Today in Cryptography (5830)

ECC wrapup
Hybrid encryption
OpenPGP standard
TextSecure

Katz-Lindell Chapter 10.3 (Hybrid Encryption)
RFC 4880 (OpenPGP standard)

Elliptic curves

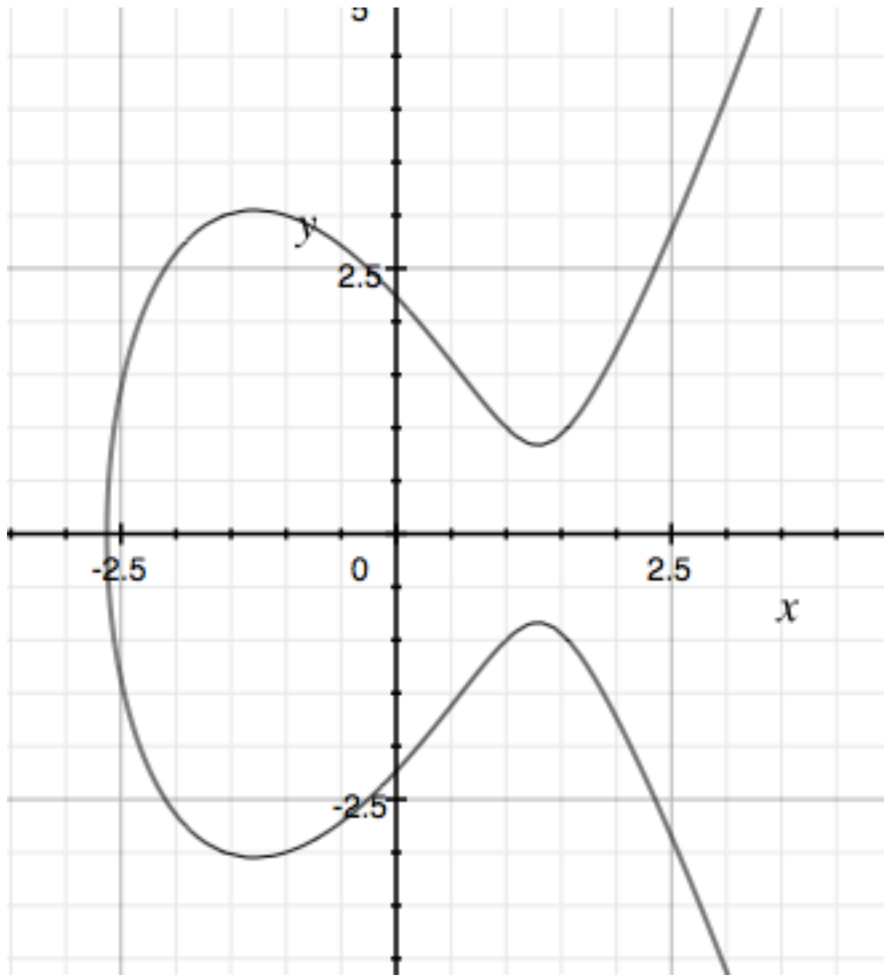
- An elliptic curve is set of x,y points in $\mathbf{F}_p$ defined by an equation

$$E = \{(x, y) \mid y^2 = x^3 + ax + b \bmod p\}$$

a,b are fixed values also from $\mathbf{F}_p$. Technical condition:

$$4a^3 + 27b^2 \neq 0$$

  – Plus one special point $\mathcal{O}$ called the "point at infinity"
- Defined group operation: point addition
  – Gives us a cyclic group
- NIST curves, Curve25519



$y^2 = x^3 - 5x + 5$ (over the reals)

Elliptic curve DH

X →

← Y

Pick random x from $\mathbf{Z}_q$
$X = xP$

Pick random y from $\mathbf{Z}_q$
$Y = yP$

$K = H(xY)$

$K = H(yX)$

"Additive notation" vs. "multiplicative notation"

X →

← Y

Pick random x from $\mathbf{Z}_{|G|}$
$X = g^x$

Pick random y from $\mathbf{Z}_{|G|}$
$Y = g^y$

$K = H(Y^x)$

$K = H(X^y)$

Elliptic curve DLP

- Given xP compute x
- Same as g^x compute x, just different group!

