EECS 575: Advanced Cryptography
Fall 2021
Lecture 3

Paul Grubbs

paulgrub@umich.edu

Beyster 4709

Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

Announcements

- Homework 2 was released on Monday at 10pm
  - Due 9/22
- Lecture topic vote

Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

Recap from last time

Model of bounded computation:

- Non-uniform → advice strings
- probabilistic → flip coins
- poly-time → $O(n^c)$

Negligible functions:

$\nu(n)$ is negligible if $\nu(n) = o(n^{-c})$ for ALL $c > 0$

One-Way Functions

$f : \{0,1\}^n \to \{0,1\}^n$ is a OWF if:

- Easy to compute:
  poly-time
  $\exists$ uniform, deterministic algorithm F
  s.t. $F(x) = f(x)$ $\forall x$

- Hard to invert
  $\forall$ nuppt $\mathcal{I}$. the Advantage of $\mathcal{I}$

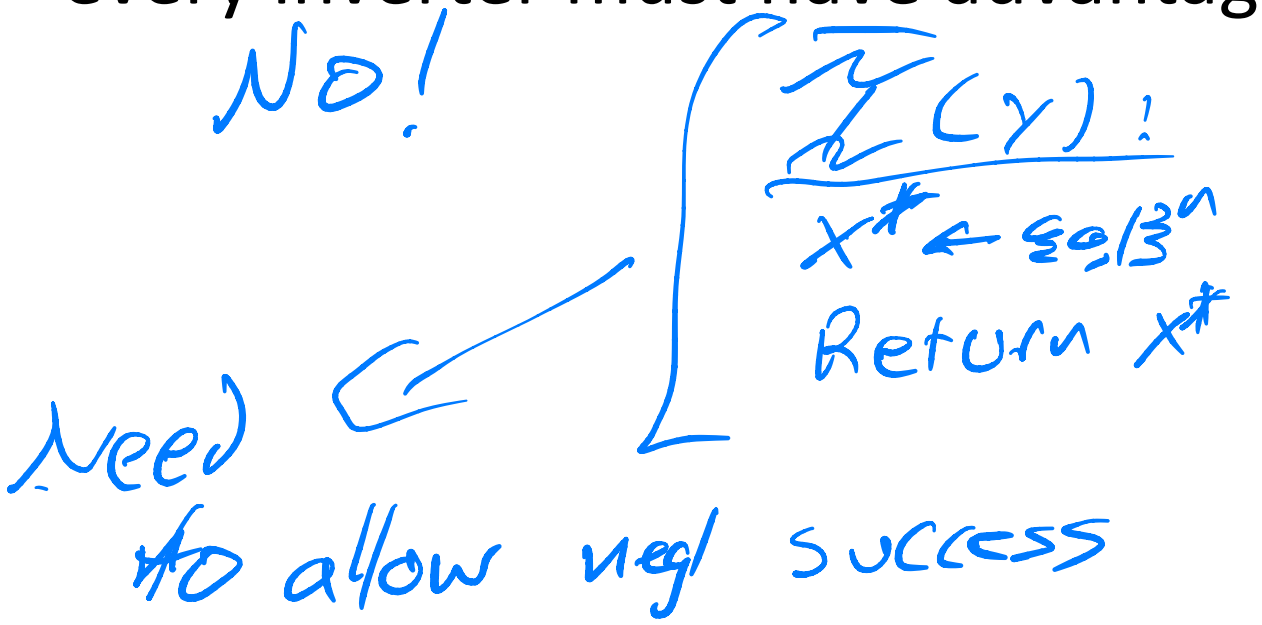
$$\underline{Adv(\mathcal{I})} = \Pr_{x \leftarrow \{0,1\}^n}\left[\mathcal{I}(1^n, f(x)) \in f^{-1}(f(x))\right] = negl(n)$$

- Average-case hardness

- unary encoding of n

Question about One-Way Functions

Is it possible to build a “perfect” OWF, where every inverter must have advantage 0?



Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

Proofs by reduction

What is a reduction? why do we need them?

