DIGITAL SIGNATURE

INTRODUCTION, CLASSIFICATION, DELEGATION

M.SC. BARTŁOMIEJ SŁOTA

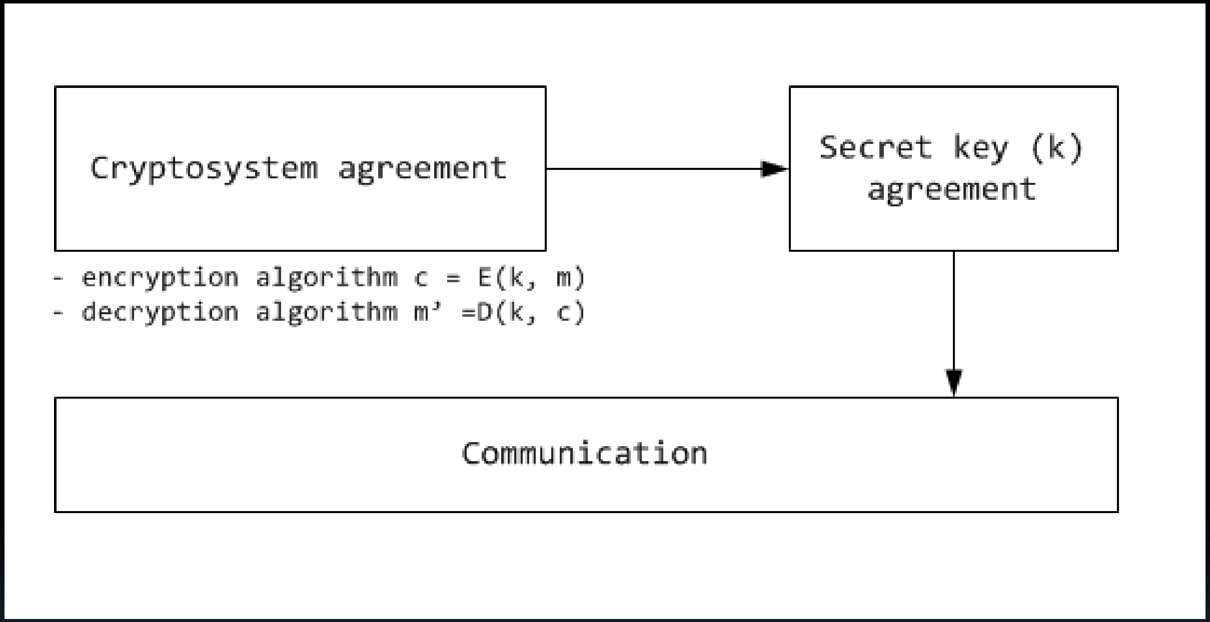
ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Conventional cryptography



Conventional cryptography

- Key must be transmitted by means of a secure channel (courier/meeting)
- If compromised – key may be misused (decryption of real messages, encryption of false messages, etc.)
- There’s no way to conclude from the ciphertext who was the sender (Bob can send message to himself) – source of forgery
- Key management – n(n-1)/2