- Trivial algorithm requires time O(q), q size of ECC group
- Best known algorithm against well-chosen ECC group version runs in time $q^{0.5}$
  - Algorithm is ***generic:*** works against any cyclic group

Baby-Step Giant-Step algorithm

- ECDLP: Given xP for random x, compute x

```
Rewrite x as x = am + b  with m = ceil(q^0.5)
                 xP + (-am)P  =  bP
For b = 1, ..., m
     Store (b,bP)
For a = 1, ..., m
     Check if xP + (-amP) equals one of precomputed bP
     Return am + b
```

- Works in time $O(q^{0.5})$ and space $O(q^{0.5})$
- Pollard rho method: reduce space to constant

Baby-Step Giant-Step algorithm

- DLP: Given g^x for random x, compute x

Rewrite x as $x = az + b$ with $z = \text{ceil}(q^{0.5})$

$$g^x g^{-az} = g^b$$

For b = 1, ..., z

  Store (b, g^b)

For a = 1, ..., z

  Check if $g^x g^{-az}$ equals one of precomputed g^b values

  Return $az + b$

- Works in time $O(q^{0.5})$ and space $O(q^{0.5})$
- Pollard rho method: reduce space to constant

Comparison

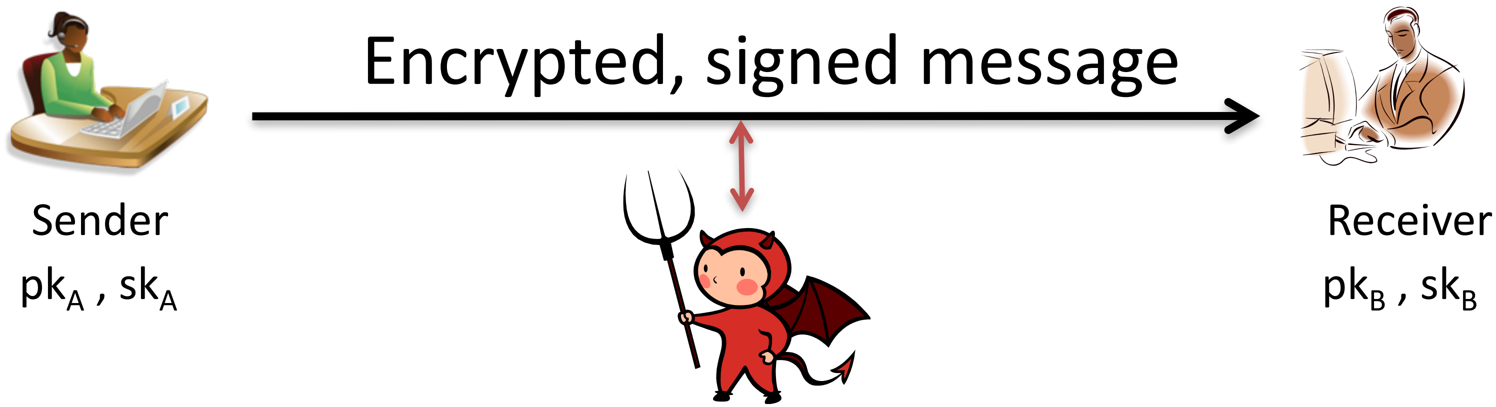
Security level	RSA size (log N)	DLP in finite field (log p)	DLP subgroup size (log q)	ECC group size (log q)
80	1024	1024	160	160
112	2048	2048	224	224
128	3072	3072	256	256
256	15360	15360	512	512

ECC has smallest representations and fastest performance of all asymptotic primitives we will see

