Lecture 3

Encryption

Suggested readings:

- **Chs 1 & 2 in KPS**
- **Ch 1 in Stinson (recommended)**

1

Encryption Principles

A cryptosystem has (at least) five ingredients:

1. Plaintext
2. Secret Key
3. Ciphertext
4. Encryption algorithm
5. Decryption algorithm

Security usually depends on the secrecy of the key, not the secrecy of the algorithm (i.e., the open design model!)

2

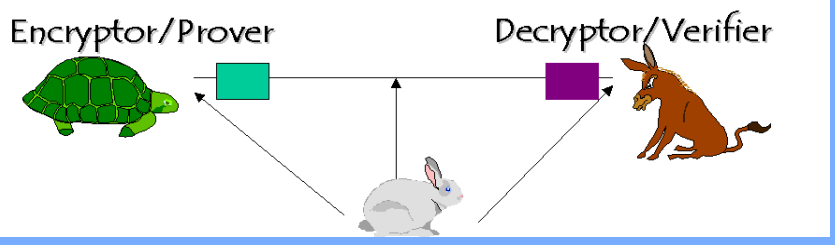
Crypto Basics

Crypto Attacks:

- ciphertext only
- known plaintext
- chosen plaintext
- chosen ciphertext

Cryptosystem:

- *P -- plaintext*
- *C -- ciphertext*
- *K -- keyspace*
- *E -- encryption rules*
- *D -- decryption rules*



3

Average time required for exhaustive key search (for brute force attacks)

Key Size (bits)	Number of Alternative Keys	Time required at 10^6 Decr/μs
32	$2^{32} = 4.3 \times 10^9$	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	10 hours
128	$2^{128} = 3.4 \times 10^{38}$	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	5.9×10^{30} years

4

Types of Attainable Security

• Perfect, unconditional or "information theoretic": the security is evident free of any assumptions

• Reducible or "provable": security can be shown to be based on some common (often unproven) assumptions, e.g., the conjectured difficulty of factoring large integers

• Ad hoc: the security seems good → often snake oil…

Take a look at (strongly recommended):

http://www.ciphersbyritter.com/GLOSSARY.HTM

5

Computational Security

- Encryption scheme is *computationally secure* if
  - cost of breaking it (via brute force) exceeds the value of the encrypted information; or
  - time required to break it exceeds useful lifetime of the encrypted information
- Most good modern schemes we will see are considered computationally secure
  - Usually rely on very large key-space, impregnable to brute force
- Most advanced schemes rely on lack of effective algorithms for certain hard problems, not on a proven inexistence of such algorithms → reducible security!
  - Such as: factorization, discrete logarithms, quadratic residuosity, etc.

6

Cryptosystems

Classified along three dimensions:

- Type of operations used for transforming plaintext into ciphertext
  - Binary arithmetic: shifts, XORs, ANDs, etc.
    - Typical for **conventional** encryption
  - Integer arithmetic
    - Typical for public key encryption
- Number of keys used
  - Symmetric or conventional (single key used)
  - Asymmetric or public-key (2 keys: 1 to encrypt, 1 to decrypt)
- How plaintext is processed:
  - One bit at a time
  - A string of any length
  - A block of bits

7

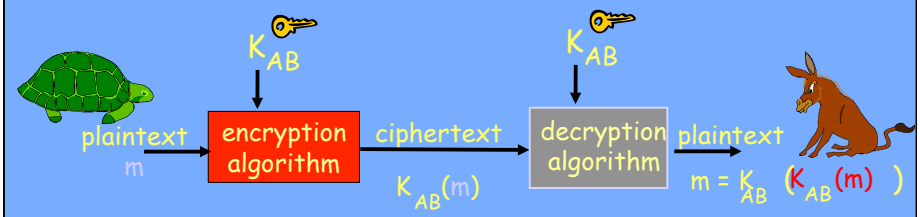
Complexity reminder/re-cap

- **P: problems that can be solved in polynomial time, i.e., problems that can be solved/decided "efficiently"**
- **NP: broad set of problems that includes P;**
  - **answers can be verified "efficiently";**
  - **solutions can't always be efficiently found.**
- **NP-complete: believed-to-be-hard decision problems in NP, they appear to have no efficient solution; answers are efficiently verifiable, solution to one is never much harder than a solution to another**
- **NP-hard: hardest; cannot be solved by a non-deterministic TM. Many computation version of NP-complete problems are NP-hard.**
- **Examples:**
  - Factoring, discrete log are in NP, not know if in NP-complete or in P
  - Primality testing was recently shown to be in P
  - Knapsack is in NP-complete