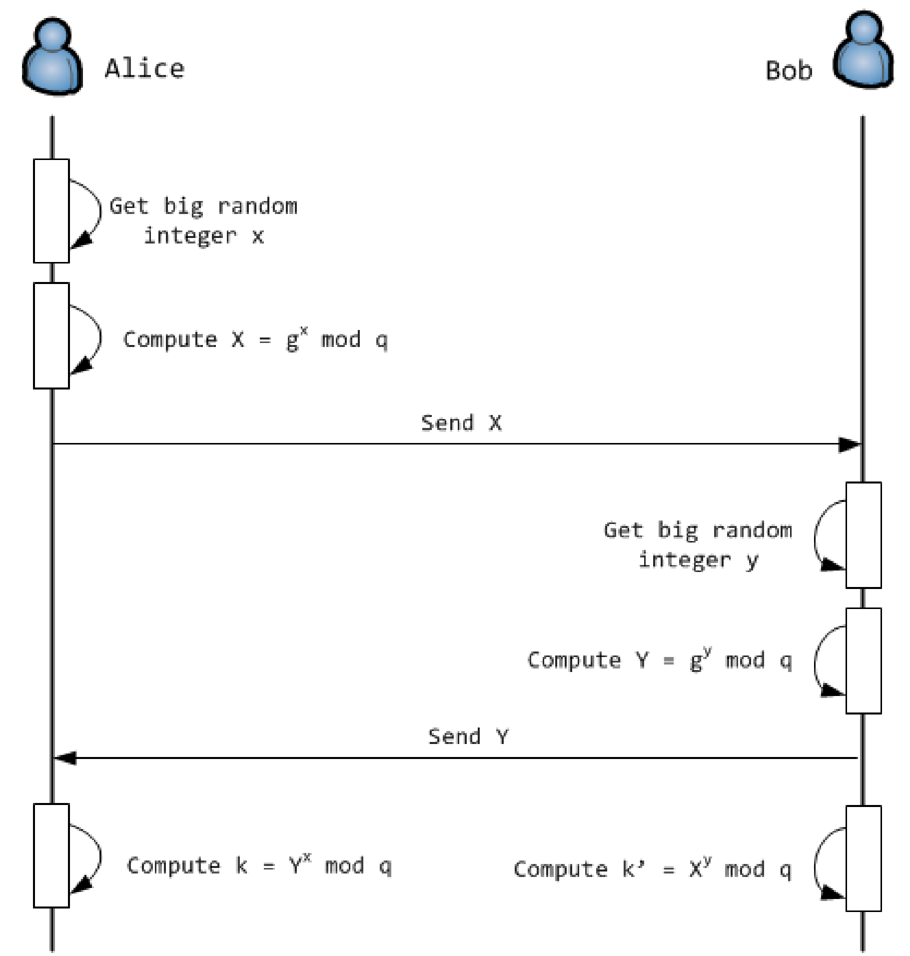
ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Diffie-Hellman concept

- Exponential key agreement protocol
- First known public key algorithm
- Cryptosystem
  - q – power of a prime number, defines the order of the finite field F_q
  - g – generator of the multiplicative group of order q-1
- Security based on discrele logarithm problem

Diffie-Hellman concept



$$k = Y^x \bmod q = g^{xy} \bmod q = X^y \bmod q = k'$$

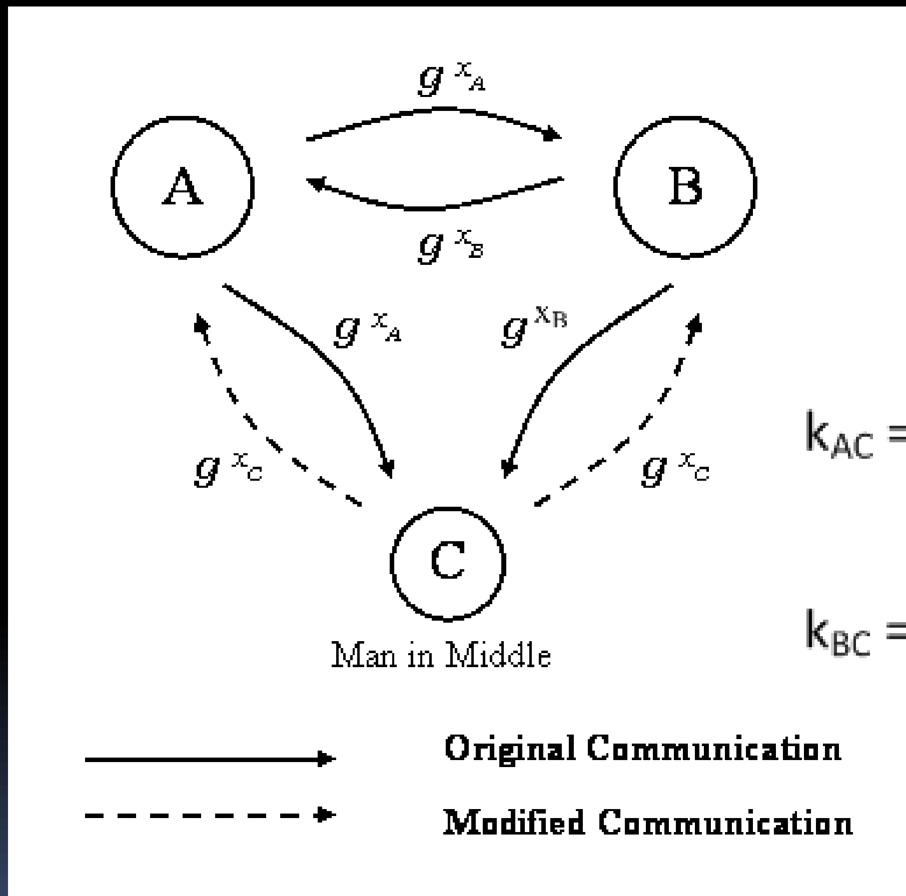
ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Man-in-the-middle attack