Application-layer crypto

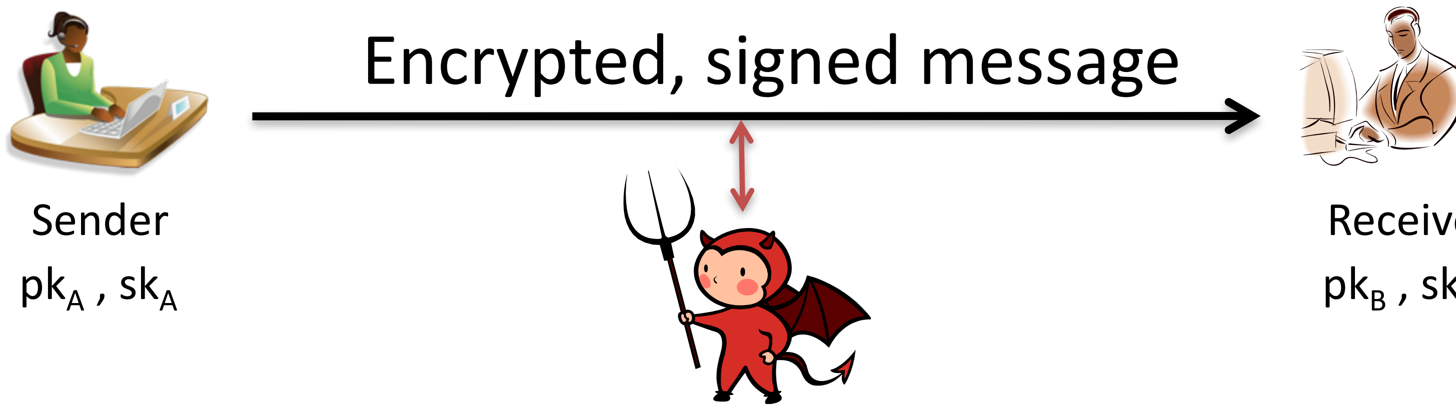
- So far focused on TLS as running example
  - Transport Layer Security
  - Provides network socket style stream interface
- What about if an application wants to encrypt discrete messages (as opposed to stream)?
  - Email
  - Text messages
  - Etc.

Email encryption



- Message may be large (body of email, PDF of attachments)
- Desire authenticity and confidentiality
- Public-keys delivered out-of-band
    - Websites, key parties, key directory servers

Email encryption



How should we design a solution?

Public-key encryption

Digital signatures

Symmetric authenticated encryption with associated data

ElGamal public-key encryption

g is generator for group of order p

Kg outputs pk = $(g, X = g^x)$ and sk = (g, x)

<u>Enc((g,X), M, R)</u>

r <-\$ $\mathbf{Z}_p$

$C1 = g^r$

$C2 = X^r * M$

Return C1 , C2

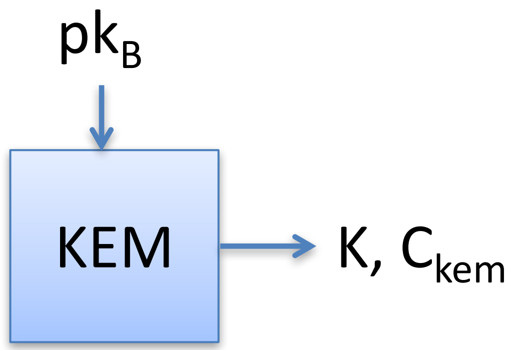
<u>Dec((g,x), C1, C2):</u>

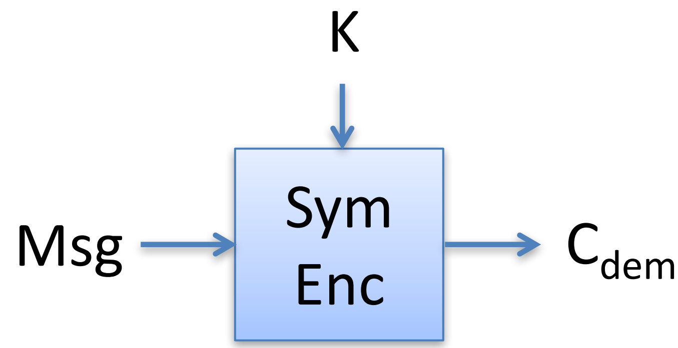
Return $C2 * C1^{-x}$

This is only at most chosen-plaintext attack secure. CCA attacks?

Only encrypts messages of size up to about log p bits

Hybrid encryption (KEM/DEM)





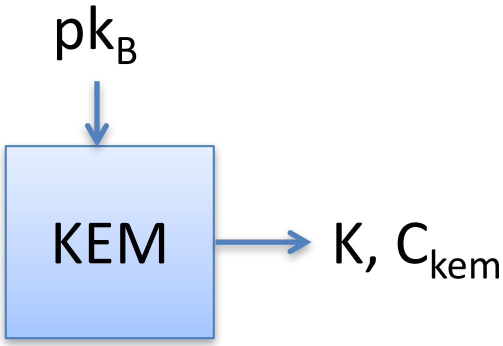
KEM = key encapsulation mechanism
Randomized public-key primitive

DEM = data encapsulation mechanism
One-time secure authenticated encryption

<u>HybEnc(pk, M)</u>
K, C_{kem} <-\$ KEM(pk)
C_{dem} <- Enc(K,M)
Return C_{kem} , C_{dem}

<u>HybDec(sk, C_{kem} , C_{dem})</u>
K <- KEM^{-1}(sk, C_{kem})
M <- Dec(K, C_{dem})
Return M

