COS433/Math 473: Cryptography

Mark Zhandry
Princeton University
Spring 2020
Announcements

PR2 Due April 19th
HW6 Due April 23rd
Previously on COS 433...
PKE Syntax

Message space $\mathcal{M}$

Algorithms:
• $(sk,pk) \leftarrow \text{Gen}(\lambda)$
• $\text{Enc}(pk,m)$
• $\text{Dec}(sk,m)$

Correctness:
$\Pr[\text{Dec}(sk,\text{Enc}(pk,m)) = m: (sk,pk) \leftarrow \text{Gen}(\lambda)] = 1$
Security

One-way security

Semantic Security

CPA security

CCA Security
One-way Security

\[(sk, pk) \xleftarrow{\text{Gen}} \]
\[m' \xleftarrow{\text{M}} \]
\[c \xleftarrow{\text{Enc}(pk,m)} \]

\[pk, c \xrightarrow{\text{pk,c}} \]
Semantic Security

\[
\begin{align*}
&(sk, pk) \leftarrow \text{Gen}() \\
&c \leftarrow \text{Enc}(pk, m_b)
\end{align*}
\]
CPA Security

\[(sk, pk) \leftarrow \text{Gen}()\]

\[c \leftarrow \text{Enc}(pk, m_b)\]
CCA Security

$(sk, pk) \leftarrow \text{Gen}()$

c $\leftarrow \text{Enc}(pk, m_b)$
Q: Why no public key authenticated encryption?
One-Way Encryption from TDPs

\( \text{Gen}_E() = \text{Gen}_{\text{TDP}}() \)

\( \text{Enc}(pk,m): \text{Output } c = F(pk,m) \)

\( \text{Dec}(sk,c): \text{Output } m' = F^{-1}(sk,c) \)
ElGamal

Group $G$ of order $p$, generator $g$
Message space = $G$

$Gen()$:
• Choose random $a \leftarrow \mathbb{Z}_p^*$, let $h \leftarrow g^a$
• $pk=h$, $sk=a$

$Enc(pk,m \in \{0,1\})$:
• $r \leftarrow \mathbb{Z}_p$
• $c = (g^r, h^r \times m)$

$Dec?$
Today

CCA Secure Encryption

Digital Signatures
CCA-Secure Encryption

Non-trivial to construct with provable security

Most efficient constructions have heuristic security
CCA Secure PKE from TDPs

Let $\left(\text{Enc}_{\text{SKE}}, \text{Dec}_{\text{SKE}}\right)$ be a CCA-secure secret key encryption scheme.

Let $\left(\text{Gen}, F, F^{-1}\right)$ be a TDP

Let $H$ be a hash function
CCA Secure PKE from TDPs

\[
\begin{align*}
\text{Gen}_{\text{PKE}}() &= \text{Gen}() \\
\text{Enc}_{\text{PKE}}(pk, m): &
\begin{itemize}
  \item Choose random \( r \)
  \item Let \( c \leftarrow F(pk, r) \)
  \item Let \( d \leftarrow \text{Enc}_{\text{SKE}}(H(r), m) \)
  \item Output \((c_0, c_1)\)
\end{itemize}
\]

\[
\begin{align*}
\text{Dec}_{\text{PKE}}(sk, (c, d)): &
\begin{itemize}
  \item Let \( r \leftarrow F^{-1}(sk, c) \)
  \item Let \( m \leftarrow \text{Dec}_{\text{SKE}}(H(r), d) \)
\end{itemize}
\end{align*}
\]
CCA Secure PKE from TDPs

**Theorem:** If \((\text{Enc}_{\text{SKE}}, \text{Dec}_{\text{SKE}})\) is a CCA-secure secret key encryption scheme, \((\text{Gen}, F, F^{-1})\) is a TDP, and \(H\) is modeled as a random oracle, then \((\text{Gen}_{\text{PKE}}, \text{Enc}_{\text{PKE}}, \text{Dec}_{\text{PKE}})\) is a CCA secure public key encryption scheme.
Theorem: For RSA TDP, if $G, H$ are modeled as a random oracles, then $(Gen_{PKE}, Enc_{PKE}, Dec_{PKE})$ is a CCA secure public key encryption scheme.
Insecure OAEP Variants

\[ c = F(pk, (m, O^+, y)) \]

May contain \( m \) in the clear

- \( F(pk, (m, x, y)) \)
  \[ = (m, F'(pk, (x, y)) \) \]
Insecure OAEP Variants
Why padding?

