Chapter 9: Basic Cryptography

• Classical Cryptography
  – Cæsar cipher
  – Vigènere cipher
  – DES

• Public Key Cryptography
  – Diffie-Hellman
  – RSA
Cryptosystem

- Quintuple \((E, D, M, K, C)\)
  - \(M\) set of plaintexts
  - \(K\) set of keys
  - \(C\) set of ciphertexts
  - \(E\) set of encryption functions \(e: M \times K \rightarrow C\)
  - \(D\) set of decryption functions \(d: C \times K \rightarrow M\)
Example

- Example: Cæsar cipher
  - \( \mathcal{M} = \{ \text{sequences of letters} \} \)
  - \( \mathcal{K} = \{ i \mid i \text{ is an integer and } 0 \leq i \leq 25 \} \)
  - \( \mathcal{E} = \{ E_k \mid k \in \mathcal{K} \text{ and for all letters } m, \ E_k(m) = (m + k) \mod 26 \} \)
  - \( \mathcal{D} = \{ D_k \mid k \in \mathcal{K} \text{ and for all letters } c, \ D_k(c) = (26 + c - k) \mod 26 \} \)
  - \( \mathcal{C} = \mathcal{M} \)
Attacks

- Opponent whose goal is to break cryptosystem is the adversary
  - Assume adversary knows algorithm used, but not key

- Three types of attacks:
  - ciphertext only: adversary has only ciphertext; goal is to find plaintext, possibly key
  - known plaintext: adversary has ciphertext, corresponding plaintext; goal is to find key
  - chosen plaintext: adversary may supply plaintexts and obtain corresponding ciphertext; goal is to find key
Basis for Attacks

• Mathematical attacks
  – Based on analysis of underlying mathematics

• Statistical attacks
  – Make assumptions about the distribution of letters (1-gram), pairs of letters (2-gram), triplets of letters (3-gram), etc.
    • Called *models of the language*
  – Examine ciphertext, correlate properties with the assumptions.
Classical Cryptography

- Sender, receiver share common key
  - Keys may be the same, or trivial to derive from one another
  - Sometimes called *symmetric cryptography*
- Two basic types
  - Transposition ciphers
  - Substitution ciphers
  - Combinations are called *product ciphers*
Transposition Cipher

• Rearrange letters in plaintext to produce ciphertext

• Example (Rail-Fence Cipher)
  – Plaintext is HELLO WORLD
  – Rearrange as

    HLOOL
    ELWRD

  – Ciphertext is HLOOL ELWRD
Attacking the Cipher

- Anagramming (Attacking transposition cipher)
  - If 1-gram frequencies match English frequencies, but other \( n \)-gram frequencies do not, probably transposition
  - Rearrange letters to form \( n \)-grams with highest frequencies
Example

- Ciphertext: HLOOLELWRD
- Frequencies of 2-grams beginning with H
  - HE 0.0305
  - HO 0.0043
  - HL, HW, HR, HD < 0.0010
- Frequencies of 2-grams ending in H
  - WH 0.0026
  - EH, LH, OH, RH, DH ≤ 0.0002
- Implies E follows H
Example

- Arrange so the H and E are adjacent
  HE
  LL
  OW
  OR
  LD

- Read off across, then down, to get original plaintext
Substitution Ciphers

- Change characters in plaintext to produce ciphertext
- Example (Caesar cipher)
  - Plaintext is HELLO WORLD
  - Change each letter to the third letter following it (X goes to A, Y to B, Z to C)
    - Key is 3, usually written as letter ‘D’
  - Ciphertext is KHOOR ZRUOG
Attacking the Cipher

- Exhaustive search
  - If the key space is small enough, try all possible keys until you find the right one
  - Cæsar cipher has 26 possible keys

- Statistical analysis
  - Compare to 1-gram model of English
Statistical Attack

- Compute frequency of each letter in ciphertext:
  
