Cryptography

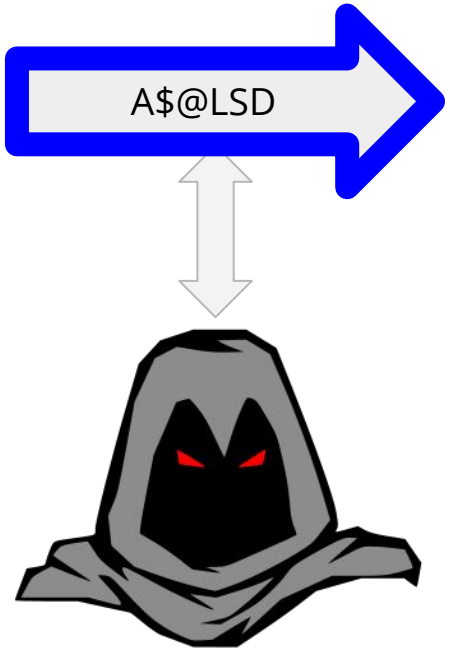
24-02-2018

Anson Ho

香港電腦奧林匹克競賽
Hong Kong Olympiad in Informatics

Origin

Alice



Eve

Bob

Application - Military Communication

Enigma

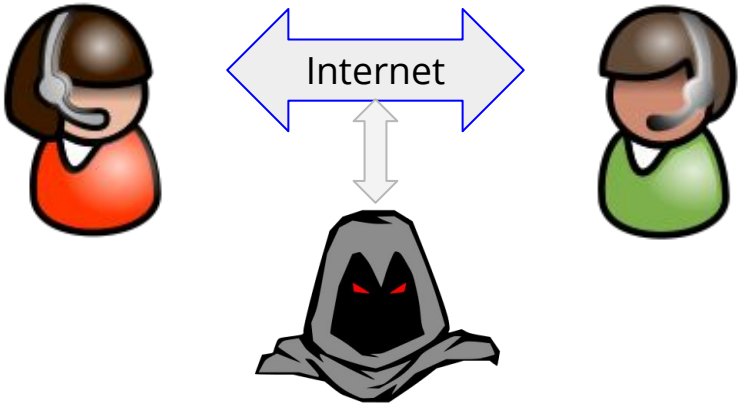
German soldier A

German soldier B

Alan Turing

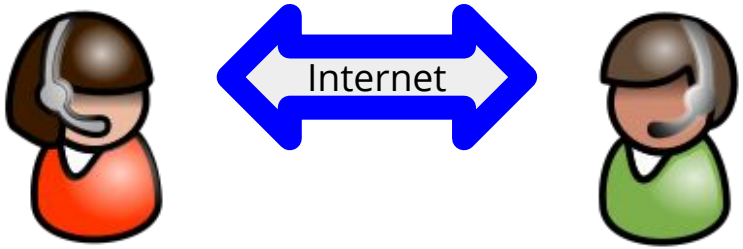
Application - HTTPS

HTTP



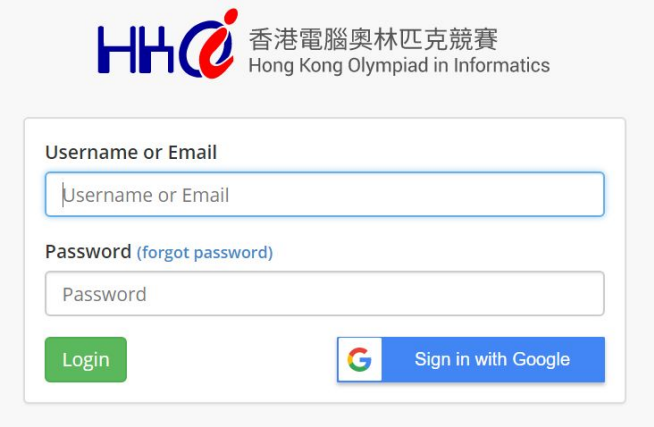
MITM

HTTPS



- Man-in-the-middle attack
  - middle = VPN, WiFI router, ...
  - attack = record / modify data

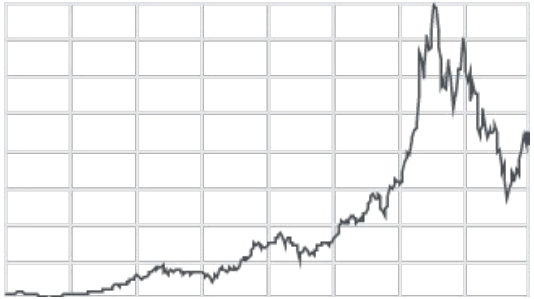
Application - Authentication



Other: zero-knowledge proof

Application - Cryptocurrency

- Block chain
  - decentralization -> P2P
  - Record storage and verification are spread across the network



ethereum

bitcoin

Application(?) - Cryptoworm

Classical Cipher

- Shift cipher
  - Caesar cipher `DEF -> ABC`
  - ROT13 `DEF -> QRS`

= 26

- Substitution cipher
  - `DEF -> SWF`
  - `DEF -> PAW`

= 26!