All ciphertexts decrypt to valid messages
• Makes it hard to argue security
Digital Signatures
(aka public key MACs)
Message Authentication Codes

Goal: If Eve changed $m$, Bob should reject.
Problem

What if Alice and Bob have never met before to exchange key $k$?

Want: a public key version of MACs where Bob can verify without having Alice’s secret key
Message Integrity in Public Key Setting

Goal: If Eve changed $m$, Bob should reject $m'$
Digital Signatures

Algorithms:
- \( \text{Gen}() \rightarrow (sk,pk) \)
- \( \text{Sign}(sk,m) \rightarrow \sigma \)
- \( \text{Ver}(pk,m,\sigma) \rightarrow 0/1 \)

Correctness:
\[ \Pr[\text{Ver}(pk,m,\text{Sign}(sk,m))=1: (sk,pk) \leftarrow \text{Gen()}) = 1 \]
Security Notions?

Much the same as MACs, except adversary gets verification key
1-time Security For Signatures

\((sk, pk) \leftarrow \text{Gen}()\)

\(\sigma \leftarrow \text{Sign}(sk, m)\)

Output 1 iff:

- \(m^* \neq m\)
- \(\text{Ver}(pk, m^*, \sigma^*) = 1\)

\(1\text{CMA-Adv}(\cdot) = \Pr[\text{outputs } 1]\)
Many-time Signatures

\[ \text{Output 1 iff:} \]

\[ \bullet \; m^* \notin \{m_1, \ldots\} \]
\[ \bullet \; \text{Ver}(pk, m^*, \sigma^*) = 1 \]

\[ \text{CMA-Adv} = \Pr[ \text{outputs 1} ] \]
Strong Security

\[(m^*, \sigma^*) \not\in \{(m_1, \sigma_1)\ldots}\]  
\[\text{Ver}(pk, m^*, \sigma^*) = 1\]

\[\text{CMA-Adv}(\mathcal{A}) = \Pr[\text{\text{ outputs 1}}]\]
Building Digital Signatures

Non-trivial to construct with provable security

Most efficient constructions have heuristic security
Signatures from TDPs?

\[ \text{Gen}_{\text{Sig}}() = \text{Gen}() \]

\[ \text{Sign}(sk,m) = F^{-1}(sk,m) \]

\[ \text{Ver}(pk,m,\sigma): F(pk, \sigma) == m \]
Signatures from TDPs

\( \text{Gen}_{\text{Sig}}() = \text{Gen}() \)

\( \text{Sign}(\text{sk}, m) = F^{-1}(\text{sk}, H(m)) \)

\( \text{Ver}(\text{pk}, m, \sigma): F(\text{pk}, \sigma) = H(m) \)

**Theorem:** If \( (\text{Gen}, F, F^{-1}) \) is a secure TDP, and \( H \) is “modeled as a random oracle”, then \( (\text{Gen}_{\text{Sig}}, \text{Sign}, \text{Ver}) \) is (strongly) CMA-secure.
Basic Rabin Signatures

\textbf{Gen}_{\text{Sig}}(): \text{let } p, q \text{ be random large primes}

\text{sk} = (p, q), \text{ pk } = N = pq

\textbf{Sign(sk,m): } \text{Solve equation } \sigma^2 = H(m) \mod N \text{ using factors } p, q

• Output \sigma

\textbf{Ver(pk,m,\sigma): } \sigma^2 \mod N == H(m)
Problems

\( H(m) \) might not be a quadratic residue
Can only sign roughly \( \frac{1}{4} \) of messages

Suppose adversary makes multiple signing queries on the same message
- Receives \( \sigma_1, \sigma_2, \ldots \) such that \( \sigma_i^2 \mod N = H(m) \)
- After enough tries, may get all 4 roots of \( H(m) \)
- Suppose \( \sigma_1 \neq \pm \sigma_2 \mod N \)
- Then \( \text{GCD}(\sigma_1 - \sigma_2, N) \) will give a factor
One Solution