KEM from PKE



KEM = key encapsulation mechanism
Public-key primitive

<u>KEM(pk)</u>
Choose randomness R
C_{kem} <- PKE-Enc(pk,R)
Return H(R), C_{kem}

ElGamal KEM

Kg outputs pk = $(g, X = g^x)$ and sk = (g, x)
g is generator for group of order prime p

<u>EG-KEM((g,X), R)</u>
$r = R \bmod p$
$C_{kem} = g^r$
$K = X^r$
Return $H(K)$, C_{kem}

<u>Dec((g,x), C_{kem}):</u>

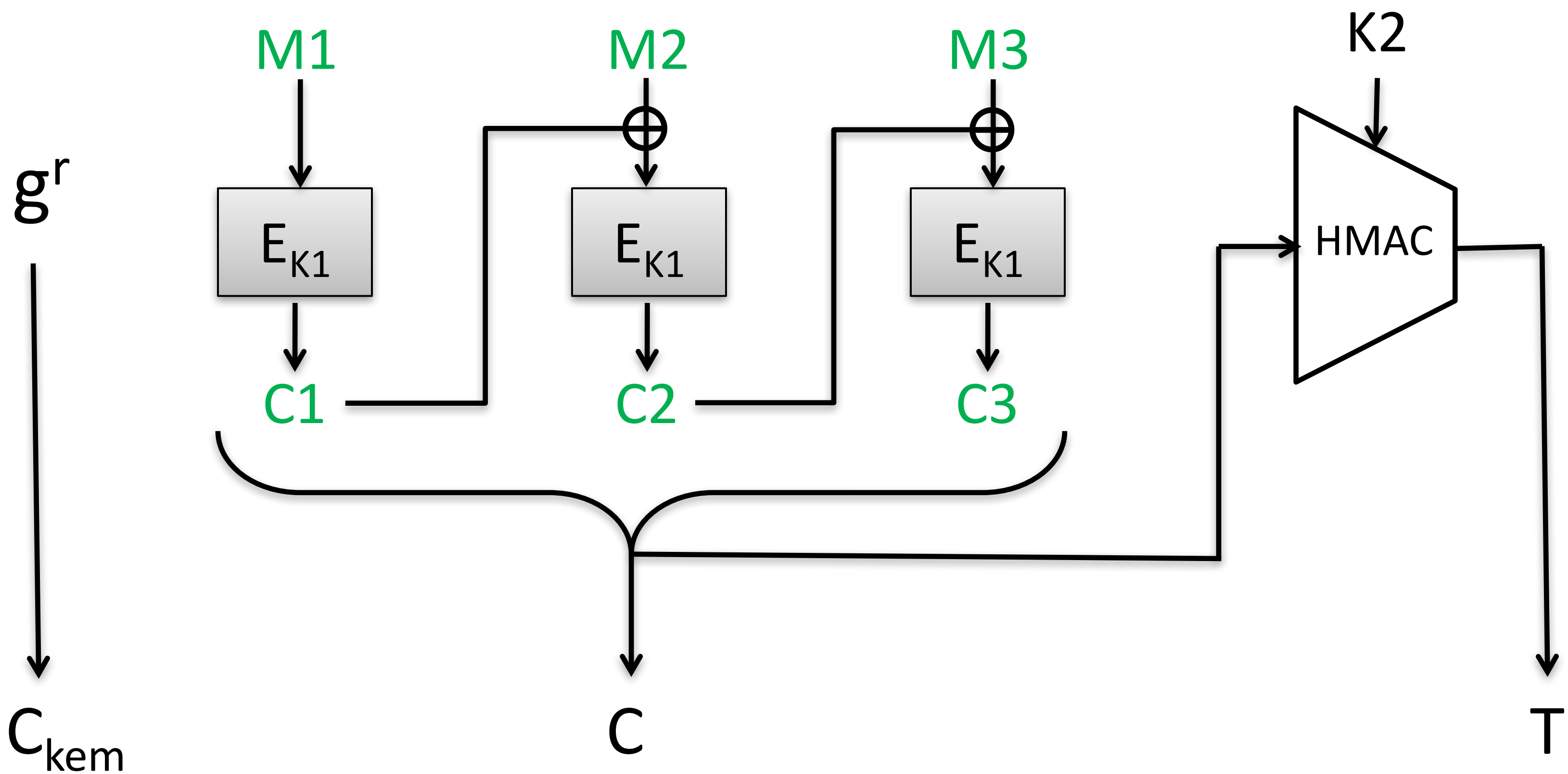
Return $H(C_{kem}^x)$

Secure if computational Diffie-Hellman assumption holds in group

