CSC 429/529 – Cryptography

Instructor Bruce Kapron

Office ECS 620

Phone 472-5725

Office Hours TBA

Cryptography – Introduction

Basic goals of this course:

- Present basic cryptographic tools, e.g., private- and public-key encryption, cryptographic hash functions, zero-knowledge protocols, pseudo-random generators, etc.

- Look at applications of these tools in various security tasks, e.g., authentication, key exchange, identification, secret sharing, etc.

Along the way, we will examine topics from information theory, algebra, probability and complexity, as needed

Resources

J. Katz & Y. Lindell, *Introduction to Modern Cryptography*

Basic reference, textbook for this course. E-book available at `www.crcpress.com`

D. Boneh and V. Shoup, *A Graduate Course in Applied Cryptography*. Available at `crypto.stanford.edu/~dabo/cryptobook/`

N. Smart, *Cryptography Made Simple*. Available on-line through U Vic Library.

B. Schneir, *Applied Cryptography*

Less formal, focus on applications. Somewhat out of date.

Other resources: *Journal of Cryptology*, `iacr.org`

My research in cryptography and security

- Verification of cryptographic primitives and protocols
- Extended security notions for encryption (e.g. what happens when we use an encryption scheme to encrypt its own keys.)
- Rational adversaries (e.g. does knowledge of an attacker's utility enhance or ability to encrypt securely)
- Zero-knowledge protocols
- Automatic generation of malware variants for testing of detection software
- Social network anonymization

Cryptography vs. security

N.B. This course is about *cryptography*, not *security*.

While cryptographic tools are important part of computer security, they are not sufficient in themselves to assure security.

See Bruce Schneir's article "Security pitfalls in cryptography",

`http://www.counterpane.com/pitfalls.html`

Cryptography vs. security

“Longer keys don’t always mean more security. Compare the cryptographic algorithm to the lock on your front door. Most door locks have four metal pins, each of which can be in one of ten positions. A key sets the pins in a particular configuration. If the key aligns them all correctly, then the lock opens. So there are only 10,000 possible keys, and a burglar willing to try all 10,000 is guaranteed to break into your house. But an improved lock with 10 pins, making 10 billion possible keys, probably won’t make your house more secure. Burglars don’t try every possible key (a brute-force attack); most aren’t even clever enough to pick the lock (a cryptographic attack against the algorithm). They smash windows, kick in doors, disguise themselves as policemen, or rob keyholders at gunpoint. One ring of art thieves in California defeated home security systems by taking a chainsaw to the house walls. Better locks don’t help against these attacks”

Classical cryptology

Classically, *cryptology* is the science concerned with secret communication.

Cryptography is the branch of cryptology concerned with creating systems which enable secret communication.

Cryptanalysis is the branch of cryptology concerned with finding and exploiting flaws in cryptographic systems.

We will use the term *cryptography* in a more general sense: the design of tools (primitives), schemes, protocols and systems to achieve specified *security* goals

Security goals include: secrecy (confidentiality), authentication (integrity), verifiability, non-repudiation, etc.

Topics

The list of topics that we cover will include some of the following:

- Classical cryptography and cryptanalysis
- Entropy and secrecy
- DES and related block ciphers
- Differential cryptanalysis
- AES
- Modes of operation for block ciphers
- Public key systems: RSA, El Gamal, probalistic encryption
- Digital signature schemes
- Cryptographic hash functions
- Authentication
- Key establishment and distribution
- Pseudo-randomness
- Zero-knowledge
- Provable security

Our primary focus is on encryption and authentication

Breakdown

1. Foundations – history, information-theoretic security, pseudorandomness and formal definitions of computational security

2. Private key encryption and authentication – DES, cryptanalysis, maybe AES, MAC's based on block ciphers and hash functions

