$K_B^+(m)$

encrypt m

$K_B^-\left(K_B^+(m)\right) = m = K_B^+\left(K_B^-(m)\right)$

decrypt

not encryption

Digital Signatures

integrity - message has not changed

non-repudiation - the signer is known

hash: Data: d

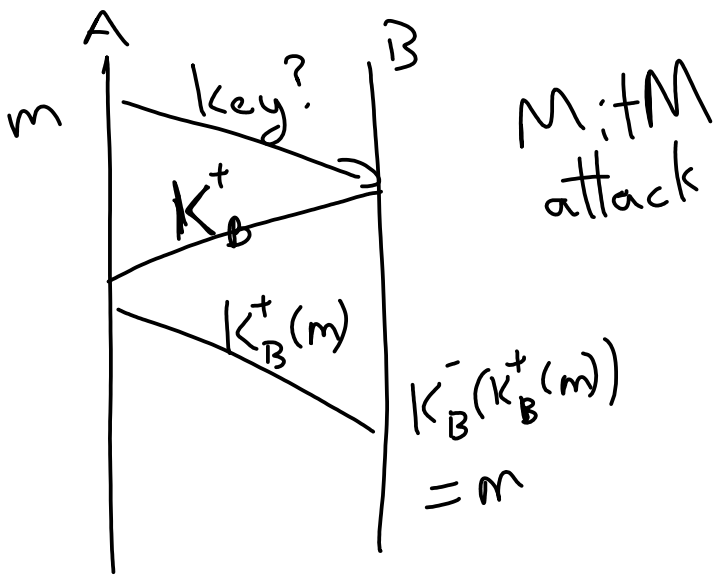
Hash: h(d)

$\rightarrow h(d) \longrightarrow$ sign hash $K_B^-(h(d))$

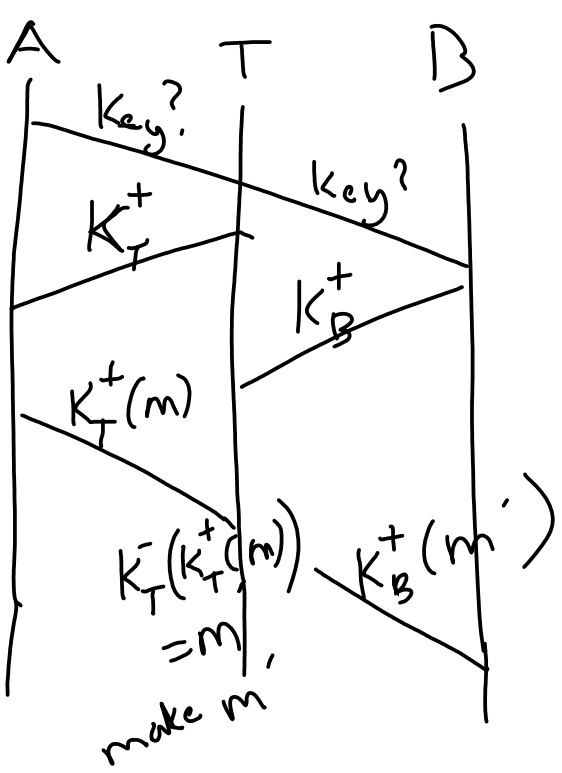
verify

$d \rightarrow h(d)$

$K_B^+\left(K_B^-(h(d))\right) \stackrel{?}{=} h(d)$



MitM attack



ARP spoofing
session hijacking

Public Key Infrastructure (PKI)

Certificates: device/domain, public key

Certificate Authority (CA)

chain of trust: trust root CA who trusts another CA who trusts the server

Server's certificate signed by CA

Validation

Domain Validation: requester controls the domain

Organization Validation
vetted by CA

Extended Validation
background check by CA

---

ssh:

phase 1:
- client makes connection
- server sends public host key
- client verifies
- Diffie-Hellman key exchange (session key)

Phase 2: User authentication
Server: ask for password
Client: send password

OR Key pair authentication
1. client sends an ID
2. server checks for the ID
3. server generates NONCE, encrypts w/ user's public key
4. server sends encrypted NONCE
5. client decrypts NONCE w/ private key

6. client combines NONCE w/ session key, uses MD5 to hash result.
7. client sends hash to server
8. server compares hash to its calculated hash