<table>
<thead>
<tr>
<th>Letter</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>0.1</td>
</tr>
<tr>
<td>H</td>
<td>0.1</td>
</tr>
<tr>
<td>K</td>
<td>0.1</td>
</tr>
<tr>
<td>O</td>
<td>0.3</td>
</tr>
<tr>
<td>R</td>
<td>0.2</td>
</tr>
<tr>
<td>U</td>
<td>0.1</td>
</tr>
<tr>
<td>Z</td>
<td>0.1</td>
</tr>
</tbody>
</table>

- Apply 1-gram model of English
  - Frequency of characters (1-grams) in English is on next slide
# Character Frequencies

<table>
<thead>
<tr>
<th>Character</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0.080</td>
</tr>
<tr>
<td>h</td>
<td>0.060</td>
</tr>
<tr>
<td>n</td>
<td>0.070</td>
</tr>
<tr>
<td>t</td>
<td>0.090</td>
</tr>
<tr>
<td>b</td>
<td>0.015</td>
</tr>
<tr>
<td>i</td>
<td>0.065</td>
</tr>
<tr>
<td>o</td>
<td>0.080</td>
</tr>
<tr>
<td>u</td>
<td>0.030</td>
</tr>
<tr>
<td>c</td>
<td>0.030</td>
</tr>
<tr>
<td>j</td>
<td>0.005</td>
</tr>
<tr>
<td>p</td>
<td>0.020</td>
</tr>
<tr>
<td>v</td>
<td>0.010</td>
</tr>
<tr>
<td>d</td>
<td>0.040</td>
</tr>
<tr>
<td>k</td>
<td>0.005</td>
</tr>
<tr>
<td>q</td>
<td>0.002</td>
</tr>
<tr>
<td>w</td>
<td>0.015</td>
</tr>
<tr>
<td>e</td>
<td>0.130</td>
</tr>
<tr>
<td>l</td>
<td>0.035</td>
</tr>
<tr>
<td>r</td>
<td>0.065</td>
</tr>
<tr>
<td>x</td>
<td>0.005</td>
</tr>
<tr>
<td>f</td>
<td>0.020</td>
</tr>
<tr>
<td>m</td>
<td>0.030</td>
</tr>
<tr>
<td>s</td>
<td>0.060</td>
</tr>
<tr>
<td>y</td>
<td>0.020</td>
</tr>
<tr>
<td>g</td>
<td>0.015</td>
</tr>
<tr>
<td>z</td>
<td>0.002</td>
</tr>
</tbody>
</table>
Statistical Analysis

- \( f(c) \) frequency of character \( c \) in ciphertext
- \( \varphi(i) \) correlation of frequency of letters in ciphertext with corresponding letters in English, assuming key is \( i \)
  
  \[ \varphi(i) = \sum_{0 \leq c \leq 25} f(c)p(c - i) \] so here,

  \[ \varphi(i) = 0.1p(6 - i) + 0.1p(7 - i) + 0.1p(10 - i) + 0.3p(14 - i) + 0.2p(17 - i) + 0.1p(20 - i) + 0.1p(25 - i) \]

- \( p(x) \) is frequency of character \( x \) in English
Correlation: $\varphi(i)$ for $0 \leq i \leq 25$

<table>
<thead>
<tr>
<th>$i$</th>
<th>$\varphi(i)$</th>
<th>$i$</th>
<th>$\varphi(i)$</th>
<th>$i$</th>
<th>$\varphi(i)$</th>
<th>$i$</th>
<th>$\varphi(i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0482</td>
<td>7</td>
<td>0.0442</td>
<td>13</td>
<td>0.0520</td>
<td>19</td>
<td>0.0315</td>
</tr>
<tr>
<td>1</td>
<td>0.0364</td>
<td>8</td>
<td>0.0202</td>
<td>14</td>
<td>0.0535</td>
<td>20</td>
<td>0.0302</td>
</tr>
<tr>
<td>2</td>
<td>0.0410</td>
<td>9</td>
<td>0.0267</td>
<td>15</td>
<td>0.0226</td>
<td>21</td>
<td>0.0517</td>
</tr>
<tr>
<td>3</td>
<td>0.0575</td>
<td>10</td>
<td>0.0635</td>
<td>16</td>
<td>0.0322</td>
<td>22</td>
<td>0.0380</td>
</tr>
<tr>
<td>4</td>
<td>0.0252</td>
<td>11</td>
<td>0.0262</td>
<td>17</td>
<td>0.0392</td>
<td>23</td>
<td>0.0370</td>
</tr>
<tr>
<td>5</td>
<td>0.0190</td>
<td>12</td>
<td>0.0325</td>
<td>18</td>
<td>0.0299</td>
<td>24</td>
<td>0.0316</td>
</tr>
<tr>
<td>6</td>
<td>0.0660</td>
<td>18</td>
<td></td>
<td>25</td>
<td>0.0430</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Result

