Cryptography
(the art of scrambling)

Beside programming e-commerce applications, what other cs issues are there?

- Application (web) design: HCI
- Data mining
- Server and client security (how can we protect our systems and data
  - hackers
  - malicious code
  - denial-of-service (DOS) attacks
  - privacy
- Electronic document authentication
Major issues

- Secret message
  - Write a message that only your friend can read while passing it through enemy lines

- Message authentication

  Dear Jean,
  I love you
  George
  
  This is $1000 Dollar (US!!)

more formally …

1. Confidentiality:
   - how can I make sure that an eavesdropper cannot read my message

2. Authentication:
   - how do I know that the message is from a particular person?

3. Message integrity:
   - how do I know that the message has not been modified on its travel?
Basic Cryptography

- Ciphers
- Symmetric Key Algorithms
- Public Key Algorithms
- Message Digests
- Digital Signatures
- Trust networks

(Outline)
(1. Confidentiality)
(2. Authentication)
(3. Message integrity)

1. Confidentiality
Encryption-Decryption

- Main idea: scramble a message so that it is impossible (or very difficult) to read the message unless I tell you another secret that makes it possible to de-scramble it.

- Two route solution to privacy:
  - Secret scrambling procedure (not good)
  - Secret input to scrambling procedure (good)

- Key could be

```
guvf zrffntr vf frperg
  ___s __ss__ _s s_____ 
  __is __ss__ is s____
```
guvf zrffntr vf frperg
___s ___ss__ _s s____
__is ___ss__ is s____
__is _ess_e is se__e_
this _ess_e is se__et
this message is secret

ROT13 algorithm (cipher):

abcdefghijklmnopqrstuvwxyz

↓

nopqrstuvwxyzabcdefghijklm
Definitions
(Encryption, Decryption, Plaintext, Ciphertext)

Types of cipher:
- Stream cipher
  - Each bit (or byte) is encrypted or decrypted individually
  - Simple substitution ciphers (ROT13, XOR)
- Block cipher
  - A sequence of bits (or bytes) is used at each step in the encryption and decryption process (DES, AES)

Symmetric Key Algorithms

Public Key Cryptography
Symmetric Key Algorithms

General:
- Substitution (ROT13, Cryptoquotes)
- Transposition
- XOR
- One Time Pad

} most practical algorithms use a combination of these

Specific algorithms:
- DES (data encryption standard, 56-bit key, Triple-DES)
- IDEA (international data encryption algorithm, 128-bit key, patents)
- RC2, RC4, RC5 (Ronald Rivest RSA, variable key length)
- Rijndael (AES) (advanced encryption standard adapted in 2001)

Rijndael: Iterated Block Cipher

- 10/12/14 times applying the same round function

- Round function: uniform and parallel, composed of 4 steps

- Each step has its own particular function:
  1. ByteSub: nonlinearity
  2. ShiftRow: inter-column diffusion
  3. MixColumn: inter-byte diffusion within columns
  4. Round key addition
Round step 1: ByteSub

Bytes are transformed by applying invertible S-box.
One single S-box for the complete cipher
High non-linearity

Round step 2: MixColumn

Bytes in columns are linearly combined
Based on theory of error-correcting codes
High intra-column diffusion
Round step 3: ShiftRow

- Rows are shifted over 4 different offsets
- Interaction with MixColumn
- High diffusion over multiple rounds

Round step 4: Key addition

- Makes round function key-dependent
- Computation of round keys: “keep it simple”
- Small number of operations
- Small amount of memory
What is an appropriate length for a key?

