Computer Science Basics
Symmetric Cryptography

Fall Term 2023/2024

Emmanuel Benoist | BFH-TI

Symmetric Cryptography

- Last Week
- Problem statement
- Caesar Cypher
- One time pad
- Advanced Encryption Standard
- Conclusion
- Exercise

Last Week

Lossy Compression

Loseless vs Lossy compression

- Loseless for programs, documents that can not be changed
- Lossy for multimedia content, where losing a small part of the information is not a problem

Image compression, JPEG

- Y'CbCr image, Downsampling of Cb Cr (because Y' is more important for the eye).
- Discrete Cosine Transform (DCT) + Quantization = most of the elements are zero
- Lose quality : impossible to reconstruct the first image.
- Compression factor is around 10:1

Problem statement

Problem

Alice and Bob want to communicate

- Their conversation must remain secret
- No one should be able to listen to the conversation
- Alice wants that only Bob can read the message
- Bob wants to be certain that the message comes from Alice

Protocol

- A and B define together a common key K
- A writes a message M
- A encrypts M with the key K
- the encrypted message MK' travels on the internet
- B reads MK' and uses K to decrypt

Results

- A is sure that only B can read her message (if the key remains secret).
- B is not sure if the message comes from A

Check Integrity

Addition to the protocol

- A must do a “checksum” of his message (cryptographic hash function) and put it in his message.
- If the decrypted message is consistent with the checksum, it comes from A
- (or someone who has stolen the key)

Caesar Cypher

Caesar Cypher

Caesar's Cypher

- Used by Cesar for its secret messages

Substitution cipher

- A letter is replaced by a letter at a fixed distance
- For example: we decide that the distance (i.e. the key) is three letters to the right
- A becomes D,
- B becomes E,
- It becomes F,
- Bonjour becomes Erqmrxu

Decryption

- As the correspondent knows the key (shift 3 to the right) they can decypt the message
- Erqmrxu becomes Bonjour again

Existing protocols

Examples of protocols

- One time pad
- DES (Data Encryption Standard (or 3DES) considered too weak
- AES (Advanced Encryption Standard),
- Blowfish, $CAST_5$, RC_4, RC_5, RC6, and IDEA

What is symmetric encryption used for

It is faster than asymmetric encryption (see next week)

- Faster Encryption
- Used for bulk Encryption / Encrypting large amounts of data

Examples

- TLS uses symmetric encryption to send data for web pages (https)
- All the algorithms used in real life use symmetric algorithms for encryption of data.

One time pad

One time pad

The only one algorithm to be proven secure

- Proof by Shannon in the 1940s (publication 1949)

Encryption

- The key K used to encrypt a message M has the same length than M
  length(K) = length(M)
- The key is used only once (need a new key with each new message)
- To encrypt the message one to an XOR
  M' = M XOR K

Decryption

- Two times XOR is returning to the original message
  M = M' XOR K

One time pad

Very secure

- bits in M' are totally random (if K is random)
- knowing M' and not K, ANY message can be obtained with a different key K'.
- Knowing M' and the message you think is good, you can always generate K'.

Must be used only once

- If used twice, the messages can be used to find the key.

Advanced Encryption Standard

AES - Advanced Encryption Standard

Successor of DES (Data Encryption Standard)

- DES was proven week, (key length 56bit)
- NIST (National Institute of Standards and Technology) organized a contest
- Algorithm Rijndael won

Block cypher

- Data is split in Blocks of 128 bits

Supports various key lengths

- 128, 192, 256 bit

AES

Advantages

- AES is very fast
- Can be software implemented
- Can be hardware implemented

Quite secure

- There is no known attacks for a correct implementation of AES

Conclusion

Conclusion

Problem

- Alice and Bob must share the same key
- Alice and Bob must know each other
- How to do it on the Internet?
- I don’t know Google or Facebook

Exercise

Exercise

Install Open SSL in Linux

```
apt install openssl
```

Create a file (and write a secret message)

```
emacs file.txt
```

Encrypt a file using OpenSSL (You must define a password with a friend before)

```
openssl enc -aes-256-ecb -salt -pbkdf2 -in file.txt -out file.txt.enc
```

Send this message to your friend
Decrypt the message of your friends

```
openssl enc -aes-256-ecb -d -in file.txt.enc -out file.txt
```

More Information `https://www.shellhacks.com/encrypt-decrypt-file-password-openssl/`

References

Web Sites

- https://en.wikipedia.org/wiki/One-time_pad
- https://en.wikipedia.org/wiki/Advanced_Encryption_Standard