Contents

1 Crytography

2 Encryption

- Encryption: We encode a message such that only authorised users may read it
- Cipher: takes a string of plaintext, and converts it into a string of ciphertext
- Encryption can provide: Confidentiality, Integrity, Authenticity

3 Notation

- A *cipher* converts a plaintext message **M** into a *ciphertext* **C** under the control of a key **K**
- C is **not a secret**, but without knowledge of the key, it should be impossible to **reconstruct** M
- Comes in two forms: *Symmetric* same key for encryption / decryption. *Asymmetric* separate keys

4 The One Time Pad

Using XOR to encrypt the message:

- Use a key thats the **same length** as the message
- XOR each message bit with each key bit
- Any possible plaintext can be recovered depending on the key
- This is an example where $M = C - K mod 26$

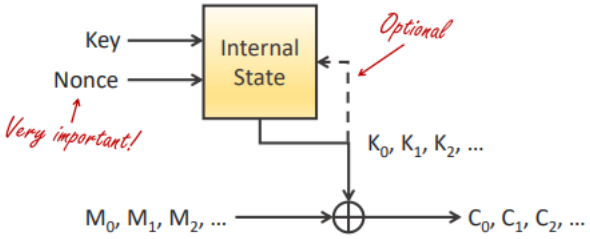
C	MXFLCRDH	MXFLCRDH
K	emrqytpn	eqfsytpn
M	iloveyou	ihateyou

It is an impractical method because:

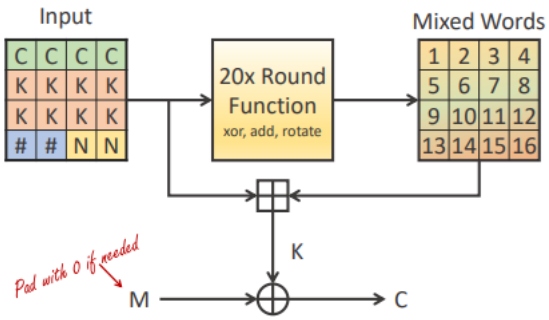
- A 1GB file would need a 1GB key!
- How are we transporting these keys? Or storing them?
- If you ever reuse a key, the entire cipher is broken

5 Modern Stream Ciphers

- Modern stream ciphers use an initial seed key to generate an infinite pseudorandom keystream
- The message and keystream are usually combined using XOR - - which is reversible if applied twice
- Common to seed an initial state using a key, then update this state for as long as needed



5.1 ChaCha20



- ChaCha performs 20 rounds: alternates Column and Diagonal Rounds, each round is 4 quarter rounds
- It's fast, because ChaCha20 is based on ARX (Addition-Rotation-XOR) which are **CPU friendly instruction**

5.2 Advantages of Stream Ciphers

- Encrypting long continuous streams, possibly of unknown length
- E.g. GSM mobile communications
- Extremely **fast with a low memory footprint**, ideal for low-power devices
- If designed well, can seek to any location in the steam. E.g. Streaming video with DRM

5.3 Vulnerabilities

- Stream ciphers give us confidentiality, but not integrity
- We must include another mechanism

6 Hash functions

- A hash function is any function that can be used to map data of arbitrary size to data of a **fixed size**
- Hash functions are used everywhere. Message authentication, integrity, passwords etc
- Usually hash functions iteratively jumble blocks of a message after another
- Once all the message is gone, we read the current hash.
- The initial hash is usually defined in the spec

6.1 Strong Hash Functions

- The output must be indistinguishable from random noise
- Bit changes must diffused through the entire output. A small change to a message should change the hash value so extensively that the new hash value appears uncorrelated with the old hash value
- For a hash function to be useful, we need it to have some important properties:
  - One-way: Given h(M), we cant calculate M easily
  - Weak collision resistance: Given M and h(M), we cannot find some M s.t. h(M) = h(M)
  - Strong collision resistance: We cant find some message pair M M s.t. h(M) = h(M)

6.2 The Birthday Attack

- **The Birthday Paradox**: concerns the probability that, in a set of n randomly chosen people, some pair of them will have the same birthday
- The output of the hash must be long enough to avoid a birthday attack.
- Given a hash function outputs n bit hashes, You will find a collision after about $2^{n/2}$ random attempts

Name	Output Length	Rounds	Security
MD5	128-bit	4	Broken
SHA-1	160	80	Theoretically vulnerable, not trusted
SHA-2	224, 256, 384, 512	64, 80	Some theories, currently considered safe
SHA-3 (Keccak)	224, 256, 384, 512 SHAKE128, 256	24	Secure, but relatively untested, strength variable

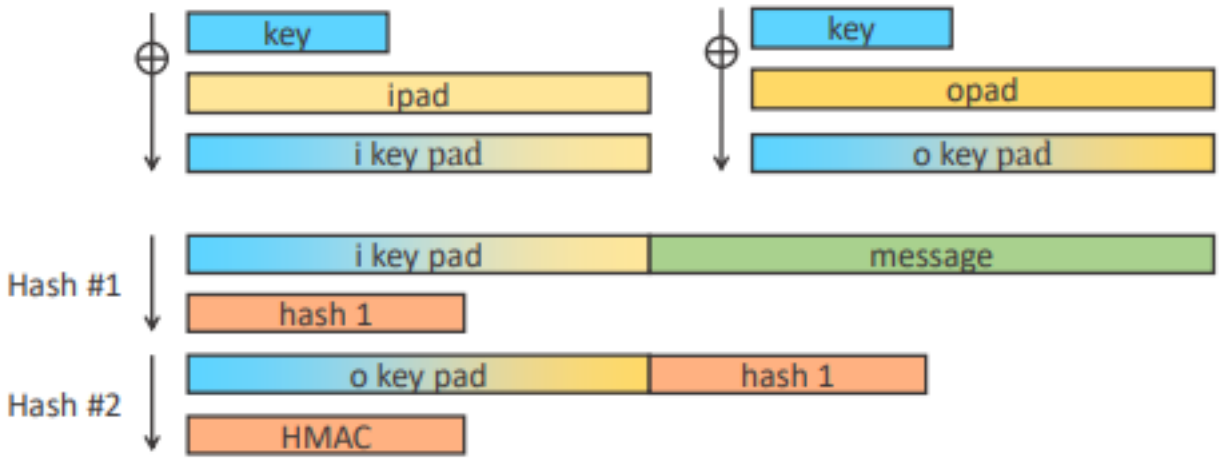
7 Message Authentication Codes

- Provide integrity and authenticity, not confidentiality: protecting system files, ensuring messages havent been altered
- Calculate a **keyed hash** of the message, then append this to the end of the message

7.1 HMAC

Double hashing in HMAC handles some weaknesses with traditional MACs.

$$HMAC(k, M) = H\left((k \oplus opad) \parallel H\left((k \oplus ipad) \parallel M\right)\right)$$



Reference section

placeholder