For more info, see: **http://www.nist.gov/dads/HTML**

8

Conventional (Symmetric) Cryptography



- Alice and Bob share a key K_{AB} which they somehow agree upon (how?)
  - key distribution / key management problem
  - ciphertext is roughly as long as plaintext
  - examples: Substitution, Vernam OTP, DES, AES

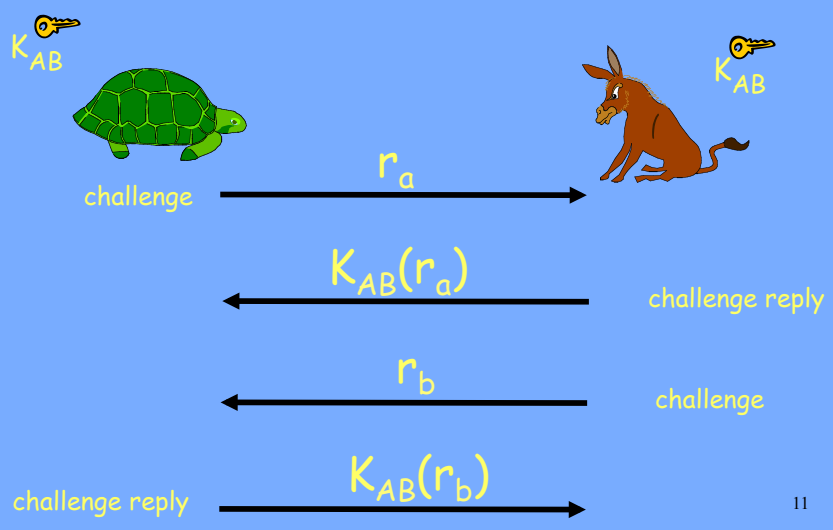
9

Uses of Conventional Cryptography

- Message transmission (confidentiality):
  - Communication over insecure channels
- Secure storage: crypt on Unix (a form of comm-n)
- Strong authentication: proving knowledge of a secret without revealing it:
  - See next slide
  - Eve can obtain chosen <plaintext, ciphertext> pair
  - Challenge should be chosen from a large pool
- Integrity checking: fixed-length checksum for message via secret key cryptography
  - Send MAC along with the message MAC=H(m,K)

10

Challenge-Response Authentication Example



11

Conventional Cryptography

- Advantages
  - high data throughput
  - relatively short key size
  - primitives to construct various cryptographic mechanisms
- Disadvantages
  - key must remain secret at both ends
  - key must be distributed securely and efficiently
  - relatively short key lifetime

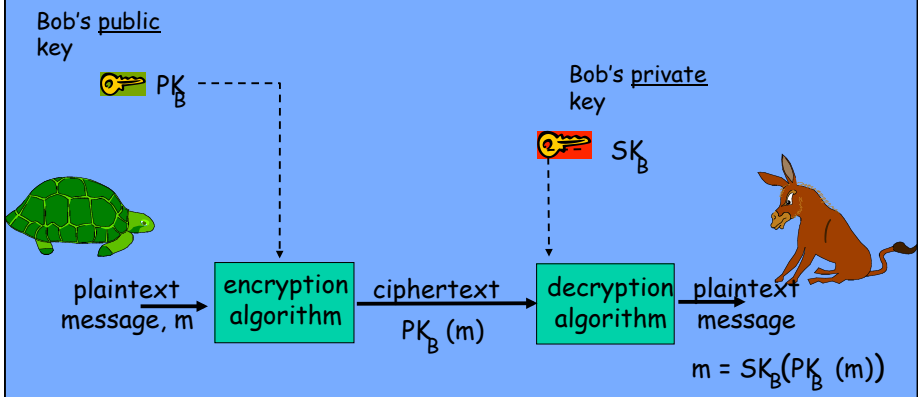
12

Public Key Cryptography

- Asymmetric cryptography
- Invented in 1974-1978
- Two keys: private (SK), public (PK)
  - Encryption: with public key;
  - Decryption: with private key
  - Digital Signatures: Signing by private key; Verification by public key. i.e., "encrypt" message digest/hash -- $h(m)$ -- with private key
    - Authorship (authentication)
    - Integrity: Similar to MAC
    - Non-repudiation: can't do with secret key cryptography
- Much **slower** than conventional cryptography
  - Often used together with conventional cryptography, e.g., to encrypt session keys

