Lecture 1: Introduction to CS355!

Let's begin our exciting journey into advanced Cryptography!

2023-24 Instructors

- Wilson Nguyen
- Aditi Partap
- Trisha Datta

TAs

- Nolan Miranda
- Rohit Nema

Outline for Today

1. What is CS355 about?
2. Course logistics
3. One Way Functions (OWFs)
4. Pseudo-Random Generators (PRGs)
5. Hardcore Bits

Goals

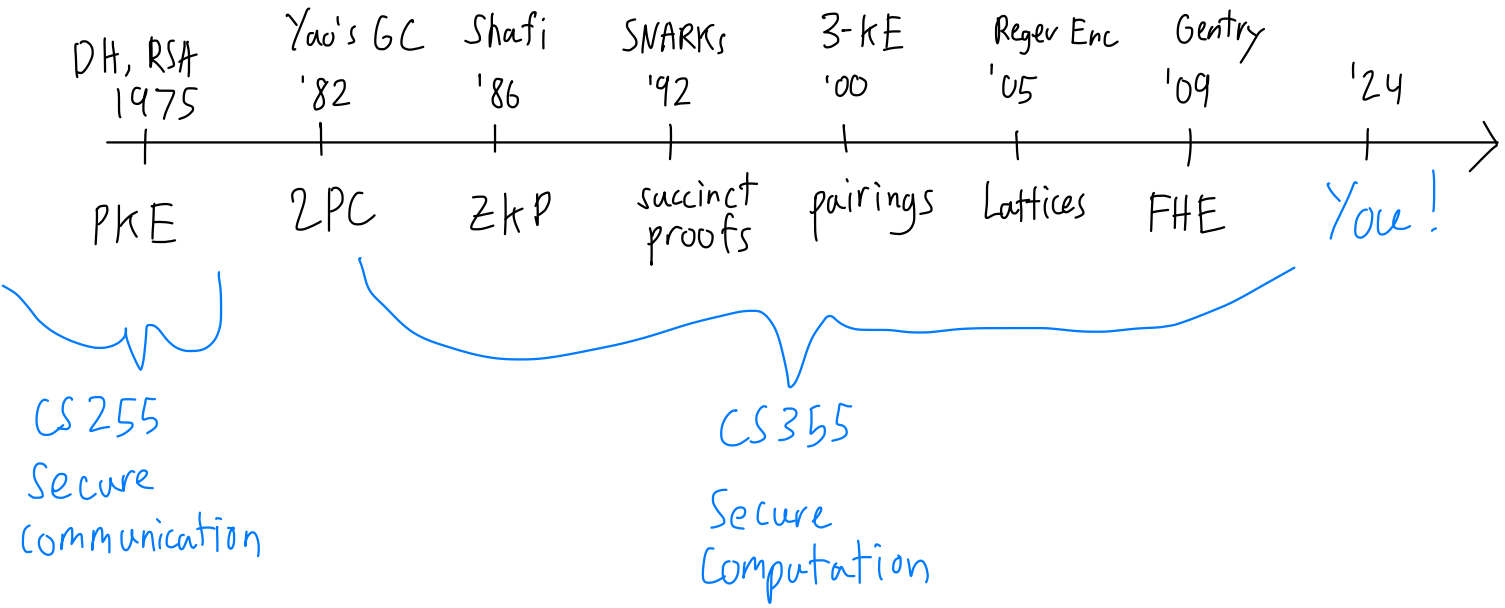
Be your first advanced crypto course

- Learn techniques and formalism
- Understand Open Research Problems
- Prepare you for research

Be your last advance crypto course

- Applications of cutting edge crypto
- Digest new papers
- Prepare you to use crypto to change the world!

Historical Perspective



Topics

1) Foundations

2) Cryptanalysis

3) Elliptic Curves

4) Zero Knowledge

5) Multiparty Computation

6) Lattice-Based Cryptography (FHE)

Why take CS355?

- About the balance of Power & Privacy
- Widely deployed
- Beautiful Math!

Why NOT take CS355?

