COS433/Math 473: Cryptography

Mark Zhandry
Princeton University
Fall 2020
Announcements/Reminders

HW6 due Nov 24

PR2 due Dec 5

No lecture on Thursday (Nov 19)
Previously on COS 433...
Crypto from Minimal Assumptions
What’s Known

CRH

CPA - PKE

CCA - PKE

TDP

OWF

PRG

Com

Sig

PRF

MAC

 Auth

Enc

PRP

SKE

ZK

TDP
A PRF

\[ F(k,1) \quad F(k,2) \quad F(k,3) \quad F(k,4) \quad F(k,5) \quad F(k,6) \quad F(k,7) \quad F(k,8) \quad F(k,9) \quad F(k,10) \quad F(k,11) \quad F(k,12) \quad F(k,13) \quad F(k,14) \]
Today

OWP $\rightarrow$ PRGs
OWF $\rightarrow$ One-time Signature
Black box separations

If time, cryptocurrencies
One-way permutation $\rightarrow$ PRGs

OWP = OWF that is also a permutation
  - $F: D \rightarrow D$ is a permutation

Examples:
  - RSA function
  - Discrete exponentiation
One-way permutation $\rightarrow$ PRGs

Let $h$ be a hardcore bit for $F$

Hardcore bit equivalent to PRG security
Hardcore bits for OWPs?

Known OWPs have hardcore bits
  • E.g. **LSB, Half** for RSA, **Half** for Dlog

What about general OWPs?
Yao’s Method

Let $F$ be a OWP with domain $\{0,1\}^n$

Claim: $\exists i$ such that $\forall$ PPT $A$

$$\Pr[A(F(x)) = x_i] < 1 - 1/2^n$$

Proof: otherwise, $\forall i$, $\exists A_i$ s.t.

$$\Pr[A_i(F(x)) = x_i] \geq 1 - 1/2^n$$

Adversary $A(y) = A_1(y) || A_2(y) || ...$

$$\Pr[A(F(x)) = x] \geq 1/2$$
Yao’s Method

Let $F$ be a OWP with domain $\{0,1\}^n$

Claim: $\exists i$ such that $\forall$ PPT $A$
$Pr[A(F(x)) = x_i] < 1 - 1/2^n$

Let $F'(x^{(1)},...,x^{(t)}) = (F(x^{(1)}),...,F(x^{(t)}))$
$h(x^{(1)},...,x^{(t)}) = x^{(1)}_i \oplus x^{(2)}_i \oplus ... \oplus x^{(t)}_i$

Yao’s XOR lemma $\Rightarrow h$ is hardcore for $F'$
Goldreich Levin

Let $F$ be a OWP with domain $\{0,1\}^n$ and range $Y$

Let $F':\{0,1\}^{2n} \rightarrow \{0,1\}^n \times Y$ be:

$$F'(r,x) = r, F(x)$$

Define $h(r,x) = \langle r,x \rangle = \Sigma r_i x_i \mod 2$

**Theorem (Goldreich-Levin):** If $F$ is one-way, then $h$ is a hc bit for $F'$
OWF $\rightarrow$ PRGs

Yao, Goldreich-Levin also work for general OWFs

However, $(F(x), h(x))$ may not be a PRG for a general OWF
• Output may be shorter than input
• $F$ may be biased

With some effort, can build PRF from any one-way function using similar ideas
Lamport Signatures

Let $F : X \rightarrow Y$ be a one-way function

Let $M = \{0,1\}^n$ be message space

Gen():

$\begin{align*}
X & \downarrow \\
\begin{array}{cccccc}
X_{1,0} & X_{2,0} & X_{3,0} & X_{4,0} & X_{5,0} \\
X_{1,1} & X_{2,1} & X_{3,1} & X_{4,1} & X_{5,1}
\end{array} & \Rightarrow \\
F & \Rightarrow \\
\begin{array}{cccccc}
y_{1,0} & y_{2,0} & y_{3,0} & y_{4,0} & y_{5,0} \\
y_{1,1} & y_{2,1} & y_{3,1} & y_{4,1} & y_{5,1}
\end{array} & \text{pk} \\
sk & \Rightarrow
\end{align*}$
Lamport Signatures

\textbf{Sign}(sk, m): \((x_{i,m_i})_{i=1,\ldots,n}
\)

\begin{align*}
\begin{array}{cccccc}
  x_{1,0} & x_{2,0} & x_{3,0} & x_{4,0} & x_{5,0} \\
  x_{1,1} & x_{2,1} & x_{3,1} & x_{4,1} & x_{5,1}
\end{array}
\end{align*}

\textbf{Ver}(pk, m, \sigma): F(x_{i,m_i}) = y_{i,m_i}

\begin{align*}
\begin{array}{cccccc}
  y_{1,0} & y_{2,0} & y_{3,0} & y_{4,0} & y_{5,0} \\
  y_{1,1} & y_{2,1} & y_{3,1} & y_{4,1} & y_{5,1}
\end{array}
\end{align*}
Lamport Signatures