13

Public key cryptography



14

Uses of Public Key Cryptography

- Data transmission (confidentiality):
  - Alice encrypts m_a using PK_B, Bob decrypts it to obtain m_a using SK_b.
- Secure Storage: encrypt with own public key, later decrypt with own private key
- Authentication:
  - No need to store secrets, only need *public* keys.
  - Secret key cryptography: need to share *secret* key for every person one communicates with
- Digital Signatures (authentication, integrity, non-repudiation)

15

Public Key Cryptography

- Advantages
  - only the private key must be kept secret
  - relatively long life time of the key
  - more security services
  - relatively efficient digital signatures mechanisms
- Disadvantages
  - low data throughput
  - much larger key sizes
  - distribution/revocation of public keys
  - security based on conjectured hardness of certain computational problems

16

Comparison Summary

- Public key
  - encryption, signatures (esp., non-repudiation) and key management
- Conventional
  - encryption and some data integrity applications
- Key sizes
  - Keys in public key crypto must be larger (e.g., 1536 bits for RSA) than those in conventional crypto (e.g., 112 bits for 3-DES or 256 bits for AES)
    - most attacks on "good" conventional cryptosystems are exhaustive key search (brute force)
    - public key cryptosystems are subject to "short-cut" attacks (e.g., factoring large numbers in RSA)

17

"Modern" Block Ciphers

DES

18

Feistel Cipher Structure

- Virtually all conventional block encryption algorithms, including DES, have a structure first described by Horst Feistel of IBM in 1973
- Specific realization of a Feistel Network depends on the choice of the following parameters and features:

19

Feistel Cipher Structure

- **Block size:** larger block sizes mean greater security
- **Key Size:** larger key size means greater security
- **Number of rounds:** multiple rounds offer increasing security
- **Subkey generation algorithm:** greater complexity will lead to greater difficulty of cryptanalysis.
- **Fast software en/de-cryption:** speed of execution of the algorithm becomes a concern

20

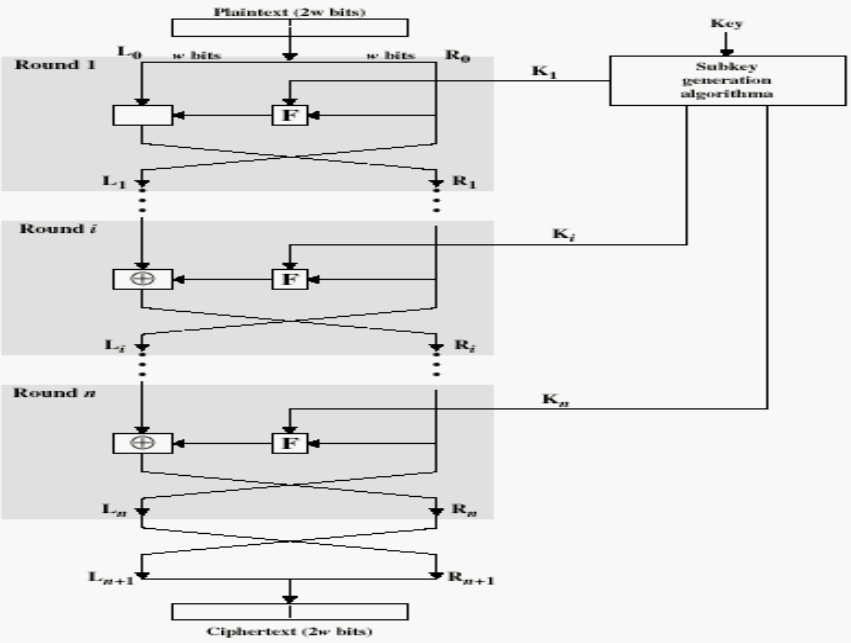


Figure 2.2 Classical Feistel Network

Block Ciphers

- Originated with early 1970's IBM effort to develop banking security systems
- First result was Lucifer, most common variant has 128-bit key and block size
- Wasn't secure in any of its variants
- Called a Feistel or product cipher
- f()-function is a simple transformation, doesn't have to be reversible
- Each step is called a round; the more rounds, the greater the security (to a point)
- Most famous example of this design is DES

22

Conventional Encryption Algorithms - DES

- Data Encryption Standard (DES)
  - Most widely used encryption method
  - Block cipher (in native ECB mode)
  - Plaintext processed in 64-bit blocks
  - Key is 56 bits

23