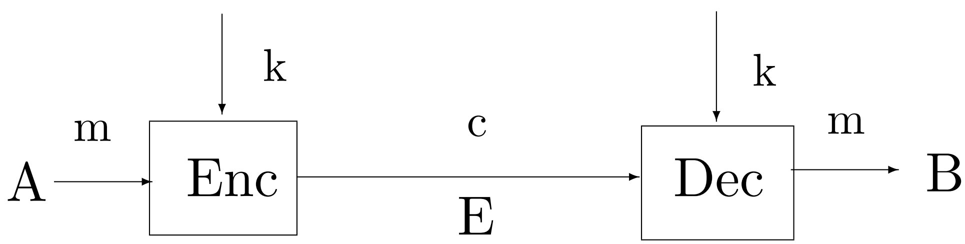
3. Public-key primitives – RSA, El Gamal, encryption, digital signatures

Private-key cryptography

Until the 1970's, all cryptography was private-key, i.e., based on a single key shared by the communicating parties.

Sometimes called *symmetric* cryptography, since the same key is used for encryption and decryption.

Schematically, we have



Here, m is the *message* or *plaintext*, c is the *ciphertext*, and k is the *key*.

Private-key cryptography

We will use the following notation

$\mathcal{P}$ is the set of *plaintexts* or *messages*

$\mathcal{C}$ is the set of *ciphertexts*

$\mathcal{K}$ is the set of *keys* (the *keyspace*)

A *private-key cryptosystem* as a 3-tuple $\langle \mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec} \rangle$, where

Gen is the (randomized) *key generation algorithm*

$\mathsf{Enc} : \mathcal{K} \times \mathcal{P} \to \mathcal{C}$ is the (randomized) *encryption algorithm*

and $\mathsf{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{P}$ is the *decryption algorithm*

Enc and Dec satisfy the basic correctness property: for all $k \in \mathcal{K}$ and $m \in \mathcal{P}$, $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$

Cryptanalysis of private-key cryptosystems

Basic questions

1. What information is available to an adversary?

2. What is the computational power of the adversary?

3. How is adversary *success* determined? (i.e. when has adversary broken the encryption)

Attack models

- *Ciphertext only*
  Given: $c_1, c_2, \ldots, c_n$ where $c_i = E_k(m_i)$
  Infer: k, or if lacking this ability, as many of $m_1, \ldots, m_n$ as possible
- *Known plaintext*
  Given: $m_1, E_k(m_1), m_2, E_k(m_2), \ldots, m_n, E_k(m_n)$
  Infer: k, or m_{n+1} given $c_{n+1} = E_k(m_{n+1})$
- *Chosen plaintext*
  Given: $c_1, c_2, \ldots, c_n$, where $c_i = E_k(m_i)$ for (adaptively) chosen $m_1, \ldots, m_n$
  Infer: k, or m_{n+1} given $c_{n+1} = E_k(m_{n+1})$
- *Chosen ciphertext*
  Given: $m_1, m_2, \ldots, m_n$, where $c_i = E_k(m_i)$ for (adaptively) chosen $c_1, \ldots, c_n$
  Infer: k, or m_{n+1} given $c_{n+1} = E_k(m_{n+1})$

While some of these stronger attack models may seem unrealistic, we will later discuss how they may arise in practice.

Adversary power

- Realitistically, an advesary must be a computational agent – it is reasonable to bound the compuational resources (e.g. time, space)
- *Computational* vs *information-theoretic* securty
- Our primary focus will be computational, although we will consider the information-theoretic setting for motivation and historical background
- The computational setting leads us to a basic paradigm of modern cryptography: *reduction-based security*
  - A system is secure if breaking the system would mean solving a problem which is assumed to be *computationally hard*
  - We will have much more to say about this later in the course

Adversary success

This is an essential ingredient in defining security. What are the possibilities?

1. Recovering secret key

2. Recovering encrypted (challenge) message

3. Recoving *any information* about the message

Why might we favour (3) over (1) or (2)? What exactly do we mean by *any information*?

Kerckhoff's Principle

Auguste Kerkhoff *La Cryptographie Militaire*, 1883

Stated six principles for the design of military ciphers, including

> *The cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience*

(cf. Shannon "the enemy knows the system")

In short, we can't rely on the secrecy of our encryption algorithm – security must derive from the secrecy of the key.