**Theorem:** If $F$ is a secure OWF, then $(\text{Gen,Sign,Ver})$ is a (weakly) secure one-time signature scheme.
Proof

Since $m^* \neq m$, $\exists i \text{ s.t. } m^*_i \neq m_i$

Suppose we know $i$, $m_i = 1-b$, $m^*_i = b$

Construct adversary that inverts OWF
Proof

\[
\begin{align*}
\mathbf{y}^* &= y_{1,0} \quad y_{2,0} \quad \Box \quad y_{4,0} \quad y_{5,0} \\
&\quad y_{1,1} \quad y_{2,1} \quad y_{3,1} \quad y_{4,1} \quad y_{5,1} \\
&\quad \mathbf{x}^* \\
\end{align*}
\]
Proof

View of exactly as in 1-time CMA experiment, assuming
• $i$th bit of $m = b$
• $i$th bit of $m^* = 1 - b$

If always chooses $m, m^*$ with these properties, and forges with probability $\epsilon$, then inverts with probability $\epsilon$
Proof

In general, may choose \( m, m^* \) to differ at arbitrary places
• May be randomly chosen, may depend on \( pk \), may even depend on \( \sigma \)
• May never be at certain places

How do we make still succeed?
Proof

\[
\begin{align*}
\gamma_{1,0} & \quad \gamma_{2,0} & \quad \gamma^* & \quad \gamma_{4,0} & \quad \gamma_{5,0} \\
\gamma_{1,1} & \quad \gamma_{2,1} & \quad \gamma_{3,1} & \quad \gamma_{4,1} & \quad \gamma_{5,1}
\end{align*}
\]

\[i, b \in [n] \times \{0, 1\}\]

\[
\begin{align*}
X_{1,0} & \quad X_{2,0} & \quad i, b & \quad X_{4,0} & \quad X_{5,0} \\
X_{1,1} & \quad X_{2,1} & \quad X_{3,1} & \quad X_{4,1} & \quad X_{5,1}
\end{align*}
\]

If need \(X_{i,b}\), abort

If no \(X_{i,b}\), abort

\[
\begin{align*}
X_{1,0} & \quad X_{2,0} & \quad \gamma^* & \quad X_{4,0} & \quad X_{5,0} \\
X_{1,1} & \quad X_{2,1} & \quad X_{3,1} & \quad X_{4,1} & \quad X_{5,1}
\end{align*}
\]
Proof

\( pk \) independent of \((i, b)\)
- \( m \) independent of \((i, b)\)
- Therefore, \( \Pr[m_i=1-b]=\frac{1}{2} \)

Conditioned on \( m_i=1-b \),
- Signing succeeds
- \( \sigma \) independent of \( i \)
- \( \text{螯} \) forges with probability \( \varepsilon \), independent of \( i \)
Proof

We know if Showcase forgives, then $m^* \neq m$

Since $m^*$ independent of $i$, have prob at least $1/n$ that $m^*_i = 1 - m_i = b$

In this case, Showcase succeeds in inverting $y^*$

- Prob = $\frac{1}{2} \times \varepsilon \times \frac{1}{n} = \varepsilon / 2n$
What's Known

- CRH
- CPA - PKE
- CCA - PKE
- TDP
- OWF
- PRG
- Com
- PRF
- MAC
- Auth
- Enc
- PRP
- SKE
- ZK
- Sig
- CPA - PKE
- TDP
Generally Believed That...

\[
\text{OWF} \not\Rightarrow \text{CRHF, OWP, PKE}
\]

\[
\text{CRHF} \not\Rightarrow \text{OWP, PKE}
\]

\[
\text{OWP} \not\Rightarrow \text{CRHF, PKE}
\]

\[
\text{PKE} \not\Rightarrow \text{CRHF}
\]
Black Box Separations

How do we argue that you cannot build collision resistance from one-way functions?

• We generally believe both exist!

Observation: most natural constructions treat underlying objects as black boxes (don’t look at code, just input/output)

Maybe we can rule out such natural constructions
Black Box Separations

Present a world where one-way functions exist, but collision resistance does not

Hopefully, natural (black box) constructions make sense in this world
• Can construct PRGs, PRFs, PRPs, Auth-Enc, etc
Separating PKE from OWF, CRHF

Recall: random oracle model

Computation power is unlimited, but number of calls to random oracle is polynomial
Separating PKE from OWF

In ROM, despite unlimited computational power, one-way functions, CRHF exist

• $F(x) = H(x)$
• Can only invert oracle by brute-force search (exponentially many queries)
• Can only find collisions by birthday attack (also exponentially many queries)
Separating PKE from OWF

**Theorem:** If $H$ is a random oracle, then for any PKE in which Alice and Bob make at most $n$ queries, there is an (inefficient) adversary than makes at most $O(n^2)$ queries.