Existence of most crypto relies (at least) on $P \neq NP$

Proofs by reduction

Let $f : \{0,1\}^n \to \{0,1\}^n$ be a OWF

Define $f' : \{0,1\}^n \to \{0,1\}^{2n}$ as

Thm: f' is a OWF. $\{f'(x) = (f(x), f(f(x)))\}$

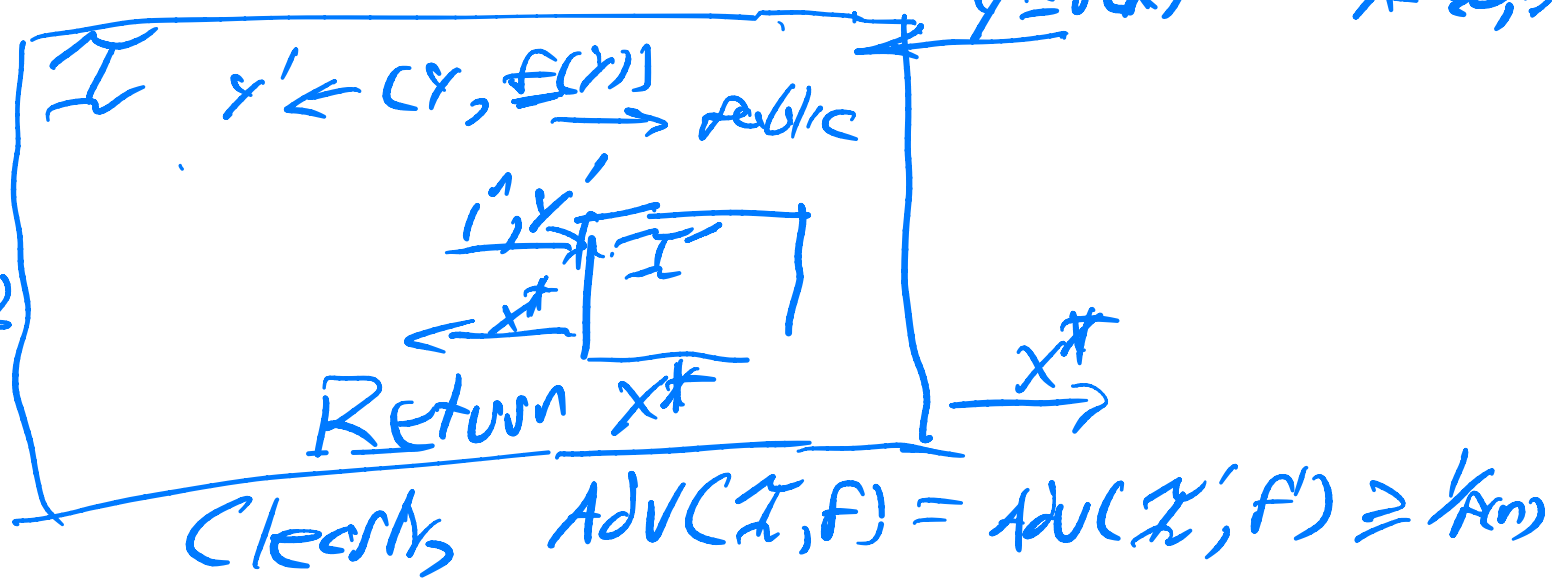
Proof idea: contrapositive

If f is a OWF, then f' is a OWF

If f' is not a OWF, then f is NOT a OWF

Graphically:

Assume $\mathcal{I}'$ so that $Adv(\mathcal{I}', f') > 1/p(n)$



Clearly, $Adv(\mathcal{I}, f) = Adv(\mathcal{I}', f') \geq 1/p(n)$

Our first reduction (!)

Take-Home Exercise

If f is a OWF, must $f'(x_1, x_2) = (f(x_1), f(x_2))$ also be a OWF?

Prove by reduction

Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

A Candidate One-Way Function

Multiplication

$F_{mult} : \mathbb{N}^2 \to \mathbb{N}$

$F_{mult}(x, y) = \begin{cases} 1 & \text{if } x = 1 \text{ or } y = 1 \\ x \cdot y & \text{o/w} \end{cases}$

Inverting f_{mult} is factoring integers.

$$\Pr_{\substack{x \leftarrow [1, 2^n] \\ y \leftarrow [1, 2^n]}} \left[\mathcal{I}(1^n, x \cdot y) = (x, y) \right]$$

$\forall y \in \mathbb{N}, F_{mult}^{-1}(y) = (1, y)$

rules out

A Candidate One-Way Function

Multiplication

$$\Pr_{x \leftarrow [1, 2^n],\ y \leftarrow [1, 2^n]} \left[A(1^n, x \cdot y) = (x, y) \right]$$

If x or y even, $(2, \frac{xy}{2})$ is a preimage!

Happens w.p. $3/4$

Not a OWF!

Fixing f_{mult}

Idea: Change input dist to "focus" on hard inputs.

$\Pi_n = \{p : p \in [1, 2^n] \text{ is prime}\}$

Conjecture:

If nuppt $\mathcal{I}$,

$\Pr_{\substack{p \leftarrow \Pi_n \\ q \leftarrow \Pi_n}}[\mathcal{I}(1^n, p \cdot q) = (p, q)] = \text{negl}(n)$

Sampling Random Primes

Evaluating f_{mult} needs sampling random elements of Π_n

How do we do this?

Check if p is prime?

Deterministic PPT: AKS '02

- Randomized tests (Miller-Rabin,
- Deterministic, based on conjecture

Algorithm $\text{GenPrime}(1^n)$:

1. Choose uniform integer $p \in [1, 2^n]$
2. If p prime output p. Else Goto 1

What is runtime of this alg?

Sampling Random Primes

What is runtime? Define $\pi(N) = \#\text{primes} \le N$

Lemma (Chebyshev):

$$\text{If } N > 1, \quad \pi(N) > \frac{N}{2\log_2 N}$$

Probability of sampling an n-bit prime:

$$\frac{|\Pi_A|}{2^n} = \frac{\pi(2^n)}{2^n} \overset{\text{Chebyshev}}{>} \frac{2^n}{2^n \cdot 2\log_e(2^n)} = \frac{1}{2n}$$

GenPrime(1^n) terminates in poly-time (Exercise)

Agenda for this lecture

- Announcements
- One-way functions (OWFs)
- Proof by reduction
- Candidate OWFs based on factoring
- Amplifying weak OWFs (if time)

Amplifying Weak OWFs