- Assume Mathematical maturity
  - Shouldn't be your first proof-based course
    - Possible Prereqs: CS255, 265, 254, MATH 120, ...
  - Time consuming

CS 355 = CS255 + 100*

Logistics

Website: cs355.stanford.edu (make sure you're on 24sp)

Contact:
- Ed main communication, don't spoil problems!
- Anonymous Feedback Form (staff page)
- cs355@cs.stanford.edu (for personal questions)

Lectures:
- In person, not recorded
- Notes available after class
- No textbooks (supplemental readings online)
  (cryptobook.us - Boneh & Shoup)

Office Hours:
- schedule online
- group work environment (w/ Prof / TAs)

Problem Sets

- Every 2 weeks (5 total, each 20% of total grade)
- due Fridays, 6pm on Gradescope, Latex!
- 1 is out! Due April 12th
- Do not search for solutions (Math lemmas fine)
- Work Together, but Write Solo (list collaborators)
- 3 late days total, at most 1 late day for HW5
- NO Exams !!

Hardness & Reductions

tunable security parameter ↓

- Honest Parties are efficient (polynomial in λ)
- Attacks are inefficient

Ex) Decryption with key is eff, w/o key it's intractable

- Security ~~Proofs~~ Reductions

Assumption Hard Problem $\Rightarrow$ Secure System

Eff Solution to problem $\Leftarrow$ Eff Attack on System

Ex) factoring, d-log, DDH, CDH, AES is a secure block cipher

Today & Tomorrow: Build all of symmetric crypto from 1 minimal assumption!

"One Way Functions"

OWFs $\Rightarrow$ PRG_{+1} $\Rightarrow$ PRG $\Rightarrow$ PRF $\Rightarrow$ Block Ciphers

Today :) Tomorrow Bonus Day

OWFs!

- Let $X_\lambda := \{X_1, X_2, \ldots\}$ and $Y_\lambda := \{Y_1, Y_2, \ldots\}$ be families of finite sets indexed by security parameter λ.

(Input space; Output Space)

- Let $f : X_\lambda \to Y_\lambda$ be a deterministic, poly(λ) time algorithm.

(usually omit λ)

Intuition: f is easy to compute but hard to invert

Def: $f : X_\lambda \to Y_\lambda$ is a one way function if for all efficient algorithms (probabilistic, poly(λ) time) A,

$$\Pr_{A,x}\left[f\left(A(f(x))\right) = f(x) \;:\; x \xleftarrow{r} X_\lambda \right] \leq \text{negl}(\lambda)$$

Q: Why phrase it this way?

A function $g(\lambda)$ is $\underline{\text{negl}(\lambda)}$ if $\forall c \in \mathbb{N}$, $g(\lambda) \in O(\lambda^{-c})$.

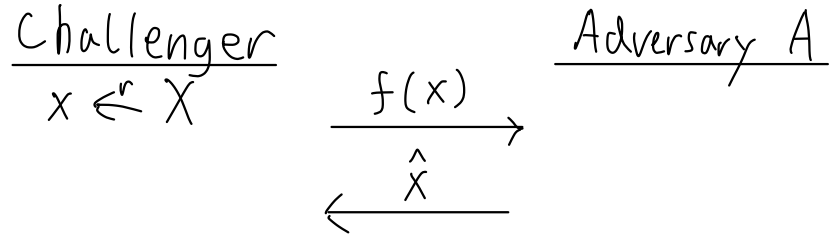
Ex) $2^{-\lambda} < 2^{-\lambda/2} < 2^{-\sqrt{\lambda}} \leq \text{negl}(\lambda)$

$\lambda^{\log_\lambda 2^{-\lambda}} = \lambda^{-\lambda \log_\lambda 2} = \lambda^{-2\frac{\lambda}{\log\lambda}}$ ← super constant

Facts) • $f \leq \text{negl}$, $g \leq \text{poly}$ $\to$ $fg \leq \text{negl}$

• $f \not\leq \text{negl}$, $g \leq \text{poly}$ $\to$ $f/g \not\leq \text{negl}$

As a Security Game:



$$\text{OWFadv}[A,f] = \Pr_{A,x}\left[f(x) = f(\hat{x})\right]$$

f is OWF if $\forall$ PPT A, $\text{OWFadv}[A,f] \leq \text{negl}(\lambda)$.

Q: Can f be a OWF if $X_\lambda = \{0,1\}^{\log \lambda}$?

