# Hash

HW5 out: due next Mon
proj topic due next Wed

## Difficulty of Hash Collisions

$H: \{0,1\}^* \longrightarrow \{0,1\}^n$

how long to find a collision?

pigeonhole: $2^n$ $(+1)$ to guarantee collision

birthday bound: $\sim \sqrt{2^n} = 2^{n/2}$ to get good ($\sim 50\%$) probability

Ex: $B: \{\text{people}\} \longrightarrow \{1, \ldots, 365\}$

how long to find 2 different people $B(x) = B(x')$

pigeonhole: $365 + 1$ to guarantee collision

pigeonhole: $\sqrt{365}$

$$\Pr[\text{collision among 18 people}]$$

$$= 1 - \left(1 - \frac{1}{365}\right)\left(1 - \frac{2}{365}\right) \cdots \left(1 - \frac{17}{365}\right)$$

$$0.632 \frac{18 \cdot 17}{2 \cdot 365} \leq \Pr \leq \frac{18 \cdot 17}{2 \cdot 365}$$

Ex: use H w/ 128-bits of output

$\Rightarrow$ attack on collision resistance costs $2^{64}$

**Ex:** Bitcoin hash rate:

23 billion TH/s

$= 23 \times 10^9 \times 10^{12}$ hashes/sec

$= 23 \times 10^{21}$

$= 2^{74}$ hashes/sec

$\log_2(x) = \log(x)/\log(2)$

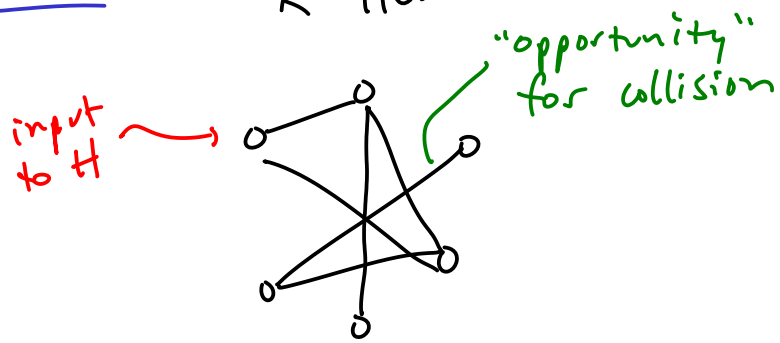
**Q:** brute force birthday attack on hash function:
try $\sim 2^{n/2}$ things, see if <u>any 2</u> collide

**HW:** find 2 values that collide under H w/ 40 bits of output

- "mikero | ??????"
- "?????? | 20180226"

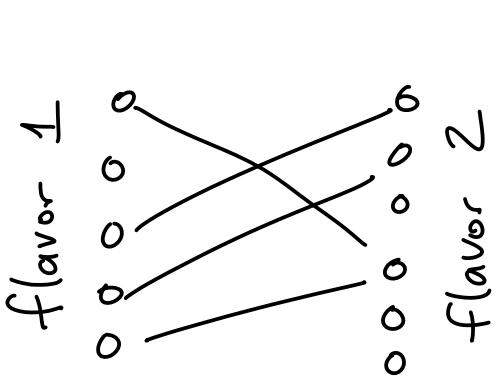
$2^{40}$ takes longer than due date of HW

**hint:**




$\binom{k}{2} \approx k^2$ edges

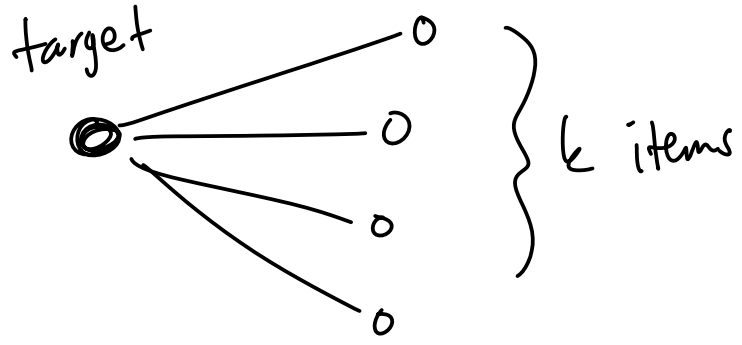
$\Rightarrow \approx \frac{k^2}{2^n}$ chance of collision




$k^2$ edges

$\Rightarrow \approx \frac{k^2}{2^n}$ chance of collision

# target / 2nd preimage collisions




so prob. of collision is $k/2^n$

$\Rightarrow$ cost $\sim 2^n$ to break target collision or 2nd preimage

## Application:

use HF to build signature scheme

- prove security of signature scheme using collision-resistance of H
  - $\Rightarrow$ H must have 256 bits output in practice
- prove security of signature scheme using second preimage sec. of H
  - $\Rightarrow$ H can have 128 bits output
  - $\Rightarrow$ signatures now half the length!!

## Hash Family

$\mathcal{H}$ = set of many hash functions

each application of hashing, choose $H \leftarrow \mathcal{H}$

make $H$ public

use $H(x)$ in your application

## Salt

$H$ = SHA256

for each application choose random string $s \leftarrow \{0,1\}^\lambda$ (SALT)

make $s$ public

use $\text{SHA256}(s \| x)$ in your application