Lecture 15 : Signatures and public-key encryption from lattices

Recap: SIS $\to$ CRHF

Today: More from lattices

SIS $\to$ trapdoor OWF $\to$ signatures

LWE : $\boxed{\text{new assumption}}$

↳ Regev's encryption scheme

↳ Post-quantum key-exchange (HW5)

↳ Google has implemented this!

↳ Ongoing NIST competition to develop standards for post-quantum cryptography

$SIS(n, m, q, B)$

integer parameters

Challenger

$A \xleftarrow{R} \mathbb{Z}_q^{n \times m}$

$\xrightarrow{A}$

Adversary

$\downarrow$

$x \in \mathbb{Z}^m, \; x \neq \vec{0}$

Adversary wins if :
1) $Ax = 0 \mod q$
2) $\|x\|_\infty \leq B$

Hash-function from SIS :

$$H_{SIS}(A, x) = A x \mod q$$

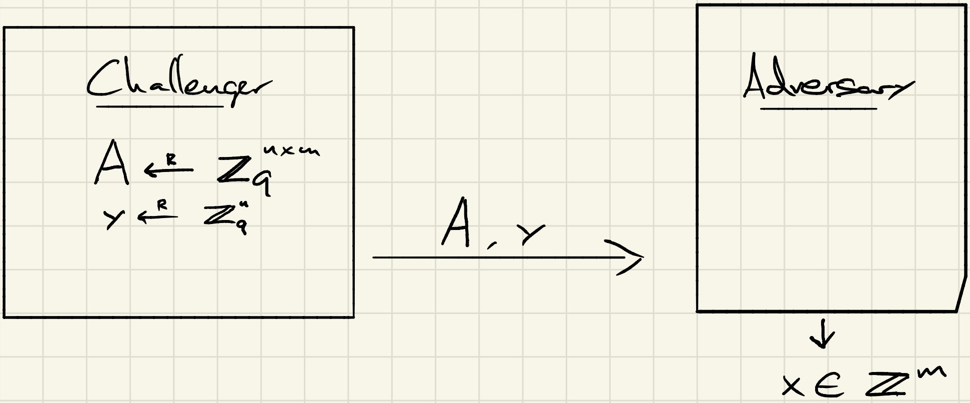
previous lecture

<u>Fact 1</u> : If SIS is hard, H_{SIS} is collision-resistant

<u>Fact 2</u> : For $\overbrace{\text{appropriate parameters } (n, m, q, B)}^{\text{HW 4 Problem 5b}}$, if SIS is hard then we can get a one-way function !

"There are only two hard things in CS: cache invalidation and naming things"
(Phil Karlton)

Inhomogenous SIS (ISIS)



Adversary wins if :
1) $Ax = y \mod q$
2) $\|x\|_\infty \leq B$

Fact: ISIS(n, m, q, B) is as hard as SIS(n, m, q, B)

↳ "there's nothing special about homogenous systems of equations"

$$\underset{\text{(SIS)}}{\text{solving } Ax = 0} \approx \underset{\text{(ISIS)}}{\text{solving } Ax - y = 0}$$

Trapdoors

To construct public-key primitives, we need certain tasks (e.g. signing, decryption) to be easy given some private information, and hard otherwise. Moreover, some tasks (e.g. encryption, sig verification) should be easy for everyone.

For symmetric-crypto, the "simplest" primitive is the one-way function. → gives PRFs, PRGs, blockciphers

For asymmetric-crypto, the "simplest" primitive is a trapdoor one-way function.

↳ gives PKE, digital signatures

Trapdoor one-way function (Diffie-Hellman 1976)

A collection of functions $\{f_k : X \to Y\}_{k \in K}$ is a trapdoor one-way function if:

- There is an efficient $\text{TrapGen}(1^\lambda)$ algorithm that outputs a "public" key $k \in K$ and a trapdoor td_k
- Given k, $f_k(x)$ can be efficiently computed for any $x \in X$
- Given k and $y = f_k(x)$ for $x \xleftarrow{R} X$, it is hard to find $x' \in X$ s.t. $f_k(x') = y$

$\Big\}$ "keyed OWF"

- There is an efficiently computable function $f^{-1}(td_k, y)$ that outputs $x \in X$ s.t. $f_k(x) = y$

Example:

$\boxed{\text{RSA}}$ $\quad f(x) = x^e \bmod N$, $\quad k = e$, $td_k = d$ s.t. $e \cdot d = 1 \bmod \phi(N)$

Remark: OWFs are necessary and sufficient for symmetric crypto

i.e, $\text{OWFs} \rightleftarrows \text{PRGs} \rightleftarrows \text{PRFs} \rightleftarrows \text{PRPs}$

Trapdoor OWFs are sufficient for public-key crypto but not necessary:

$\text{Trapdoor OWF} \rightarrow \text{PKE}$, $\quad \text{Trapdoor OWF} \not\leftarrow \text{PKE}$ [Gertner, Malkin, Reingold]

Lattice trapdoors

Let $f_A(x) = Ax$ (this is just the SIS hash function). We will show that we can use f_A as a trapdoor function.