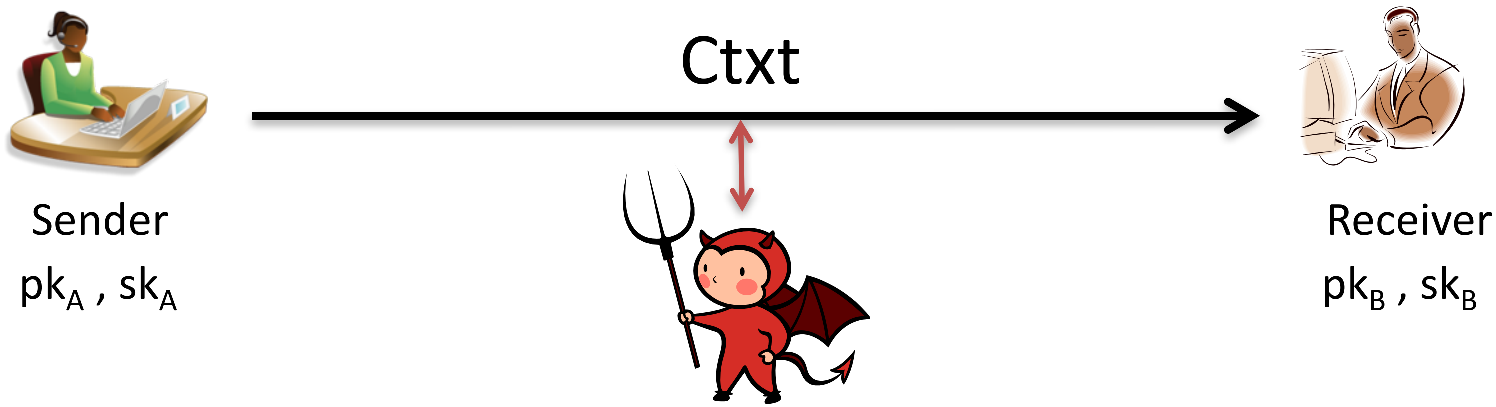
Example hybrid encryption

Enc(X,M):

$K1 \,||\, K2 = SHA256(g^{xr})$



Email encryption



- To digitally sign, let M = Msg || Sign(sk_A , Msg)
- Ctxt = Encrypt(pk_B , M)

PGP history

- Phil Zimmerman released "Pretty Good Privacy" in 1991 on a USENET post marked as "US only"

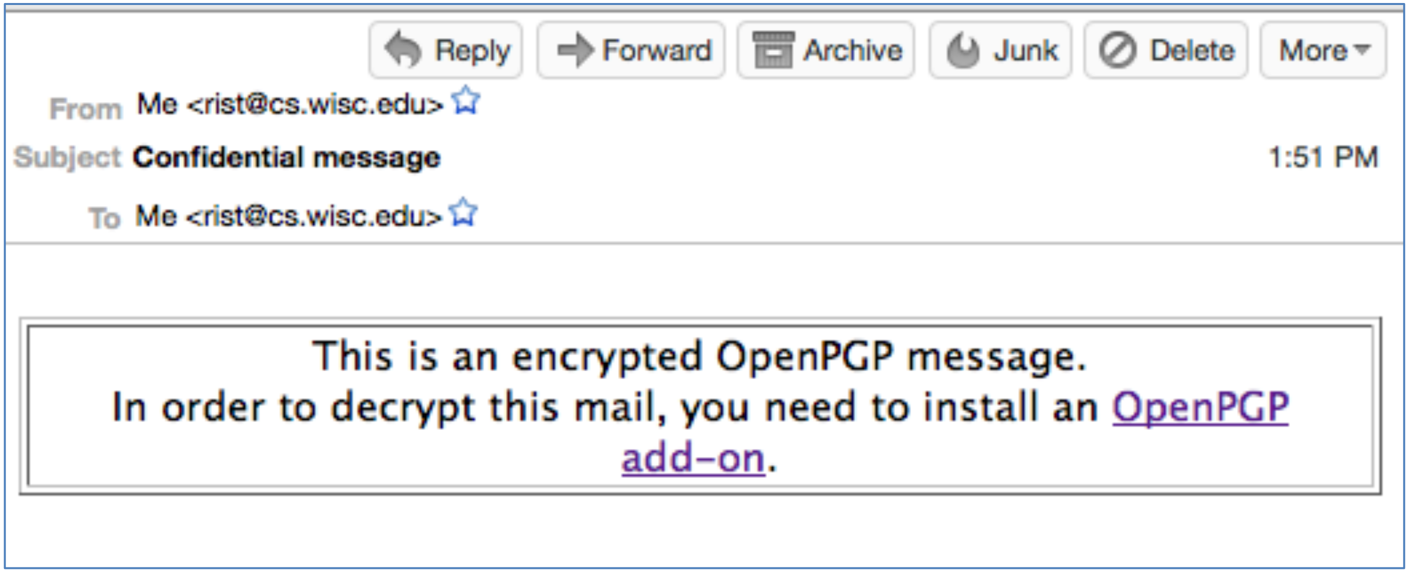
- 1993: Criminal investigation by US government for munitions export without a license.
  - Printed PGP source code into a book. First amendment gambit