\textbf{Gen}_{\text{Sig}}(): \text{ let } p, q \text{ be primes, } a, b, c \text{ s.t.}
\begin{itemize}
  \item \text{a} \text{ is a non-residue } \text{mod } p \text{ and } q,
  \item \text{b} \text{ is a residue } \text{mod } p \text{ but not } q,
  \item \text{c} \text{ is a residue } \text{mod } q \text{ but not } p
\end{itemize}
\text{ sk } = (p, q, a, b, c), \text{ pk } = (N = pq, a, b, c)

\textbf{Sign}(sk, m):
\begin{itemize}
  \item \text{Solve equation } \sigma^2 \in \{1, a, b, c\} \times H(m) \text{ mod } N
  \item \text{Output } \sigma
\end{itemize}

\textbf{Ver}(pk, m, \sigma): \sigma^2 \text{ mod } N \in \{1, a, b, c\} \times H(m)
One Solution

Exactly one of $\{1, a, b, c\} \times H(m)$ is a residue $\text{mod } N$

$\Rightarrow$ Solution guaranteed to be found

Still have problem that multiple queries on same message will give different roots
One Solution

Possibilities:

• Have signer remember all messages signed

• Choose root that is itself a quadratic residue
  (if $-1$ is not a residue mod $p,q$, there will be exactly one)
Another Solution

\[ \textbf{Gen}_{\text{Sig}}(): \text{ let } p, q \text{ be random large primes } \]
\[ \quad sk = (p, q), \ pk = N = pq \]

\[ \textbf{Sign}(sk, m): \text{ Repeat until successful:} \]
\[ \quad \bullet \text{ Choose random } u \leftarrow \{0,1\}^\lambda \]
\[ \quad \bullet \text{ Solve equation } \sigma^2 = H(m, u) \mod N \]
\[ \quad \bullet \text{ Output } (u, \sigma) \]

\[ \textbf{Ver}(pk, m, (u, \sigma)): \sigma^2 \mod N = H(m, u) \]
Another Solution

In expectation, after 4 tries will have success

(Whp) Only ever get a single root of a given $H(m,u)$

**Theorem:** If factoring is hard and $H$ is modeled as a random oracle, then Rabin signatures are (weakly) CMA secure
Another Solution

**Sign**(sk,m): Repeat until successful:
- Choose random $u \leftarrow \{0,1\}^\lambda$
- Solve equation $\sigma^2 = H(m,u) \mod N$ using factors $p,q$, where $\sigma < (N-1)/2$
- Output $(u,\sigma)$

**Ver**(pk,m,(u,σ)): $\sigma^2 \mod N = H(m,u)^{\land\sigma < (N-1)/2}$

**Theorem:** If factoring is hard and $H$ is modeled as a random oracle, then Rabin signatures are strongly CMA secure
Schnorr Signatures

\[ sk = w \]
\[ pk = h := g^w \]

**Sign\((sk, m)\):**
- \( r \leftarrow \mathbb{Z}_p \)
- \( a \leftarrow g^r \)
- \( b \leftarrow H(m, a) \)
- \( c \leftarrow r + wb \)
- Output \((a, c)\)

**Ver\((h, m, (a, c))\):**
- \( b \leftarrow H(m, a) \)
- \( a \times h^b =? g^c \)

**Theorem:** If Dlog is hard and \( H \) is modeled as a random oracle, then Schnorr signatures are strongly CMA secure
What’s the Smallest Signature?

RSA Hash-and-Sign: 2 kilobits

ECDSA (variant of Schnorr using “elliptic curves”): around 512 bits

BLS: 256 bits

Are 128-bit signatures possible?
- No fundamental reason for impossibility, but all (practical) schemes require 256 bits or more
Digital Signatures and the Public Key Infrastructure
Digital Signatures and the Public Key Infrastructure
Digital Signatures and the Public Key Infrastructure

\[ c' = \text{Enc}(pk', m) \]