- Most probable keys, based on $\varphi$:
  - $i = 6$, $\varphi(i) = 0.0660$
    - plaintext EBIIL TLOLA
  - $i = 10$, $\varphi(i) = 0.0635$
    - plaintext AXEEH PHKEW
  - $i = 3$, $\varphi(i) = 0.0575$
    - plaintext HELLO WORLD
  - $i = 14$, $\varphi(i) = 0.0535$
    - plaintext WTAAD LDGAS

- Only English phrase is for $i = 3$
  - That’s the key (3 or ‘D’)

17
Cæsar’s Problem

• Key is too short
  – Can be found by exhaustive search
  – Statistical frequencies not concealed well
    • They look too much like regular English letters

• So make it longer
  – Multiple letters in key
  – Idea is to smooth the statistical frequencies to make cryptanalysis harder
Vigènere Cipher

• Like Cæsar cipher, but use a phrase
• Example
  – Message THE BOY HAS THE BALL
  – Key VIG
  – Encipher using Cæsar cipher for each letter:
    
    | key   | VIG VIG VIG VIG VIG VIG VIG |
    | plain | THE BOY HASTHE BALL           |
    | cipher | OPKWWECIYOPKWIRG             |
Relevant Parts of Tableau

|   | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| A | G | I | V |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| B | G | I | V |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| C | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |   |
| D |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

- Tableau shown has relevant rows, columns only
- Example encipherments:
  - Key V, letter T: follow V column down to T row (giving “O”)
  - Key I, letter H: follow I column down to H row (giving “P”)

20
Useful Terms

- **period**: length of key
  - In earlier example, period is 3
- **tableau**: table used to encipher and decipher
  - Vigènere cipher has key letters on top, plaintext letters on the left
- **polyalphabetic**: the key has several different letters
  - Cæsar cipher is monoalphabetic
Attacking the Cipher

• Approach
  – Establish period; call it $n$
  – Break message into $n$ parts, each part being enciphered using the same key letter
  – Solve each part
    • You can leverage one part from another
One-Time Pad

• A Vigenère cipher with a random key at least as long as the message
  – Provably unbreakable
  – Why? Look at ciphertext DXQR. Equally likely to correspond to plaintext DOIT (key AJIY) and to plaintext DONT (key AJDY) and any other 4 letters
  – Warning: keys must be random, or you can attack the cipher by trying to regenerate the key
    • Approximations, such as using pseudorandom number generators to generate keys, are not random
Overview of the DES

• A block cipher:
  – encrypts blocks of 64 bits using a 64 bit key
  – outputs 64 bits of ciphertext
• A product cipher
  – basic unit is the bit
  – performs both substitution and transposition (permutation) on the bits
• Cipher consists of 16 rounds (iterations) each with a round key generated from the user-supplied key
Encipherment

\[ L_0 \]
\[ L_1 = R_0 \]
\[ R_0 \]
\[ R_1 = L_0 \oplus f(R_0, K_1) \]

\[ f \]

\[ R_{16} = L_{15} - f(R_{15}, K_{16}) \]
\[ L_{16} = R_{15} \]

\[ \text{IP} \]
\[ \text{IP}^S_1 \]
\[ \text{output} \]
The $f$ Function

- $R_{i\delta 1}$ (32 bits)
- $E$
- $R_{i\delta 1}$ (48 bits)
- $K_i$ (48 bits)
- $\oplus$
- $S_1$ $S_2$ $S_3$ $S_4$ $S_5$ $S_6$ $S_7$ $S_8$
- 6 bits into each
- 4 bits out of each
- $P$
- 32 bits
Controversy

