if B is insecure, A does not hold

Hardness Terms

- In cryptography these terms are used as:
  - given a security parameter k, easy (efficient) means it is possible to compute a function in time polynomial in k
  - hard (infeasible) means that computation cannot be performed in polynomial time (e.g., requires exponential computation)
  - impossible means that the function cannot be computed using unlimited resources
  - negligible means that the function drops faster than any polynomial (i.e., at a super-polynomial rate)