- DH cannot stand against MIM attack
- Intruder Mallory may interrupt the communication during key exchange
- Cryptosystem = DH cryptosystem
  - p, q, F_q

Man-in-the-middle attack



$$k_{AC} = g^{Xa*Xc} \bmod q$$

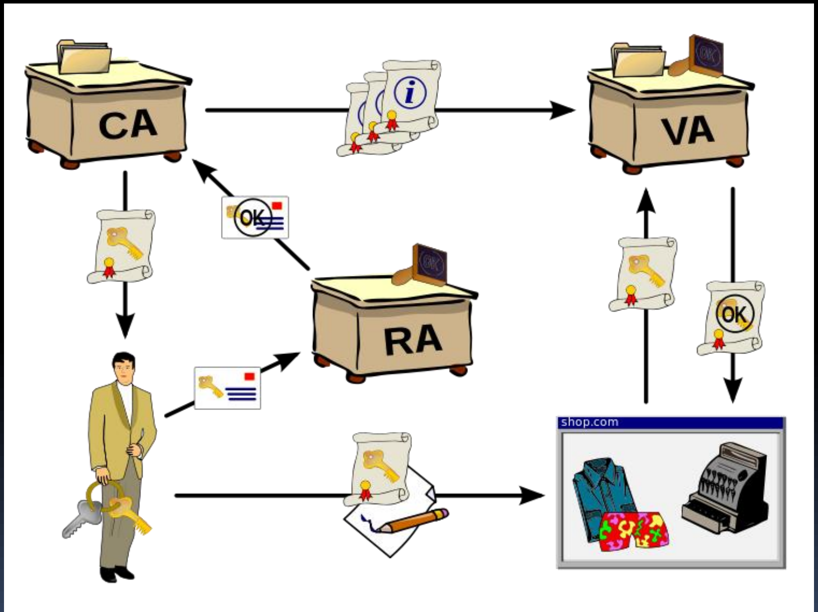
$$k_{BC} = g^{Xb*Xc} \bmod q$$

ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Public key cryptography

- Success of the MIM attack – sides cannot be sure whose key they are using
- Remedy?
  - Trusted authority
  - Digital certificate



ABSTRACT

- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Hash functions

- Mapping any length messages to fixed length message digests
- h = H(M)
  - h – message digest
  - M – message of any length
- Features
  - Having M – it is easy to compute h = H(M)
  - One way property
  - Collision resistance property

ABSTRACT

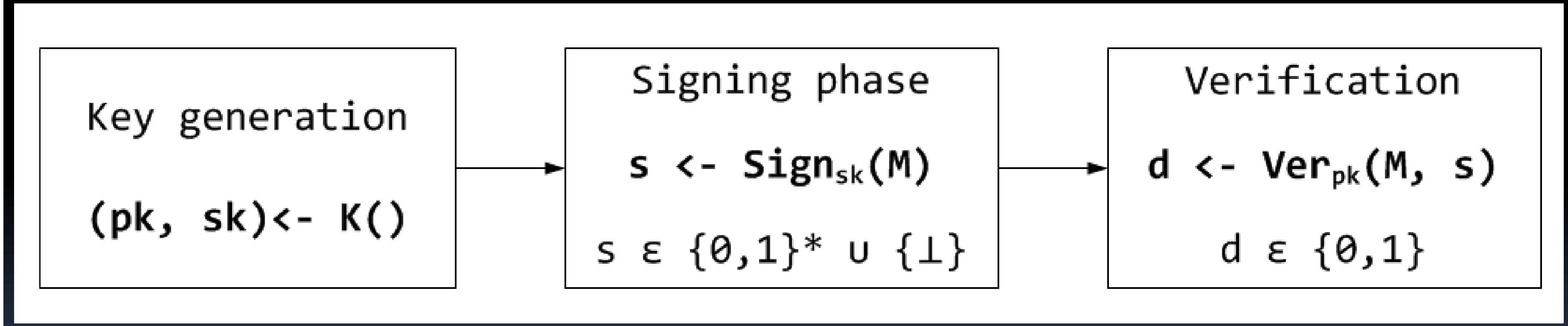
- Conventional cryptography
- Diffie – Hellman concept
- Man-in-the-middle attack
- Public key cryptography
- Hash functions
- Digital signature
  - Basic schemes
  - Classification
  - Delegated signature schemes