Comparison of cryptographic algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Description</th>
<th>Key Length</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blowfish</td>
<td>Block cipher developed by Schneier</td>
<td>4–448 bits</td>
<td>▲</td>
</tr>
<tr>
<td>DES</td>
<td>Data Encryption Standard adopted as a U.S. government standard in 1977</td>
<td>56 bits</td>
<td>▼</td>
</tr>
<tr>
<td>IDEA</td>
<td>Block cipher developed by Harvey and K尽力a</td>
<td>128 bits</td>
<td>▲</td>
</tr>
<tr>
<td>RC5</td>
<td>Block cipher developed by Rivest</td>
<td>1–128 bits</td>
<td>▲</td>
</tr>
<tr>
<td>RC6</td>
<td>Stream cipher developed by Rivest</td>
<td>1–128 bits</td>
<td>▲</td>
</tr>
<tr>
<td>RC7</td>
<td>Block cipher developed by Ronald Rivest and published in 1991</td>
<td>128–256 bits</td>
<td>▲</td>
</tr>
<tr>
<td>RC6-48</td>
<td>AES finalists developed by RSA Labs</td>
<td>128–256 bits</td>
<td>▲</td>
</tr>
<tr>
<td>Trifid</td>
<td>NIST selection for AES, developed by Boneh and Silverman</td>
<td>128–256 bits</td>
<td>▲</td>
</tr>
<tr>
<td>Serpent</td>
<td>AES finalist developed by Anderson, Biham, and Knudsen</td>
<td>128–256 bits</td>
<td>▲</td>
</tr>
<tr>
<td>Triple-DES</td>
<td>A four-field application of the DES algorithm</td>
<td>112–168–224–288 bits</td>
<td>▲</td>
</tr>
<tr>
<td>Twofish</td>
<td>AES candidate developed by Schneier</td>
<td>128–256 bits</td>
<td>▲</td>
</tr>
</tbody>
</table>

Key to ratings:
- ▲: Excellent algorithm. This algorithm is widely used and is believed to be secure, provided that keys of sufficient length are used.
- ▼: Algorithm is considered strong but is being phased out for other algorithms that are faster or thought to be more secure.
- ▼: Algorithm appears strong but will not be widely deployed because it was not chosen as the AES standard.
- ▲: Use of this algorithm is no longer recommended because of short key length or mathematical weaknesses. Data encrypted with this algorithm should be reencrypted soon with usual precautions, but would not withstand a determined attack by a moderately-advanced attacker.
- ▼: a bit old
2. Authentication

It's me .... really!

Key distribution problem

- How to ship the ‘code-book’?
- Solutions
  - Doubly padlocked box exchange
  - Diffie-Hellman key exchange
  - Public-key cryptography
    - RSA
    - elliptic curve cryptography
**Diffie-Hellman key exchange (1)**

<table>
<thead>
<tr>
<th></th>
<th>Alice</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secret part generation</td>
<td>![Diagram]</td>
<td>![Diagram]</td>
</tr>
<tr>
<td>One-way function</td>
<td>![Diagram]</td>
<td>![Diagram]</td>
</tr>
<tr>
<td>Swap</td>
<td>![Diagram]</td>
<td>![Diagram]</td>
</tr>
<tr>
<td>Key generation</td>
<td>![Diagram]</td>
<td>![Diagram]</td>
</tr>
</tbody>
</table>

**Diffie-Hellman key exchange (2)**

<table>
<thead>
<tr>
<th></th>
<th>Alice</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secret part generation</td>
<td>Choose a secret number A=3</td>
<td>Choose a secret number B=6</td>
</tr>
<tr>
<td>One-way function</td>
<td>Use one-way function a=7^4(mod 11)=2</td>
<td>Use one-way function b=7^3(mod 11)=4</td>
</tr>
<tr>
<td>Swap</td>
<td>b=4</td>
<td>a=2</td>
</tr>
<tr>
<td>Key generation</td>
<td>Another one-way function k=b^4(mod 11)=9</td>
<td>Another one-way function k=a^6(mod 11)=9</td>
</tr>
</tbody>
</table>
The Diffie-Hellman key exchange was the first widely recognized solution to the key exchange problem.

Can only be used to exchange key. Symmetric key cryptographic methods can be used to exchange secret messages.

Fairly elaborate exchange of messages.

Diffie-Hellman key exchange (3)

Public Key Cryptography

A public key - private key pair are used, one for encryption and the other for decryption.