- $\text{TrapGen}(n, m, q) \to (A, td_A)$ ("public" key: A; trapdoor: td_A)

Produces a matrix $A \in \mathbb{Z}_q^{n \times m}$ and a trapdoor td_A

- $f_A^{-1}(td_A, y) \to x$: outputs $x \in \mathbb{Z}_q^m$ s.t $Ax = y$ and $\|x\|_\infty \leq B$ ($y \in \mathbb{Z}_q^n$)

<u>Intuition:</u> Given the trapdoor td_A, solving the ISIS challenge for A is easy

There are many ways to construct a trapdoor td_A. We will (informally) describe one way: G-trapdoors

We start with a matrix $G \in \mathbb{Z}_q^{n \times m}$ such that the function $f_G(x) = Gx$ is <u>easy</u> to invert. That is given G and $y = Gx$ anyone can find $x' \in \{0,1\}^m$ such that $Gx' = y$

G is called a <u>gadget matrix</u>. We'll talk about them more on wednesday when discussing Fully Homomorphic Encryption. For now, all we need to know is that G is easy to construct (think about how you would do this!)

So f_A (for $A \xleftarrow{R} \mathbb{Z}_q^{n \times m}$) is hard to invert but has no trapdoor, and f_G is always easy to invert (so not one-way). We somehow need to mix the two.

The high-level construction is:

we assume ISIS$(n, 2m, q, B)$ is hard, for $B \approx m/4$

$\underline{\text{TrapGen}(n, 2m, q)}$:

- Sample $\bar{A} \xleftarrow{R} \mathbb{Z}_q^{n \times m}$
- Sample $R \xleftarrow{R} \{0,1\}^{m \times m}$
- Let $G \in \mathbb{Z}_q^{n \times m}$ be a public gadget matrix
- output $A = [\bar{A} \mid \bar{A}R + G] \in \mathbb{Z}_q^{n \times 2m}$

  $td_A = R$

matrix concatenation

$\underline{f_A^{-1}(td_A, y)}$: $\underline{\text{Goal}}$: output $x \in \mathbb{Z}_q^{2m}$ s.t $Ax = y$ and $\|x\|_\infty \leq B$

- Find $x^* \in \{0,1\}^m$ such that $Gx^* = y$
- set $x_0 = -Rx^*$, $x_1 = x^*$ (this step uses the trapdoor)
- output $\left[\frac{x_0}{x_1}\right]$

- $\underline{\text{Correctness}}$:

$$A\left[\frac{x_0}{x_1}\right] = \bar{A}x_0 + (\bar{A}R + G)x_1$$
$$= \cancel{-\bar{A}Rx^*} + \cancel{\bar{A}Rx^*} + \underbrace{Gx^*}_{y} = y$$

$$\|x_1\|_\infty = 1, \quad \|x_0\|_\infty = \|Rx^*\|_\infty \approx \frac{m}{4}$$

random binary matrix

binary vector independent of R

- $\underline{\text{Security}}$: This construction isn't quite secure (this requires a few more tricks)

We can show that if $m \geq 2n \log q$: $\left\{(\bar{A}, \bar{A}R) : \begin{matrix}\bar{A} \xleftarrow{R} \mathbb{Z}_q^{n \times m} \\ R \xleftarrow{R} \{0,1\}^{m \times m}\end{matrix}\right\} \overset{\text{stat}}{\approx} \left\{(\bar{A}, Y) : \begin{matrix}\bar{A} \xleftarrow{R} \mathbb{Z}_q^{n \times m} \\ Y \xleftarrow{R} \mathbb{Z}_q^{n \times m}\end{matrix}\right\}$

So A is indistinguishable from random.

If ISIS is hard, we can prove that f_A is hard to invert!

↳ this requires a powerful and useful result known as the Leftover Hash Lemma

Problem: a pre-image $x = \left[\frac{-Rx^*}{x^*}\right]$ leaks information about the trapdoor R

↳ e.g. if A and R are public/secret keys for a signature scheme (see below), then each signature contains such a pre-image x and leaks information about the signing key

Digital Signatures from ISIS

Pretty much identical to signatures constructed from the RSA trapdoor function ("Full domain hash construction")

KeyGen(1^λ): $(A, td_A) \leftarrow \text{TrapGen}(n, m, q)$

Set $pk = A$, $sk = td_A$

$H : \{0,1\}^* \to \mathbb{Z}_q^n$ modeled as a R.O.

Sign(sk, m): $y = H(m)$

$x = f_A^{-1}(td_A, y)$

output $\sigma = x$

Verify(pk, m, σ): $y = H(m)$, $x = \sigma$

check that $Ax = y$ and $\|x\|_\infty \leq B$

Security proof idea: ISIS adversary $\mathcal{A}$ gets a challenge (A, y) and sends $pk = A$ to the adversary $\mathcal{B}$ of the signature scheme.