Digital signature

- Electronic analogue to hand written signature
- Provided security services:
  - Authentication
  - Non-repudation
  - Integrity
- Messages with signature can be encrypted and time-stamped

Digital signature

- Digital signature scheme:
  - The base of every signature algorithm
  - Consists of three phases



Digital signature - RSA

- Rivest, Shamir, Adleman
- Security based on the IF problem

Key generation

- generate two large distinct random primes p and q
- compute $n = p*q$ and Euler function $\varphi(n) = (p-1)(q-1)$
- select a random integer e, so that $1 < e < \varphi(n)$, such that $gcd(e, fi(n)) = 1$
- compute d, so that $1 < d < \varphi(n)$, such that $ed \equiv 1 \mod \varphi(n)$. In other words, $d = e^{-1} \mod \varphi(n)$ is the inverse of e.
- (e, n) - public key
- (d, n) - private key
- message space M and ciphertext space C is $Z_n = \{0, 1, 2, ..., n-1\}$

Digital signature - RSA

Signature generation

- uses private key
- compute signature $s = m^d \bmod n$

Signature verification

- uses signer's public key (e, n)
- compute $m' = s^e \bmod n = m^{ed} \bmod n$
- verify if $m' = m$

Digital signature - RSA

Proof

- let's remind the Euler theorem:
  - if a and n are relatively prime, then $a^{\varphi(n)} = 1 \bmod n$
- moreover: $ed \equiv 1 \bmod \varphi(n) \Rightarrow ed = 1 + x * \varphi(n)$
- so: $m^{ed} = m * m^{x*\varphi(n)} = m * 1^{x} = m \bmod n$

Digital signature - ElGamal

- Security based on the DL problem

Key generation

- Take a large prime number p defining a finite field Zp
- Find g - generator of a multiplicative group Z*p
- compute random secret number x, $1 < x < p$
- compute $y = g^x \mod p$
- {p, g, y} - public key
- x - private key

Digital signature - ElGamal

Signature generation

- uses private key x, and cryptosystem parameters (p, g)
- select a random integer k, $1 < k < p-1$ such that gcd(k, p-1) = 1
- compute r =gk mod p and k-1 mod (p-1)
- compute $s = k^{-1} [m - x*r] \bmod (p-1)$
- (r, s, m) - signature

Signature verification

- uses public key {p, g, y}
- check if $1 < r < (p-1)$
- compute $v_1 = y^r * r^s \bmod p$
- compute $v_2 = g^m \bmod p$
- the signature is valid if and only if $v_1 = v_2$

Digital signature - ElGamal

Proof

$$v_1 = y^r * r^s \bmod p = g^{x*r} * r^{k^{-1}*[m-x*r]} \bmod p$$

$$= g^{x*r} * g^{k*k^{-1}*[m-x*r]} \bmod p = g^m \bmod p = v_2$$

Signature classification

1) By mathematical problem on which their security is based
   - Based on IF problem
   - Based on DL problem

2) By the signer identification method
   - Certificate – based signatures
   - ID-based signatures

Signature classification

3) By the usage of randomization
   - randomized schemes
   - deterministic schemes

4) By the ability to recover message
   - Schemes with appendix
   - TMR (Total Message Recovery)
   - PMR (Partial Message Recovery)