• Considered too weak
  – Diffie, Hellman said in a few years technology would allow DES to be broken in days
    • Design using 1999 technology published
  – Design decisions not public
    • S-boxes may have backdoors
Undesirable Properties

- 4 weak keys
  - They are their own inverses
- 12 semi-weak keys
  - Each has another semi-weak key as inverse
- Complementation property
  - $\text{DES}_k(m) = c \Rightarrow \text{DES}_k(m') = c'$
- S-boxes exhibit irregular properties
  - Distribution of odd, even numbers non-random
  - Outputs of fourth box depends on input to third box
Cryptanalysis

• Differential cryptoanalysis
  – A chosen ciphertext attack
    • Requires $2^{47}$ plaintext, ciphertext pairs
  – Revealed several properties
    • Small changes in S-boxes reduce the number of pairs needed
    • Making every bit of the round keys independent does not impede attack

• Linear cryptanalysis improves result
  – Requires $2^{43}$ plaintext, ciphertext pairs
Current Status of DES

- Design for computer system, associated software that could break any DES-enciphered message in a few days published in 1998
- Several challenges to break DES messages solved using distributed computing
- NIST selected Rijndael as Advanced Encryption Standard, successor to DES
  - Designed to withstand attacks that were successful on DES
Public Key Cryptography

• Two keys
  – *Private key* known only to individual
  – *Public key* available to anyone

• Idea
  – Confidentiality: encipher using public key, decipher using private key
  – Integrity/authentication: encipher using private key, decipher using public one
Requirements

1. It must be computationally easy to encipher or decipher a message given the appropriate key
2. It must be computationally infeasible to derive the private key from the public key
3. It must be computationally infeasible to determine the private key from a chosen plaintext attack
Diffie-Hellman

• Compute a common, shared key
  – Called a *symmetric key exchange protocol*

• Based on discrete logarithm problem
  – Given integers $n$ and $g$ and prime number $p$, compute $k$ such that $n = g^k \mod p$
  – Solutions known for small $p$
  – Solutions computationally infeasible as $p$ grows large
Algorithm

• Constants: prime $p$, integer $g \neq 0, 1, p–1$
  – Known to all participants

• Anne chooses private key $k_{Anne}$, computes public key $K_{Anne} = g^{k_{Anne}} \mod p$

• To communicate with Bob, Anne computes $K_{shared} = K_{Bob}^{k_{Anne}} \mod p$

• To communicate with Anne, Bob computes $K_{shared} = K_{Anne}^{k_{Bob}} \mod p$
  – It can be shown these keys are equal
Example

- Assume $p = 53$ and $g = 17$
- Alice chooses $k_{Alice} = 5$
  - Then $K_{Alice} = 17^5 \mod 53 = 40$
- Bob chooses $k_{Bob} = 7$
  - Then $K_{Bob} = 17^7 \mod 53 = 6$
- Shared key:
  - $K_{Bob}^{k_{Alice}} \mod p = 6^5 \mod 53 = 38$
  - $K_{Alice}^{k_{Bob}} \mod p = 40^7 \mod 53 = 38$
RSA

• Exponentiation cipher
• Relies on the difficulty of determining the number of numbers relatively prime to a large integer $n$
Background

- **Totient function** $\phi(n)$
  - Number of positive integers less than $n$ and relatively prime to $n$
    - *Relatively prime* means with no factors in common with $n$
  - **Example**: $\phi(10) = 4$
    - 1, 3, 7, 9 are relatively prime to 10
  - **Example**: $\phi(21) = 12$
    - 1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20 are relatively prime to 21
Algorithm

• Choose two large prime numbers $p, q$
  – Let $n = pq$; then $\phi(n) = (p-1)(q-1)$
  – Choose $e < n$ such that $e$ is relatively prime to $\phi(n)$.
  – Compute $d$ such that $ed \mod \phi(n) = 1$
• Public key: $(e, n)$; private key: $d$
• Encipher: $c = m^e \mod n$
• Decipher: $m = c^d \mod n$
Example: Confidentiality