Ans: No! Regardless of f, consider A that outputs $\hat{x} \in X_\lambda$.
It has advantage $\frac{1}{\lambda} \notin \text{negl}(\lambda)$

Candidate OWFs

1) $f(x,y) := xy$ x,y are λ-bit primes (factoring)

2) $f_{p,g}(x) := g^x \pmod p$ p is λ-bit prime, $g \in \mathbb{Z}_p$, $x \in \mathbb{Z}_{p-1}$
(finite field dlog)

3) $f(x_1, \ldots, x_n, S \subseteq [n]) = (x_1, \ldots, x_n, \sum_{i \in S} x_i)$ (subset sum)

4) Levin's universal OWF: f_L s.t. $\exists$ OWF $\Rightarrow f_L$ is OWF

Pseudo-Random Generators

A deterministic, efficient function $G: S_\lambda \to R_\lambda$

- $S_\lambda = \{0,1\}^\lambda$, $R_\lambda = \{0,1\}^{\ell(\lambda)}$ with $\ell(\lambda) \geq \lambda + 1$
stretch!

Intuition: If $\downarrow$ is random, then $\downarrow$ should look random.

Def: G is a PRG if $\forall$ PPT A, (output $\in \{0,1\}$)

$$\left| \Pr_{A,r,s}[A(G(s)) = 1 : s \xleftarrow{r} S_\lambda] - \Pr_{A,r,s}[A(r) = 1 : r \xleftarrow{r} R_\lambda] \right| \leq \text{negl}(\lambda)$$

$\text{PRGadv}[A, G]$

OWF $\rightarrow$ PRG$_{\ell(\lambda)=\lambda+1}$

Let $X_\lambda = Y_\lambda = \{0,1\}^\lambda$. (length preserving)

First Attempt!

- Define $g(x) := x_1 \| f(x)$ where $f: X_\lambda \rightarrow Y_\lambda$ is OWF
- Not necessarily a PRG, but g is a OWF that leaks the first bit of input.

Proof: (First Proof!)

Suppose g is not a OWF. Then, by def of OWF, $\exists$ PPT A, s.t. $\Pr[g(A(g(x))) = g(x)] \notin \text{negl}(\lambda)$.

We will construct a PPT B, s.t. $\Pr[f(B(f(x)) = f(x)] \notin \text{negl}(\lambda)$, which contradicts f being a OWF.

$B(y)$: Sample $b \xleftarrow{r} \{0,1\}$, output $A(b \| y)$.

$$\begin{aligned}
&\Pr[f(B(f(x))) = f(x)] \\
&\geq \Pr[f(B(f(x))) = f(x) \wedge b = x_1] \\
&= \Pr[f(B(f(x))) = f(x) \mid b = x_1] \cdot \Pr[b = x_1] \\
&= \Pr[f(A(x_1 \| f(x))) = f(x)] \cdot \tfrac{1}{2} \\
&\geq \Pr[g(A(g(x))) = g(x)] \cdot \tfrac{1}{2} \notin \text{negl}(\lambda)
\end{aligned}$$

$\uparrow$ $\notin \text{negl}(\lambda)$

$\square$

Why is g not necessarily a PRG?

- x_1 may be easy to guess, but all the bits can't be easy to guess! otherwise, f would not be OWF.

Can we derive a hard bit?

Def: An efficient predicate $b: \{0,1\}^\lambda \to \{0,1\}$ is a hardcore bit for a OWF $f: \{0,1\}^\lambda \to Y_\lambda$ if for all PPT A,

$$\Pr[A(f(x)) = b(x)] \leq \frac{1}{2} + \text{negl}(\lambda)$$

not much better than random guessing

equivalently ⇕

$$\forall \text{ PPT } A, \left| \Pr[A(f(x)) = b(x)] - \frac{1}{2} \right| \leq \text{negl}(\lambda)$$

$\text{HCB}_{\text{adv}}[A, b, f]$

Theorem [Goldreich-Levin '89]: "Every OWF has a HCB."

- Idea: random linear combination of the bits of x should be hard to compute.
- Let f be a OWF. Define

$$f'(x, r) := f(x) \| r$$

$$b(x, r) := \sum_i x_i \cdot r_i \pmod 2$$

- Then, f' is a OWF with HCB b.

You'll show a weaker version of Thm in HW 1.