Two application modes:
- Confidentiality
- Authentication
Public Key Cryptography a la RSA

Public Key:
- $n$ - product of two primes, $p$ and $q$
  
  ($p$ and $q$ are secret)
- $e$ - relatively prime to $(p-1)(q-1)$
  
  (have no common divisor)

Private Key:
- $d$ - $e^{-1} \mod (p-1)(q-1)$

Encrypting:
- $c = m^e \mod n$

Decrypting:
- $m = c^d \mod n$

Example:
- Let $p=3$, $q=11$
- $n=pq=33$
- $e$ must be relatively prime to $(p-1)(q-1)=20$
- choose $e=7$, then $d = 7^{-1} \mod 20 = 3$
- Plaintext is 3,4,2
  
  $(m_1=3, m_2=4, m_3=2)$
- $c_1 = m_1^e \mod n = 3^7 \mod 33 = 9$
- $c_2 = m_2^e \mod n = 4^7 \mod 33 = 15$
- $c_3 = m_3^e \mod n = 2^7 \mod 33 = 29$
- Ciphertext is 9,15,29
  
  $m_1 = c_1^d \mod n = 9^3 \mod 33 = 3$
  
  $m_2 = c_2^d \mod n = 15^3 \mod 33 = 4$
  
  $m_3 = c_3^d \mod n = 29^3 \mod 33 = 2$
- Plaintext is 3,4,2

3. Message Integrity
Message Digests & Hash function

- A message digest is a one-way function which maps the information contained in a (small or large) file to a single large number, typically between 128 bits and 256 bits in length.

- A good message digest function should have the following properties:
  - Every bit of the output is influenced by every bit of the input
  - Changing a single bit in the input results in every output bit having a 50% chance of changing
  - Given an input file, its corresponding digest, and the digest function, it is computationally infeasible to produce another input file which maps to the same digest

Message Digests (continued)

- Standard encryption algorithm
  - e.g. use last block in cipher feedback mode
  - Provide good message digest code
  - Computationally more demanding than other specialized functions
- MD5
  - One widely used message digest algorithm from a series of algorithms developed by Ronald Rivest
  - Does not rely on a secret key and is therefore not suitable as MAC without further provisions
- HMAC
  - The Hashed Message Authentication Code uses a shared secret key in combination with a message digest function to produce a secret message authentication code
  - Since an attacker doesn’t know the secret, the attacker cannot produce a correct authentication code if they alter the message
  - Fast to calculate, can be used as digital signature. However, a shared secret key is used.
- SHA-1
  - Developed by the NSA for use with the Digital Signature Standard

Operation of a message digest function to produce a message authentication code
RSA Digital Signature

Originator

Message

Hash Function

Digest

Encrypt

Signature

Private Key

Transmitted Message

Message

Signature

Recipient

Message

Hash Function

Public Key

Decrypt

Expected Digest

Actual Digest

If actual and expected match, the signature is verified

Types of authentication

- What you know  (username and password)
- What you have  (token, smart card)
- What you are  (biometrics)
- Where you are  (location security)
Digital Certificates

- Need a system for pairing public keys to identification information

- Certification authority (or trusted third party) issues a certificate which pairs identification information with a public key, signed with the certification authority's private key

- User must trust the certification authority, and have a valid copy of the certification authority's public key
Certification Paths

- More than one Certification Authority will be required
- If CAs trust one another, they can issue certificates for each other’s public keys
- This leads to a recursively defined path from a user under one CA to a user under another CA
Blind Signatures

- Analogy – place a document to be signed inside an envelope with a carbon paper over it, and have the signing party sign the envelope. Signing the envelope causes the document to be signed because of the carbon paper inside.

![Blind signature analogy for withdrawing Ecash coins.](image)

Figure 6.2 Blind signature analogy for withdrawing Ecash coins.

PGP: Pretty Good Privacy

- Implementation of best available cryptographic algorithms for confidentiality and authentication and integration into a freely available general-purpose application
- Package, source code, and documentation available on the web
- Low-cost commercial version initially from Network Associates (now from PGP Corporation)
- Includes AES, 3DES, CAST, IDEA; RSA DSS, Diffie-Hellman; SHA1; key management, …

Philip Zimmermann