Intuition: if Alice can send message to Bob, then either
(1) Message can be learned from communication alone, or
(2) Alice and Bob must have a common RO query

In case (2), Alice and Bob’s RO queries can’t have too much entropy → Adversary can learn with few queries
Cryptocurrency/Blockchain
Features of Physical Cash

Essentially anonymous

Hard to counterfeit

Easy to verify
Limitations of Physical Cash

Cannot be used online
• Instead, need to involve banks
• Banks see all transactions
• Merchants can also track you

Requires central government to issue
• Ok for most people in US, but maybe you don’t trust the government
Digital Cash

Currency is now 1s and 0s

Crypto can make digital currency easy to verify, hard to mint

Major challenge: prevent double spending (Also decentralizing minting process)
Solution: Public Ledger

Bank transfers $$ to Alice

Each bill has unique serial number
Solution: Public Ledger

<table>
<thead>
<tr>
<th>Bank transfers $$$ to Alice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice transfers $$$ to Bob</td>
</tr>
</tbody>
</table>
Solution: Public Ledger

Bank transfers $$ to Alice

Alice transfers $$ to Bob
Solution: Public Ledger

Bank maintain ledger?
- But then bank must be involved in every transaction
- How does bank prevent malicious Bob from claiming Alice transferred money to him?

Anonymity also lost, since all transactions public
Solution: Use Signatures

\[
\sigma_1 = \text{Sign}(sk_{\text{Bank}}, \text{"pk}_{\text{Bank}} \text{ transfers } \$\text{ to } \text{pk}_A\text{"})
\]
Solution: Use Signatures

\[
\begin{align*}
\text{pk}_{\text{Bank}} & \text{ transfers } $$ \text{ to } \text{pk}_{A}, \sigma_1 \\
\text{pk}_{A} & \text{ transfers } $$ \text{ to } \text{pk}_{B}, \sigma_2
\end{align*}
\]

\[
\sigma_2 = \text{Sign}(sk_{A}, \text{"pk}_{A} \text{ transfers } $$ \text{ to } \text{pk}_{B}\"")
\]
Solution: Use Signatures

By using public key as identity, transactions not immediately traced to individual
• Though can still trace sequences of transactions

By signing, prevents Bob from claiming Alice gave him money when she didn’t
Decentralized Currency

Removing the bank is hard:

• How is ledger maintained?
• How to prevent ledger from being tampered with
• Who mints new currency?
• How do we limit supply?
Proofs of Work

Prove that some amount of computation has been performed

Ex:
• Let $H$ be a hash function (modeled as a RO)
• An input $x$ such that $H(x) = 0^{t}*****$ is a “proof” that you computed approximately $2^{t}$ hashes
Proofs of Work and Cryptocurrency

Idea: currency is a proof of work

• Limits supply of money, so keeps inflation in check
• Now, anyone can mint new money

Proofs of work not the only option

• Proofs of stake
• Proofs of space
Blockchain

Immutable public ledger

Block:

<table>
<thead>
<tr>
<th>$pk_A$ transfers $$1$ to $pk_B$, $\sigma_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_1$</td>
</tr>
<tr>
<td>$t_1$</td>
</tr>
</tbody>
</table>

Hashes to $0^t****$
Blockchain

Immutable public ledger

$pk_E$ transfers $\$0$ to $pk_F$, $\sigma_0$

$\begin{array}{cc}
\text{H} & \\
\h_0 & \tau_0 \\
\end{array}$

$pk_A$ transfers $\$1$ to $pk_B$, $\sigma_1$

$\begin{array}{cc}
\h_1 & \tau_1 \\
\end{array}$

$pk_C$ transfers $\$2$ to $pk_D$, $\sigma_2$

$\begin{array}{cc}
\h_2 & \tau_2 \\
\end{array}$

Hashes to $0^{\dagger\ddagger\ddagger\ddagger\ddagger}$
Blockchain

By making each block a proof of work, hard to modify blockchain

So proofs of work used to:
• Mint new money
• Add transactions to blockchain

Why would anyone go through the effort of adding transactions to the blockchain?
Blockchain

Idea: combine minting and adding blocks

Block:

\[
\begin{array}{|c|c|}
\hline
pk_A & \text{transfers } \$, \sigma_1 \\
\hline
h_1 & \tau_1 \\
\hline
pk_M & \text{mined } \\
\hline
\end{array}
\]

Hashes to \(0^{\dagger\ast\ast\ast\ast\ast}\)
Double Spending

$p_k_A$ transfers $$1 to $p_k_B$, $\sigma_1$

$p_k_A$ transfers $$1 to $p_k_C$, $\sigma_2$

What if Alice now claims this is the blockchain?
Double Spending

To prevent double spending, everyone always uses longest chain as the blockchain

If Alice tries to double spend, she will need to create a separate chain that is as long as the main chain
• As long as she has <<50% of computing power of mining power, will not be possible