Now: OWP + HCB → PRG_{+1}

Easier to show for Lecture. For full construction, search "A Pseudorandom Generator from any One-way Function"

Construction: Define

$$G(x) = f(x) \,||\, b(x)$$

Theorem: If $f: \{0,1\}^\lambda \to \{0,1\}^\lambda$ is a one way permutation w/ hardcore bit b, then $G: \{0,1\}^\lambda \to \{0,1\}^{\lambda+1}$ is a PRG.

Proof:

Suppose G is not a PRG, then $\exists$ PPT A s.t. $\text{PRGadv}[A,G] \notin \text{negl}(\lambda)$.

① $$\left| \Pr[A(G(s)) = 1 : s \xleftarrow{r} \{0,1\}^\lambda] - \Pr[A(r) = 1 : r \xleftarrow{r} \{0,1\}^{\lambda+1}] \right| \geq \varepsilon(\lambda)$$

($G(s) = f(s) \,||\, b(s)$)

for $\varepsilon(\lambda) \notin \text{negl}(\lambda)$

Overview:

1) Show A notices if we flip b:

$$\frac{1}{2} \left| \Pr[A(f(s) \,||\, b(s)) = 1] - \Pr[A(f(s) \,||\, \overline{b(s)}) = 1] \right| \geq \varepsilon(\lambda)$$

2) Then, build PPT B that predicts HCB $b(s)$.

Step 1)

$$\Pr[A(r) = 1] = \Pr\left[A(y \,||\, b') = 1 : \begin{matrix} y \xleftarrow{r} \{0,1\}^\lambda \\ b' \xleftarrow{r} \{0,1\} \end{matrix}\right]$$

$$= \Pr[A(f(s) \,||\, b') = 1]$$

(f: permutation)

$$= \Pr[A(f(s)\|b')=1 \mid b'=b(s)] \cdot \Pr[b'=b(s)]$$

$$+ \Pr[A(f(s)\|b')=1 \mid b'=\overline{b(s)}] \cdot \Pr[b'=\overline{b(s)}]$$

$$= \frac{1}{2}\left(\Pr[A(f(s)\|b(s))=1] + \Pr[A(f(s)\|\overline{b(s)})=1]\right)$$

By substitution into ①, we obtain

$$\left|\Pr[A(\underbrace{G(s)}_{\uparrow f(s)\|b(s)})=1] - \frac{1}{2}(\Pr[\ldots] + \Pr[\ldots])\right| \geq \varepsilon(\lambda)$$

② $$\frac{1}{2}\left|\Pr[A(f(s)\|b(s))=1] - \Pr[A(f(s)\|\overline{b(s)})=1]\right| \geq \varepsilon(\lambda)$$

Step 2) Define $B(y)$

- $c \xleftarrow{r} \{0,1\}$
- If $A(y\|c)=1$: output c
  Else: output $\bar{c}$

Analysis:

$$\left|\Pr[B(f(s))=b(s)] - \frac{1}{2}\right|$$

$$= \left|\frac{1}{2}\Pr[B(f(s))=b(s) \mid c=b(s)] + \frac{1}{2}\Pr[B(f(s))=b(s) \mid c=\overline{b(s)}] - \frac{1}{2}\right|$$

$$= \left|\frac{1}{2}\Pr[A(f(s)\|b(s))=1] + \frac{1}{2}\Pr[A(f(s)\|\overline{b(s)})=0] - \frac{1}{2}\right|$$

$$= \left|\frac{1}{2}\Pr[A(f(s)\|b(s))=1] + \frac{1}{2}(1-\Pr[A(f(s)\|\overline{b(s)})=1]) - \frac{1}{2}\right|$$

$$= \frac{1}{2}\left|\Pr[A(f(s)\,||\,b(s))=1] - \Pr[A(f(s)\,||\,\overline{b(s)})=1]\right| \overset{②}{\geq} \varepsilon(\lambda)$$

Therefore, $HCBadv[B,b,f] \geq \varepsilon(\lambda) \notin negl(\lambda)$ □

✡ In context of Goldreich-Levin:

Recall GL HCB: $b(x,r) = \sum_i x_i \cdot r_i \pmod 2$

Assume f is OWP, then GL PRG is

$$G(x,r) := f(x)\,||\,r\,||\,b(x,r)$$

$(x,r) \xleftarrow{r} \{0,1\}^{2n}$; f' is OWP ; HCB

$2n$ bits $\rightarrow$ $2n+1$ bits !

$+1$