Birthday Paradox

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRI CITY

Outline

1. Birthday Paradox
2. Birthday Attack on Hash and DLP
3. Proof-of-Work

Hash Function

A $(m, H(m))$ m' $H(m) = H(m')$

- A hash function $H: \{0,1\}^* \rightarrow \{0,1\}^n$, where *n* is a fixed, defined as h = H(x) satisfy the properties:

  - **Pre-image resistance:** Given h, computing *x* is hard.

  - **Second pre-image resistance:** Given an input *x*, difficult to find *y (≠x)* such that *H(x) = H(y)*.

  - **Collision resistance:** Difficult to find a pair *(x, y)* of two different messages such that *H(x) = H(y)*.

  - Note that collisions may be found by a birthday attack

Attacks on Hash Functions

- Brute-force attacks and Cryptanalysis
- A preimage or second preimage attack
  - find x s.t. $H(x)$ equals a given hash value
- Collision resistance
  - find two messages x & y with same hash, that is,

$$H(x) = H(y)$$

- Hence, values $2^{m/2}$ determines strength of hash code against brute-force attacks
  - 128-bits inadequate, 160-bits suspect

Birthday Attacks

- Birthday paradox
  - In a group of **23** randomly chosen people, at least two will share a birthday with probability at least **50%**.
  - If there are **30**, the probability is around **70%**.
  - Finding two people with the same birthday is the same thing as finding a collision for this particular hash function.

D_i : No collision after having thrown in the i-th day

d_1	d_2	d_3	$\cdots$	d_{364}	d_{365}

$\{b_1, b_2, \ldots b_{23}\}$

$\Pr[\text{There is a collision}]$
$= 1 - e^{r^2/2N}$

$D1$: b_1 thrown, $\quad b_1 \to \{d_1, d_2, \ldots d_{365}\}$

$D2$: b_2 thrown, $\quad b_2 \to \{d_1, d_2, \ldots d_{365}\} \setminus \{b_1\}$

$D3$: b_3 thrown, $\quad b_3 \to \{d_1, d_2, \ldots d_{365}\} \setminus \{b_1, b_2\}$

$$\boxed{P[D_{i+1} \mid D_i] = \frac{N-i}{N} = 1 - \frac{i}{N}}$$

$$P[D_r] = P[D_r \mid D_{r-1}]\, P[D_{r-1}] = \prod_{i=1}^{r-1} P[D_{i+1} \mid D_i]$$

$$= \prod_{i=}^{r-1} \left(1 - \frac{i}{N}\right) \approx e^{-r^2/2N}.$$

Birthday Attacks

- The probability that all 23 people have **different birthdays** is

$$1\times(1-\frac{1}{365})(1-\frac{2}{365})\ldots(1-\frac{22}{365}) = 0.493$$

$e^{-\log_e 2}$

$= e^{\log_e 2^{-1}}$

$= 1/2$

Therefore, the probability of at least two having the same birthday is 1- 0.493=0.507

- More generally, suppose we have N objects, where N is large. There are r people, and each chooses an object. Then

$$P(\text{there is a match}) \approx 1-e^{-r^2/2N}$$

Birthday Attacks

- Choosing $r^2/2N = \ln 2$
- we can find that if $r \approx 1.177\sqrt{N}$, then the probability is 50% that at least two people choose the same object.

- If there are N possibilities and we have a list of length $\sqrt{N}$, then there is a good chance of a match.

- If we want to increase the chance of a match, we can make a list of length of a constant times $\sqrt{N}$.

Birthday Attack on DLP

A birthday attack on discrete logarithm

- We want to solve $\alpha^x \equiv \beta$ (mod p).
- Make two lists, both of length around $\sqrt{p}$

  1st list: α^k (mod p) for random k.

  2nd list: $\beta\alpha^{-h}$ (mod p) for random h.
- There is a good chance that there is a match

  $\alpha^k \equiv \beta\alpha^{-h}$ (mod p), hence $x=k+h$.

The birthday attack algorithm is probabilistic.

x

$\boxed{\alpha^k = \beta\alpha^{-h}}$

$\alpha^k = \alpha^x \alpha^{-h}$

α^{k+h}

$\Rightarrow \alpha^x = \alpha^{k+h}$

$\Rightarrow x = k+h$

n-bit.

$p = 2^n$

$\sqrt{p} = 2^{n/2}$

Birthday Attacks on Hash

- User prepared to sign a valid message x
- Opponent generates $2^{m/2}$ variations x’ of x, all with essentially the same meaning, and saves them
- Opponent generates $2^{m/2}$ variations y’ of a desired fraudulent message y
- Two sets of messages are compared to find pair with same hash (probability > 0.5 by birthday paradox)
- User sign the valid message, then substitute the forgery which will have a valid signature

Birthday Attacks

$\sqrt{N}$, 2 lists.

- Suppose there are N objects and there are two groups of r people. Each person from each group selects an object. What is the probability that someone from the first group choose the same object as someone from the second group?

$$P(\text{there is a match between two groups})$$

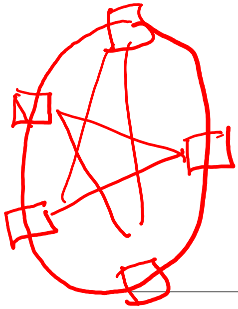
$$= 1 - e^{-r^2/N}$$

?? Solution ??

- Eg. If we take N=365 and r=30, then

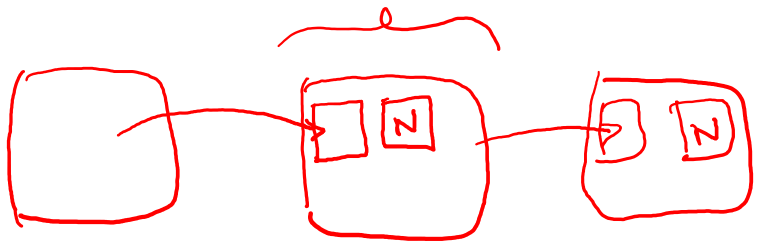
$$P(\text{there is a match between two groups})$$

$$= 1 - e^{-30^2/365} = 0.915$$



Proof of Work

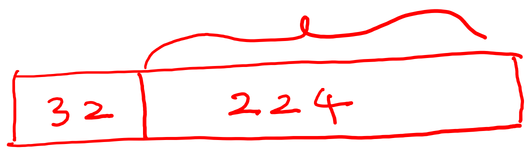
(Bitcoin Mining)



D - block data

N - nonce ??

K - no. of leading bits in hash as zero.



SHA-256

32 bits - zero

$$H(D,N) < 2^{n-k}$$

choose

$$P\left[H(D,N) < 2^{n-k}\right] = \frac{2^{n-k}}{2^{n}} = \frac{1}{2^{k}}$$