Instructor: David Wu (dwu4@cs.utexas.edu)

Previously... CS 346/388H provided introduction to cryptography
↳ Focus was on secure communication

symmetric encryption, MAC, authenticated encryption (based on OWFs / PRGs / PRFs)
public-key encryption, digital signatures (based on number-theoretic assumptions)

Today: cryptography enables secure computation
↳ Can parties compute functions on their secret inputs without sharing them? [Confidentiality]
↳ Can parties verify that a computation was performed correctly? [Integrity]

This course: Focus will be on how to construct advanced cryptographic objects for protecting + verifying computations

- Fully homomorphic encryption (FHE): given encryption of x $\Rightarrow$ encryption of $f(x)$ for any efficiently-computable f
  ↳ ciphertexts are still semantically secure!

  Enables outsourcing of computation to untrusted cloud

  Alice (x) — Server (f)

  $\xrightarrow{pk,\ \mathrm{Enc}(pk, x)}$

  $\xleftarrow{ct'}$

  Alice: ↓ $f(x)$ — Server: ↓ learns nothing

- Homomorphic signatures: given signature on x $\Rightarrow$ signature on $f(x)$ for any efficiently-computable f
  ↳ signatures are short: $|\sigma| = |f(x)| \cdot \mathrm{poly}(\lambda)$
  private: σ only reveals $f(x)$, hides x

  Enables ability to check computations on data

  Alice (f) — Server (x)

  $\xrightarrow{f}$

  $\xleftarrow{f(x),\ \sigma_{f(x)}}$

  ↓ Alice knows that f was properly computed on x (without needing to know or store x)

  for example, x might be a large database Alice cannot store and f is a query

- Functional encryption: decryption recovers a function of the message
  ↳ Ciphertexts associated with a message x $\Rightarrow$ decryption yields $f(x)$ and nothing more about x
  Keys are associated with a function f

  x could be an email
  f could be a spam filter
  } allows spam filtering on encrypted messaging

Many fancy capabilities – many of these are currently theoretical constructions
↳ Some of these are on the cusp of becoming practical and enable new privacy-preserving systems
} We will explore both!

Administrivia:

- Course website: https://www.cs.utexas.edu/~dwu4/courses/sp22
- See Piazza for announcements, notes will be posted to course website (1-2 days after lecture)
- Course consists primarily of project: can be a survey on topic of your choosing, an implementation project, or research
  - Consists of project proposal, milestone report, and final report
  - Can work in teams of 2 — See website for deadlines
  - Should work on project throughout the semester
  - Encouraged to discuss ideas with me in advance
- There will be 2 homework assignments: one due before spring break, one due at end of semester
  - Problems will be added as semester progresses (~ 1 problem every 1-2 weeks)
  - Need to submit at least 70% of problems (rounded down)
- You will need to scribe two lectures (one in each half of semester)
  - Scribe notes should be edited for completeness + typos, due week following lecture

Prerequisites: Will assume familiarity with concepts from CS 388H (e.g., hybrid arguments, random oracle model, simulation-based defs)

This semester: Lectures will be simultaneously broadcast over Zoom and recorded

Please participate virtually if you are feeling unwell

Course focus: lattice-based cryptography

- Conjectured post-quantum resilience
  - Number-theoretic assumptions like discrete log and factoring are insecure against quantum computers (typical approach is to solve a hidden subgroup problem in an abelian group)
    - ↳ let $\mathbb{G}$ be a group and $\mathbb{H}$ be a subgroup of $\mathbb{G}$

      suppose $f : \mathbb{G} \to S$ has the property $f(x) = f(y)$ whenever there exists $h \in H$ where $hx = y$ (f is fixed on the left cosets of H)

      f is fixed on H, so f defines H

      given oracle access to f, find a generating set for H
    - ↳ discrete log problem can be recast as hidden subgroup problem

      given $g, h = g^x$, find x

      suppose $\mathbb{G} = \langle g \rangle$ has prime order p

      define HSP in $\overbrace{\mathbb{Z}_p \times \mathbb{Z}_p}^{\text{additive group}}$ with function $f(\alpha, \beta) = h^\alpha g^{-\beta} = g^{\alpha x - \beta}$

      by construction, f hides the coset generated by $(1, x)$:

      $$H = \langle (1,x) \rangle = \{(0,0), (1,x), (2,2x), \dots, (p-1, (p-1)x)\}$$

      to see this, observe that if

      $$f(\alpha_1, \beta_1) = f(\alpha_2, \beta_2) \implies (\alpha_2 - \alpha_1)x = \beta_2 - \beta_1$$
      $$\implies (\alpha_2 - \alpha_1, \beta_2 - \beta_1) \in H$$
      $$\implies (\alpha_1, \beta_1) + (\alpha_2 - \alpha_1, \beta_2 - \beta_1) = (\alpha_2, \beta_2)$$

      solving HSP for f yields $(1, x)$, which gives the discrete log (polynomial-time on a quantum computer via Shor's algorithm)

      factoring can also be viewed as a HSP

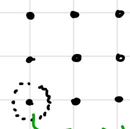
      non-abelian groups ↳ no poly-time quantum algorithms known:
      - graph isomorphism can be viewed as HSP in the symmetric group
      - various lattice problems can be viewed as HSP in the dihedral group

      we may revisit this later in the semester
  - Leading candidate in ongoing NIST post-quantum standardization efforts

- Security based on worst-case hardness
  - Cryptography has typically relied on average-case hardness (i.e., there exists some <u>distribution</u> of hard instances)
  - Lattice-based cryptography can be based on <u>worst-case</u> hardness (there does not exist an algorithm that solves <u>all</u> instances)
- Enables advanced cryptographic capabilities

<u>Definition</u>: An n-dimensional lattice $\mathcal{L} \subseteq \mathbb{R}^n$ is a discrete additive subspace of $\mathbb{R}^n$

- <u>Discrete</u>: For every $x \in \mathcal{L}$, there exists a neighborhood around x that only contains x:



neighborhood $B_\varepsilon(x) = \{ y \in \mathbb{R}^n : \|x - y\| \leq \varepsilon \}$

discrete means $B_\varepsilon(x) \cap \mathcal{L} = \{x\}$

- <u>Additive subspace</u>: For all $x, y \in \mathcal{L}$: $x + y \in \mathcal{L}$
  $-x \in \mathcal{L}$

<u>Examples</u>: $\mathbb{Z}^n$ (n-dimensional integer-valued vectors)

$q\mathbb{Z}^n$ (n-dimensional integer-valued vectors where each coordinate is multiple of q) "q-ary" lattice

Lattices typically contain infinitely-many points, but are <u>finitely-generated</u> by taking <u>integer</u> linear combinations of a small number of basis vectors:

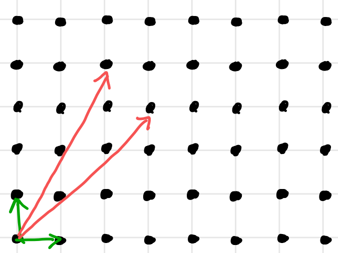
$$B = [b_1 \mid b_2 \mid \cdots \mid b_k] \in \mathbb{R}^{n \times k}$$

(vectors are linearly independent over $\mathbb{R}$)

← k is the rank of the lattice (full-rank: $k = n$)

$$\mathcal{L}(B) = \{ \sum_{i \in [n]} \alpha_i b_i \mid \alpha_i \in \mathbb{Z} \}$$
$$= B \cdot \mathbb{Z}^k$$

A lattice can have many basis:



standard basis for $\mathbb{Z}^2$

alternative basis for $\mathbb{Z}^2$

$\Big\}$ Choice of basis makes a big difference in hardness of lattice problems

↳ <u>often</u>: bad basis is public key
good basis is trapdoor

<u>Definition</u>. Let $\mathcal{L}$ be an n-dimensional lattice. Then, the minimum distance $\lambda_1(\mathcal{L})$ is the norm of the shortest non-zero vector in $\mathcal{L}$:

$$\lambda_1(\mathcal{L}) = \min_{v \in \mathcal{L} \setminus \{0\}} \|v\|$$

The i^{th} <u>successive minimum</u> $\lambda_i(\mathcal{L})$ is the smallest $r \in \mathbb{R}$ such that $\mathcal{L}$ contains i <u>linearly independent</u> basis vectors of norm at most r.