OpenPGP overview

- Standard for PGP is RFC 4880
- Key encapsulation mechanism:
  - RSA PKCS#1 v1.5 encryption
  - ElGamal over finite field or elliptic curve
- Digital signatures:
  - RSA PKCS#1 v1.5 signatures
  - DSA
- Symmetric encryption:
  - Password-based key derivations using iterated hashing
  - CFB mode using block cipher (variant of CBC mode)

OpenPGP overview

- Security problems:
  - Padding oracle attacks against CFB & PKCS#1 v1.5
  - Attacks against home-brewed integrity checks (modification detection check, MDC)
  - Subject lines always in the clear
- Usability problems:
  - Users must manage their own keys
  - Copying private keys to each device
  - Checking validity of other recipient’s public key



The Switch

Yahoo's plan to get Mail users to encrypt their e-mail: Make it simple

A | 14 | Save for Later | Reading List

Messaging encryption

Encrypted/Signed SMS or IM

Sender

pk_A , sk_A

Receiver

pk_B , sk_B

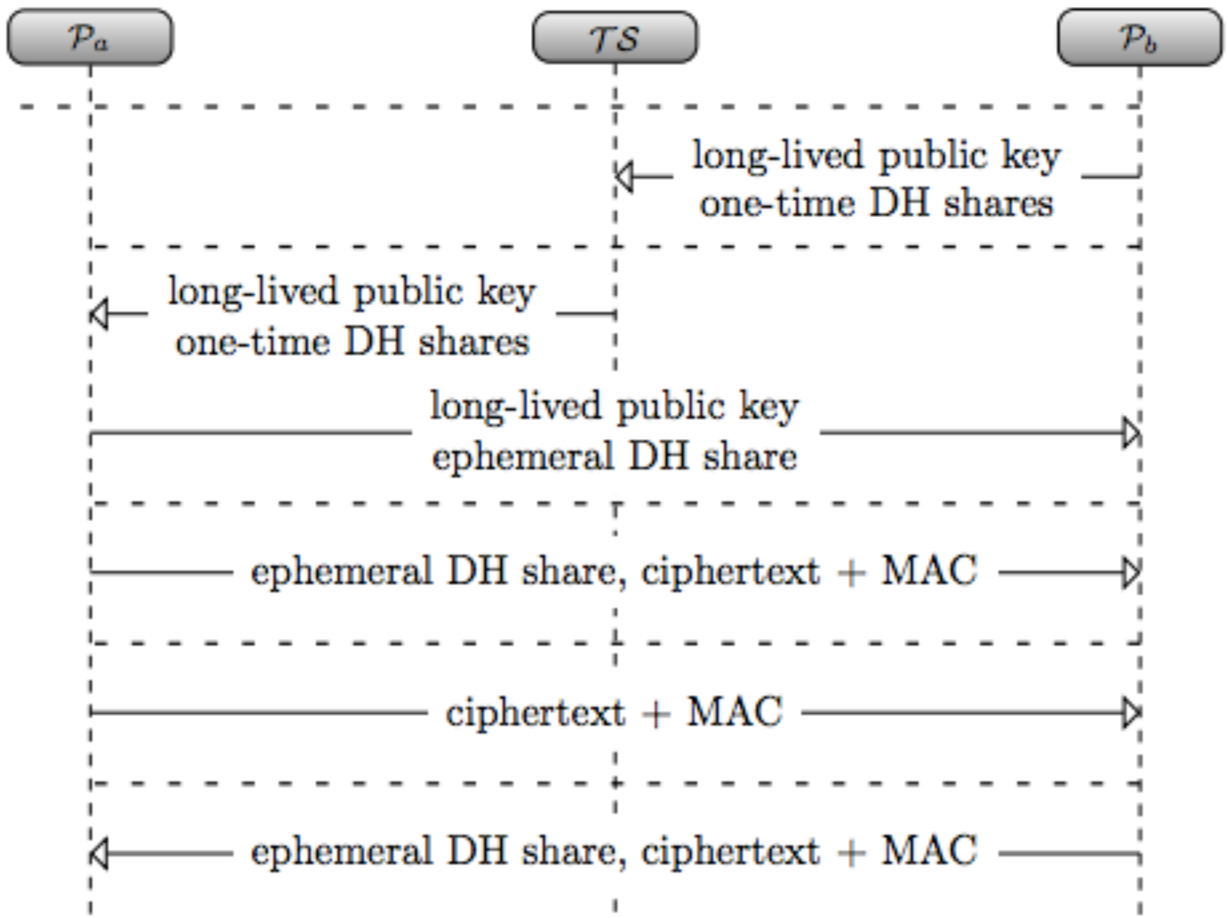
- End-to-end encrypted messaging is a big topic
- TextSecure is protocol adopted by WhatsApp (~1 billion users)