→ guess the R.O. query that corresponds to the forged message m^*, and return $H(m^*) = y$

→ on a "sign" query for m, pick a random x , set $H(m) = Ax$, return $\sigma = x$

→ if $\mathcal{B}$ outputs a forged signature σ for m^*, σ is a solution to the ISIS challenge

Learning with Errors

A powerful and easy to use Lattice assumption:

$LWE(\underbrace{n, m, q}_{\text{positive integers, same as in SIS}}, \underbrace{\chi_B}_{\text{B-bounded distribution over } \mathbb{Z}_q \,:\, \underset{e \leftarrow \chi_B}{\mathbb{P}}\left[\|e\|_\infty \leq B\right] = 1}):$

$$\left\{ (A, s^T A + e^T) \;\middle|\; \begin{matrix} A \xleftarrow{R} \mathbb{Z}_q^{n \times m} \\ s \xleftarrow{R} \mathbb{Z}_q^{n} \\ e \leftarrow \chi_B^{m} \end{matrix} \right\} \overset{c}{\approx} \left\{ (A, u^T) \;\middle|\; \begin{matrix} A \xleftarrow{R} \mathbb{Z}_q^{n \times m} \\ u \xleftarrow{R} \mathbb{Z}_q^{m} \end{matrix} \right\}$$

Alternative view (transpose): $(\underbrace{A^T}_{\in \mathbb{Z}_q^{m \times n}}, A^T s + e) \approx (A^T, u)$

This is the "decision" version of LWE.

The search version might be more intuitive: given $(A, A^T s + e)$, recover s

↳ The search and decision versions of LWE are (roughly) equally hard!

↳ Solving noisy systems of equations is hard!

Comparison with ISIS:

ISIS	LWE
solve $Ax = y$	solve $A^T s \approx y$ (s.t. $\|A^T s - y\|_\infty \leq B$)
• n equations	• m equations
• m unknowns	• n unknowns
• $m \gg n$	• $m \gg n$
$\Rightarrow$ a solution exists for any y with high probability	$\Rightarrow$ if y is random, no solution exists with high probability / if $y = A^T s + e$, no other solution s' exists with high probability

Regev encryption (Regev 2005)

KeyGen(1^λ):

$$\left.\begin{array}{l} A \xleftarrow{R} \mathbb{Z}_q^{n\times m} \\ s \xleftarrow{R} \mathbb{Z}_q^{n} \\ e \leftarrow \chi_B^m \\ b^T = s^T A + e^T \end{array}\right\}$$ choose parameters such that $q/4 > mB$

set $sk = s$, $pk = (A, b^T)$ $\quad b^T \in \mathbb{Z}_q^{1\times m}$

this scheme encrypts a single bit at a time

Encrypt(pk, $x \in \{0,1\}$):

$r \xleftarrow{R} \{0,1\}^m$

$\lfloor\cdot\rfloor$ rounds down to nearest integer

$c_0 = Ar$, $\quad c_1 = b^T r + \lfloor \frac{q}{2} \rfloor \cdot x$

output $ct = (c_0, c_1) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$

Decrypt(sk, ct): $\quad ct = (c_0, c_1)$

$\tilde{x} = c_1 - s^T c_0$

if $|\tilde{x}| < q/4$ output $x = 0$

else output $x = 1$

Correctness:

$$\tilde{x} = c_1 - c_0^T s = b^T r + \lfloor \tfrac{q}{2} \rfloor \cdot x - s^T A r$$
$$= (s^T A + e^T) r + \lfloor \tfrac{q}{2} \rfloor \cdot x - s^T A r$$
$$= \cancel{s^T A r} + e^T r + \lfloor \tfrac{q}{2} \rfloor \cdot x - \cancel{s^T A r}$$
$$= e^T r + \lfloor \tfrac{q}{2} \rfloor \cdot x$$

we have $e \leftarrow \chi_B^m$ and $r \xleftarrow{R} \{0,1\}^m$ so $|e^T r| \leq mB < q/4$

So if $x = 0$, $|\tilde{x}| < \frac{q}{4}$. If $x = 1$, $|\tilde{x}| > \lfloor \frac{q}{2} \rfloor - \frac{q}{4} \geq \frac{q}{4}$

Security: (sequence of hybrids over the view of the adversary)

real experiment

Hyb_0: $pk = (A, b^T = s^T A + e^T)$, $c_0 = Ar$, $c_1 = b^T r + \lfloor \frac{q}{2} \rfloor \cdot x$

indistinguisable by LWE

Hyb_1: $pk = (A, v^T \xleftarrow{R} \mathbb{Z}_q^m)$, $c_0 = Ar$, $c_1 = v^T r + \lfloor \frac{q}{2} \rfloor \cdot x$

statistically indistinguishable using the leftover hash lemma

Hyb_2: $pk = (A, v^T \xleftarrow{R} \mathbb{Z}_q^m)$, $c_0 \xleftarrow{R} \mathbb{Z}_q^n$, $c_1 \xleftarrow{R} \mathbb{Z}_q$

In Hyb_2, the ciphertext is random and independent of the message x.