- Take $p = 7$, $q = 11$, so $n = 77$ and $\phi(n) = 60$
- Alice chooses $e = 17$, making $d = 53$
- Bob wants to send Alice secret message HELLO (07 04 11 11 14)
  - $07^{17} \mod 77 = 28$
  - $04^{17} \mod 77 = 16$
  - $11^{17} \mod 77 = 44$
  - $11^{17} \mod 77 = 44$
  - $14^{17} \mod 77 = 42$
- Bob sends 28 16 44 44 42
Example

• Alice receives 28 16 44 44 42
• Alice uses private key, $d = 53$, to decrypt message:
  – $28^{53} \mod 77 = 07$
  – $16^{53} \mod 77 = 04$
  – $44^{53} \mod 77 = 11$
  – $44^{53} \mod 77 = 11$
  – $42^{53} \mod 77 = 14$
• Alice translates message to letters to read HELLO
  – No one else could read it, as only Alice knows her private key and that is needed for decryption
Example: Integrity/Authentication

- Take \( p = 7, q = 11 \), so \( n = 77 \) and \( \phi(n) = 60 \)
- Alice chooses \( e = 17 \), making \( d = 53 \)
- Alice wants to send Bob message HELLO (07 04 11 11 14) so Bob knows it is what Alice sent (no changes in transit, and authenticated)
  - \( 07^{53} \mod 77 = 35 \)
  - \( 04^{53} \mod 77 = 09 \)
  - \( 11^{53} \mod 77 = 44 \)
  - \( 11^{53} \mod 77 = 44 \)
  - \( 14^{53} \mod 77 = 49 \)
- Alice sends 35 09 44 44 49
Example

- Bob receives 35 09 44 44 49
- Bob uses Alice’s public key, $e = 17, n = 77$, to decrypt message:
  - $35^{17} \text{ mod } 77 = 07$
  - $09^{17} \text{ mod } 77 = 04$
  - $44^{17} \text{ mod } 77 = 11$
  - $44^{17} \text{ mod } 77 = 11$
  - $49^{17} \text{ mod } 77 = 14$
- Bob translates message to letters to read HELLO
  - Alice sent it as only she knows her private key, so no one else could have enciphered it
  - If (enciphered) message’s blocks (letters) altered in transit, would not decrypt properly
Example: Both

- Alice wants to send Bob message HELLO both enciphered and authenticated (integrity-checked)
  - Alice’s keys: public (17, 77); private: 53
  - Bob’s keys: public: (37, 77); private: 13
- Alice enciphers HELLO (07 04 11 11 14):
  - \((07^{53} \mod 77)^{37} \mod 77 = 07\)
  - \((04^{53} \mod 77)^{37} \mod 77 = 37\)
  - \((11^{53} \mod 77)^{37} \mod 77 = 44\)
  - \((11^{53} \mod 77)^{37} \mod 77 = 44\)
  - \((14^{53} \mod 77)^{37} \mod 77 = 14\)
- Alice sends 07 37 44 44 14
Security Services

• Confidentiality
  – Only the owner of the private key knows it, so text enciphered with public key cannot be read by anyone except the owner of the private key

• Authentication
  – Only the owner of the private key knows it, so text enciphered with private key must have been generated by the owner
More Security Services

• Integrity
  – Enciphered letters cannot be changed undetectably without knowing private key

• Non-Repudiation
  – Message enciphered with private key came from someone who knew it
Warnings

• Encipher message in blocks considerably larger than the examples here
  – If 1 character per block, RSA can be broken using statistical attacks (just like classical cryptosystems)
  – Attacker cannot alter letters, but can rearrange them and alter message meaning
    • Example: reverse enciphered message of text ON to get NO
Key Points

• Two main types of cryptosystems: classical and public key

• Classical cryptosystems encipher and decipher using the same key
  – Or one key is easily derived from the other

• Public key cryptosystems encipher and decipher using different keys
  – Computationally infeasible to derive one from the other