TextSecure

Encrypted/Signed SMS or IM

Sender

pk_A , sk_A



Receiver

pk_B , sk_B

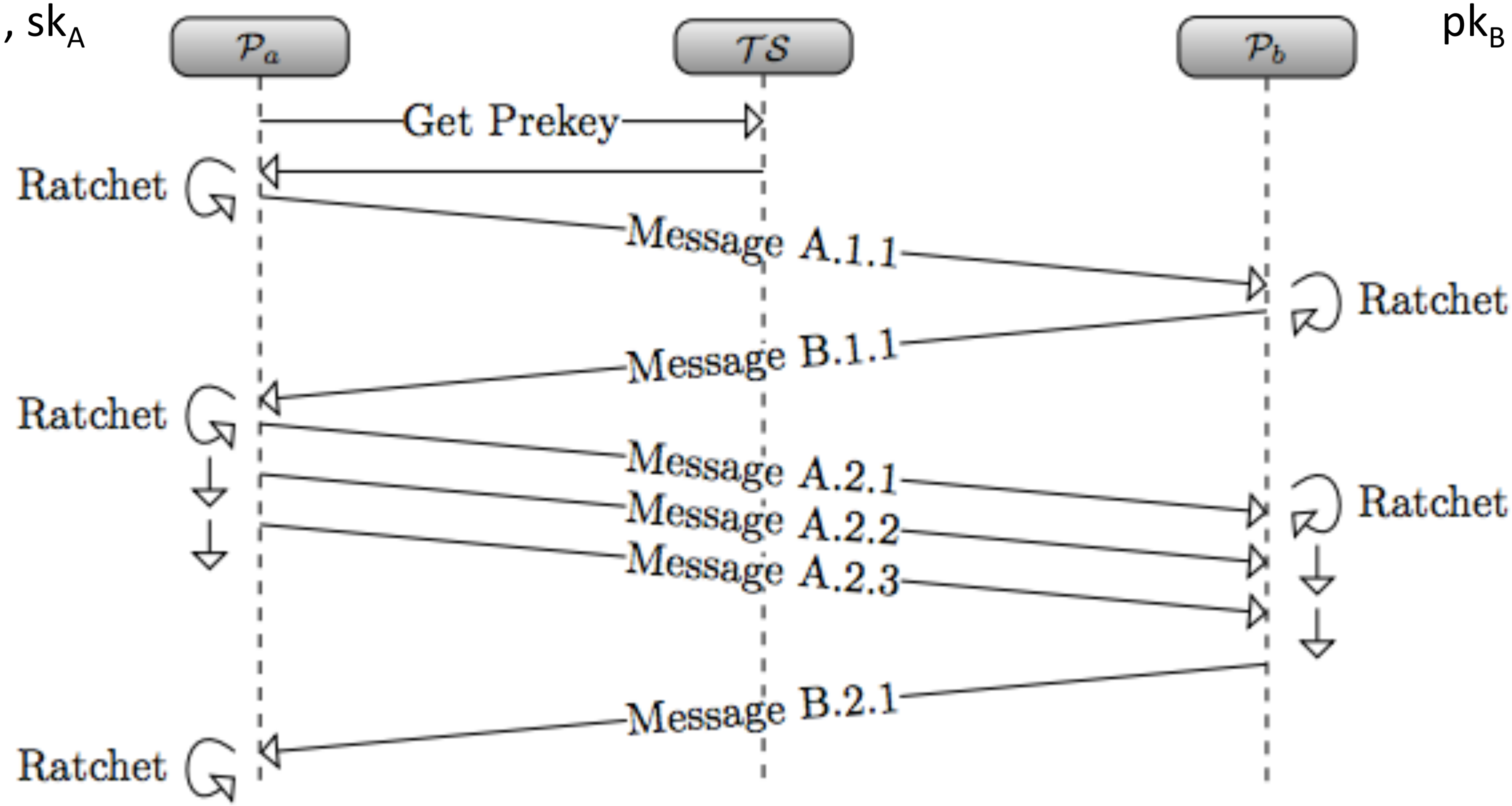
https://eprint.iacr.org/2014/904.pdf

TextSecure

Encrypted/Signed SMS or IM

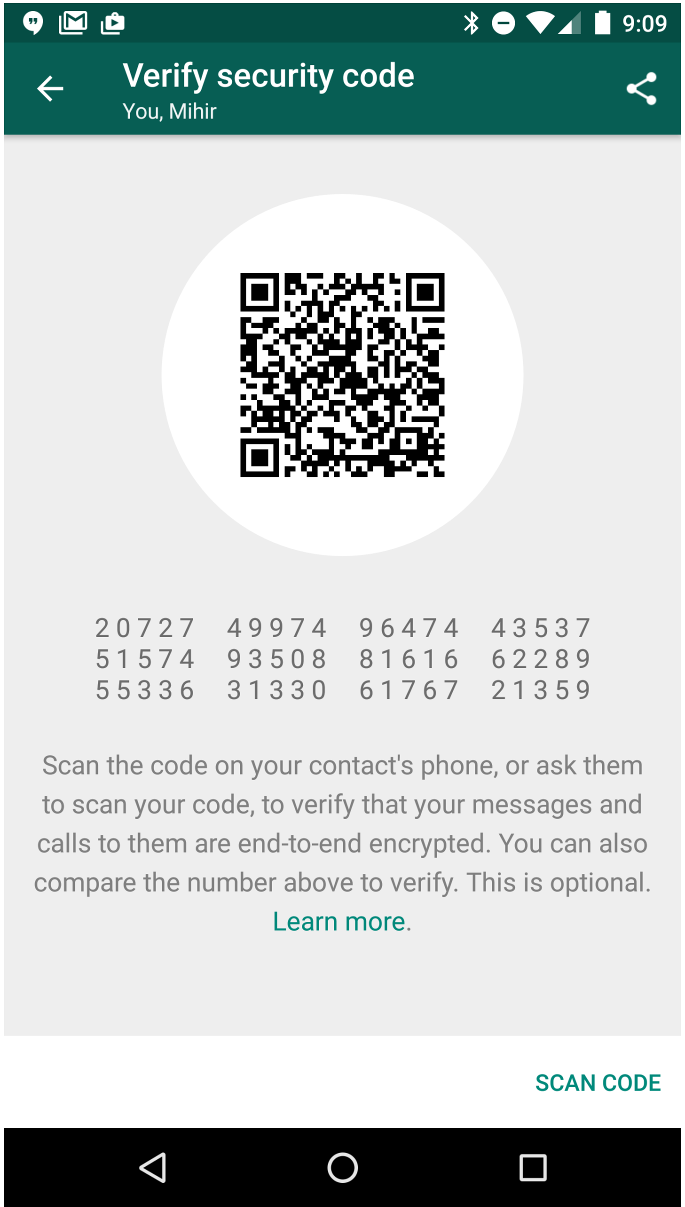
Sender
pk_A , sk_A

Receiver
pk_B , sk_B



https://eprint.iacr.org/2014/904.pdf

Verifying public keys



Summary

- Hybrid encryption uses combination of asymmetric and symmetric cryptography
  - Key encapsulation mechanisms (KEM) based on secure PKE, (elliptic curve) Diffie-Hellman
  - Use an authenticated encryption scheme for data encapsulation mechanism (DEM)
- PGP is historical example (and still somewhat widely used)
- End-to-end messaging for IM, chat hotter topic, now widely deployed