Classical Cipher

- Vigenere cipher (a + b mod 26)
  - Plaintext: `ATTACKATDAWN`
  - Key: `LEMONLEMONLE` (`LEMON`)
  - Ciphertext: `LXFOPVEFRNHR`

- Transposition cipher
  - `ABCDEF` -> `FEDCBA`
  - `ABCDEF` -> `ABC`
`FED`

Hashing

- OI
  - e.g.: rolling hash
    - will be taught in `String Algorithms`
  - memory, time <-> a small probability of WA
- General
  - for verification
  - a (hopefully) injective function
    - without collision (`f(a) = f(b)` but `a != b`)
  - easy to compute the value
  - difficult to compute the inverse value
  - usually not “continuous”
    - “`a` and `b` are close” does not implies “`f(a)` and `f(b)` are close”

MD5

- A hashing function
- Software checksum
- No longer safe
- Other:
  - SHA-2


如果新功能没出问题的话，这场成绩将计入rating。

参加本次比赛将有机会获得 UOJ 抱枕！5d1ee77c1c495a6303e781cbe002b61b 是获奖条件的 md5 码。比赛结

再次提醒大家比赛中途只测样例，最后会进行最终测试，把所有提交记录重测。

（其实感觉这次难度确实好水好水，大家捉得开心就好）

UPD： 比赛已经结束！

```
echo -n 比赛中最后一个提问的 | md5sum
5d1ee77c1c495a6303e781cbe002b61b
```

恭喜获得前 5 名的选手！

1. **Rui**

Hashing vs Encryption

- Hashing
  - one-way

- Encryption
  - two-way
  - reverse: decryption

Symmetric / Asymmetric Key Encryption

- Symmetric Key Encryption
  - same keys for encryption and decryption
  - e.g. xor
    - 123 xor 456 = 435
    - 435 xor 456 = 123
  - e.g. Advanced Encryption Standard (AES)

- Asymmetric Key Encryption
  - different keys for encryption and decryption
    - one public
    - one private
    - two keys are paired, i.e. they cannot be generated independently

RSA

- **R**ivest–**S**hamir–**A**dleman

$n = pq$ where p and q are primes

- R.H.S. -> L.H.S. is fast (multiplication)
  - e.g. FFT
- L.H.S. -> R.H.S. is slow (factorization)

RSA

- Euler's phi function φ
  - will be taught in `Mathematics in OI (II)`
  - $\varphi(pq) = (p - 1)(q - 1)$
  - for fixed a, $\varphi(n)$ is the length of a cycle of $a^m \bmod n$ (not necessary minimum)
- Extended Euclidean algorithm
  - will be taught in `Mathematics in OI (I)`
  - find G.C.D.
  - find modular inverse
- Fast exponential algorithm
  - a.k.a. big mod algorithm
  - taught in `Recursion, Divide and Conquer`

RSA

- Preparation
  - find $n = pq$
  - find $de = 1 \pmod{\varphi(n)}$ with $\gcd(e, \varphi(n)) = 1$
  - make the public key (n, e) public

RSA

- Encryption
  - have plaintext M in mind
    - assume M to be an integer
  - get public key (n, e)
  - calculate $E = M^e \pmod n$
  - send ciphertext E

RSA

- Decryption
  - receive ciphertext E
  - calculate $E^d = (M^e)^d = M \pmod n$
    - recall that
      - $de = 1 \pmod{\varphi(n)}$
      - $\varphi(n)$ is the length of a cycle of $a^m \bmod n$
  - retrieve plaintext M

RSA

- Possible attack
  - already know ciphertext E and public key (n, e)
  - want to know plaintext M
    - $E^d = (M^e)^d = M \pmod{n}$
  - require private key d
    - $de = 1 \pmod{\varphi(n)}$
  - require $\varphi(n)$
    - $\varphi(n) = \varphi(pq) = (p - 1)(q - 1)$
  - require p and q
  - need factorization