Signature classification

- Signature schemes may belong to many of presented groups
- Most of them have special features – another way of classification
- They still use RSA or ElGamal signature scheme

Blind signatures

- Introduced by D.Chaum
- The signer knows neither the message nor its signature
- Alice generates blinds message m using blinding function m' = f(m) and sends to BOB
- Bob signs s' = Sign(m') and sends the result back to Alice
- Alice computes the reverse blinding function f'(s') = s and gets the signature
- Usage: e-money, e-voting

Undeniable signatures

- Proposed by David Chaum and Hans van Antwerpen
- Digital signatures can be copied exactly
- Verification is possible only with the interaction with the signer
  - only authorized entities can access the document to verify the signature
  - signer cannot deny a valid signature (disavowal protocol)

Designated Confirmer Signatures

- Compromise between self-authenticating signatures and undeniable signatures
- A designated confirmer allows certain designated parties to confirm the authenticity at any time without asking signer
- Others are not able to verify signature without the aid of designated parties or the signer himself

Directed Signatures

- Proposed by Lim and Lee
- Self-authentication property is not suitable for applications like signing personal information (tax bills, prescriptions, etc.)
- Signer sends signed message m to the designated verifier (i.e. patient) while others know nothing on the origin and validity of the message without help of signer or the designated verifier
- Both signer and designated verifier can prove to any third party that the signature is valid

Nominative Signatures

- Includes two parties:
  - nominator - generates a digital signature
  - nominee - verifies the validity
- Only nominee can verify the nominator's signature
- Only nominee can prove to some third party that the signature is issued to him and is valid

Group Signatures

- Assume we have a group of users
  - every member is authorized to sign documents on behalf of the group
  - The signature generated by any member is called a group signature
- The receiver of the signature:
  - can verify if it represents the particular group
  - cannot identify which member signed it
- Group members or TA can identify the signer

Ring signatures

- User from the set:
  - can convince the verifier that the signer belongs to the set
  - cannot identify signer
- Unlike group signature, requires additional setup:
  - group manager
  - setup procedure
  - action of the non-signing members
- For signing purposes, signer may choose random set of other possible signers including himself

Threshold signatures

- (t, n) threshold scheme
  - a secret key k is shared among n members of a group
  - any t members are able to cooperate and reconstruct the key k
- (t-1) or less users cannot reveal nothing about the key
- But any set of t or more shareholders can impersonate any other set.
- Malicious set of signers does not have any responsibility for the signatures

Multi Signatures

- Used in applications that require require the signature of more than one person (e.g. bank account, government, etc.)
- Signing is possible only if multiple keys are available
- Each signer produces a valid partial signature on message which is combined further to get a complete signature

Proxy Signatures

- Delegated signature schemes
- Enables original signer to delegate signing authority to a proxy signer
  - temporal absence
  - lack of time or computational power
- Proxy signer can compute a signature, that can be verified with the original signer's public key

Full Delegation

- Alice gives her private key to Bob
- Bob using Alice’s private key computes signature
- such signature is indistinguishable from the normal signature

Partial Delegation

- Alice computes proxy key s from her private key x and gives it to the proxy signer Bob in a secure way
- Bob computes signature with the key s
- Such signature is distinguishable from the original signature
- For security reasons key x should not be computable from s

Delegation by Warrant

- Warrant - certificate composed of:
  - a message part (that the proxy signer is authorized to sign)
  - public key
- Delegate proxy
  - Alice signs a warrant and declares Peter as proxy signer
  - To sign message, Peter simply signs it and combines with the warrant
  - Warrant differentiates between Peter's normal signature and the proxy signature
- Bearer proxy
  - Alice computes proxy key pair, signs a warrant, and gives to Peter

Partial Delegation with Warrant

- Compromise between delegation by warrant and partial delegation
- Alice generates a secret s from her private key, and includes a warrant
- Alice sends the secret to Peter in a secure way

Threshold Delegation

- Designed for group oriented societies
- (t,n) threshold delegation
  - Alice distributes proxy signature key among n proxy signers
  - To generate a valid proxy signature we need at least t signatures (t <= n)

THANK YOU!