Alice \rightarrow Eve \rightarrow Bob \rightarrow contractor

\[ \text{pk} \rightarrow \text{sk} \]

\[ m \]

\[ \text{pk}' \rightarrow \text{sk}' \]
Digital Signatures and the Public Key Infrastructure

\[ c' = \text{Enc}(pk', m) \]

\[ c = \text{Enc}(pk, m) \]
Takeaway

Need some authenticated channel to ensure distribution of public keys

But how to authenticate channel in the first place without being able to distribute public keys?
Solution: Certificate Authorities

CA

Business
Government Agency
Department within company

$sk_{CA}$

$pk_{CA}$
Solution: Certificate Authorities

\[ \text{Cert}_{\text{CA} \rightarrow \text{B}} = \text{Sign}(\text{sk}_{\text{CA}}, "\text{Bob's public key is } \text{pk}_B") \]
Solution: Certificate Authorities

Bob is typically some website
• Obtains Cert by, say, sending someone in person to CA with $pk_B$
• Only needs to be done once

If Alice trusts CA, then Alice will be convinced that $pk_B$ belongs to Bob

Alice typically gets $pk_{CA}$ bundled in browser
Limitations

Everyone must trust same CA
• May have different standards for issuing certs

Single point of failure: if $sk_{CA}$ is compromised, whole system is compromised

Single CA must handle all verification
Multiple CAs

There are actually many CA’s, CA₁, CA₂,…

Bob obtains cert from all of them, sends all the certs with his public key

As long as Alice trusts one of the CA’s, she will be convinced about Bob’s public key
Certificate Chaining

CA issues $\text{Cert}_{CA \rightarrow B}$ for Bob

Bob can now use his signing key to issue $\text{Cert}_{B \rightarrow D}$ to Donald

Donald can now prove his public key by sending $(\text{Cert}_{CA \rightarrow B}, \text{Cert}_{B \rightarrow D})$

• Proves that CA authenticated Bob, and Bob authenticated Donald
Certificate Chaining

For Bob to issue his own certificates, a standard cert should be insufficient
• CA knows who Bob is, but does not trust him to issue certs on its behalf

Therefore, Bob should have a stronger cert:

\[ \text{Cert}_{CA \rightarrow B} = \text{Sign}(sk_{CA}, \text{“Bob’s public key is } pk_B \text{ and he can issue certificates on behalf of CA”}) \]
Certificate Chaining

One root CA

Many second level CAs CA₁, CA₂,...
- Each has $\text{cert}_{\text{CA} \rightarrow \text{CA}_i}$

Advantage: eases burden on root

Disadvantage: now multiple points of failure
Invalidating Certificates

Sometimes, need to invalidate certificates
• Private key stolen
• User leaves company
• Etc

Options:
• Expiration
• Explicit revocation
Announcements

PR2 Due April 19th
HW6 Due April 23rd
Crypto from Minimal Assumptions
Many ways to build crypto

We’ve seen many ways to build crypto
• SPN networks
• LFSR’s
• Discrete Log
• Factoring

Questions:
• Can common techniques be abstracted out as theorem statements?
• Can every technique be used to build every application?
One-way Functions

The minimal assumption for crypto

Syntax:
• Domain $\mathbb{D}$
• Range $\mathbb{R}$
• Function $F: \mathbb{D} \rightarrow \mathbb{R}$

No correctness properties other than deterministic
Security?

**Definition:** $F$ is $(t, \varepsilon)$-One-Way if, for all running in time at most $t$,

$$\Pr[x \leftarrow (F(x)), x \leftarrow D] < \varepsilon$$

Trivial example:

$F(x) =$ parity of $x$

Given $F(x)$, impossible to predict $x$
**Definition:** $F$ is $(t, \varepsilon)$-One-Way if, for all running in time at most $t$,

$$\Pr[F(x) = F(y) : y \leftarrow (F(x)), x \leftarrow D] < \varepsilon$$
Examples

Any PRG

Any Collision Resistant Hash Function (with sufficient compression)

\[ F(p,q) = pq \]

\[ F(g,a) = (g,g^a) \]

\[ F(N,x) = (N,x^3 \mod N) \text{ or } F(N,x) = (N,x^2 \mod N) \]
What’s Known

CRH

CPA
-PKE

CCA
-PKE

OWF

PRG

Com

PRF

MAC

Auth
Enc

PRP

SKE

Sig