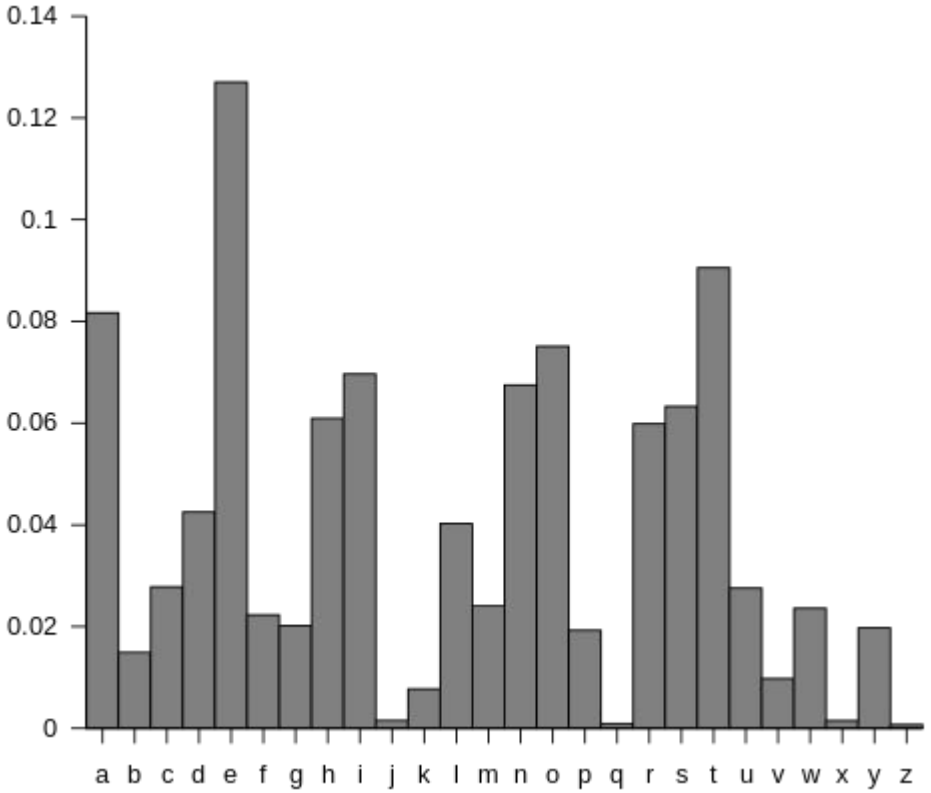
Attack

- Brute-force
  - exhaustion (of keys)
  - look for meaningful outcomes
  - computer performance is increasing incredibly

- Rainbow table
  - store all `(x, f(x))`
  - query time ↓
  - memory required ↑

Attack

- Frequency analysis
  - many `x` in ciphertext
  - ->
  - `e` is possibly replaced by `x` during encryption



Attack

- Quantum computing
  - Shor algorithm
    - polynomial time factorization algorithm
      - (in terms of number of bits)
    - RSA is becoming unsafer

Relation with OI

- Two-step tasks
  - I1123 Parrots
  - M1743 Tree Recovery II
  - S141 Dividing the Cities
  - T144 Lost Sequence
- Huffman coding
  - N1521 荷馬史詩

✓	✘
00	00
010	001
011	011
1	1

- Purpose (data compression) sometimes differs from cryptography

IOI01 Double Crypt

$$\texttt{ciphertext} = E(\texttt{plaintext}, \texttt{key})$$

$$\texttt{plaintext} = E^{-1}(\texttt{ciphertext}, \texttt{key})$$

E and E^{-1} are given functions (can be called directly).

You may assume their time complexities are O(1).

Input: `plaintext`, `E(E(plaintext, key1), key2)`

Output: `key1`, `key2`

All items are in 128-bit.

Furthermore, the last 108 bits of the keys are 0.

IOI01 Double Crypt

- Exhaustion of keys
  - 2^{40} combinations
  - 2^{40} = 1099511627776

IOI01 Double Crypt

- Meet in the middle

$$x = \texttt{plaintext}$$

$$y = E(E(\texttt{plaintext}, \texttt{key1}), \texttt{key2})$$

$$A_{\texttt{key1}} = E(x, \texttt{key1})$$

$$B_{\texttt{key2}} = E^{-1}(y, \texttt{key2})$$

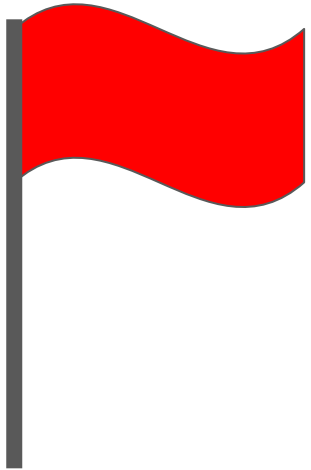
  - exhaust all A and B
  - find the matched pair

CTF

- **C**apture **T**he **F**lag

- Computer security competition
  - cryptography
  - reverse engineering
  - pwn
  - etc.

- Allowed to use online resources



For Fun

- Early April Fools?
- Answer: [A-Z]*
- Answers are somehow meaningful

1. SHOFJEYIWEETRKJSQUIQHIK
2. 181324542
3. HTEOARRLTIAIZNCODANVL