3 HASH FUNCTIONS

CRYPTO 101: Building Blocks

©Alfred Menezes

cryptography101.ca

V3 outline

- V3a: Fundamental concepts
- V3b: Relationships between PR, 2PR, CR
- V3c: Generic attacks
- V3d: Iterated hash functions
- V3e: SHA-256

 © *Alfred Menezes*

V3a
Fundamental concepts

HASH FUNCTIONS

CRYPTO 101: Building Blocks

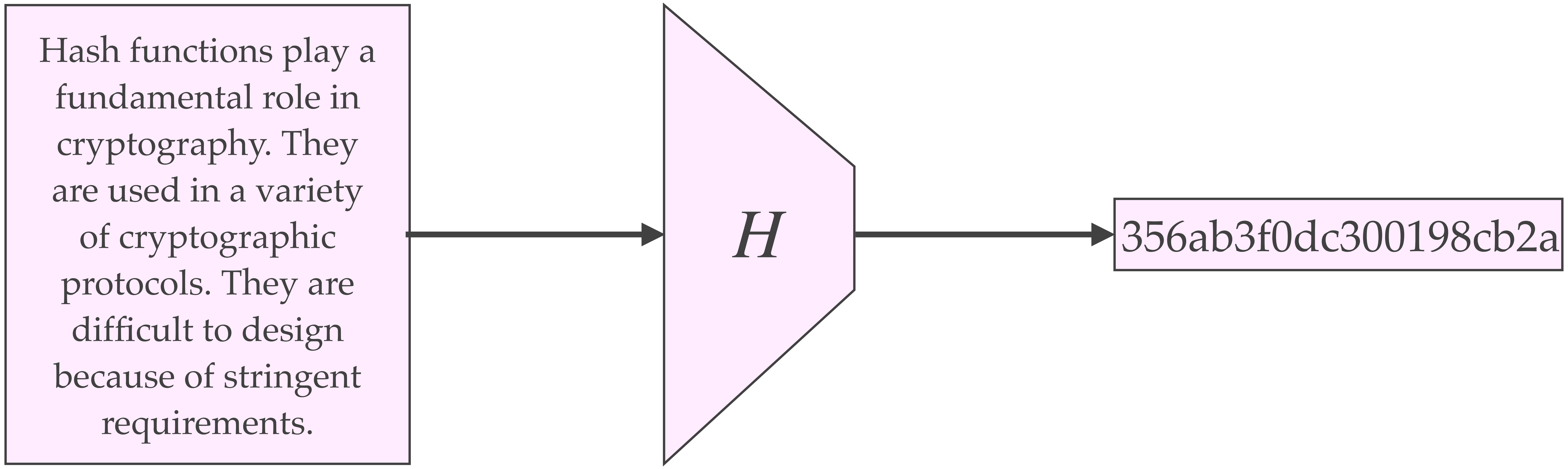
©Alfred Menezes

cryptography101.ca

Definitions and terminology

- Hash functions play a fundamental role in cryptography
- They are used in a variety of cryptographic primitives and protocols.
- They are very difficult to design because of stringent security and performance requirements.
- The most commonly used hash functions are:
  - SHA-1
  - SHA-2 family: SHA-224, SHA-256, SHA-384, SHA-512
  - SHA-3 family

What is a hash function?



See:
www.xorbin.com/tools/md5-hash-calculator (MD5)
www.xorbin.com/tools/sha1-hash-calculator (SHA-1)
www.xorbin.com/tools/sha256-hash-calculator (SHA-256)

Example: SHA-256

SHA-256 : $\{0,1\}^* \longrightarrow \{0,1\}^{256}$

SHA-256(“Hello there”) =

`0x4e47826698bb4630fb4451010062fadbf85d61427cbdfaed7ad0f23f239bed89`

SHA-256(“Hello There”) =

`0xabf5dacd019d2229174f1daa9e62852554ab1b955fe6ae6bbbb214bab611f6f5`

Definition of a hash function

A hash function is a mapping H such that:

1. H maps binary messages of arbitrary lengths $\leq L$ to outputs of a fixed length n:
   $H : \{0,1\}^{\leq L} \to \{0,1\}^n$. ($L$ is usually large, e.g., $L = 2^{64}$, whereas n is small, e.g. $n = 256$.)
2. $H(x)$ can be efficiently computed for all $x \in \{0,1\}^{\leq L}$.

- H is called an n-bit hash function. $H(x)$ is called the hash or message digest of x.
- Notes:
  - The description of a hash function is public; there are no secret keys.
  - For simplicity, we will usually write $\{0,1\}^*$ instead of $\{0,1\}^{\leq L}$.
  - More generally, a hash function is an efficiently computable function from a set S to a set T.

© *Alfred Menezes*

Toy hash function

x	$H(x)$	x	$H(x)$	x	$H(x)$	x	$H(x)$
0	00	1	01				
00	11	01	01	10	01	11	00
000	00	001	10	010	11	011	11
100	11	101	01	110	01	111	10
0000	00	0001	11	0010	11	0011	00
0100	01	0101	10	0110	10	0111	01
1000	11	1001	01	1010	00	1011	01
1100	10	1101	00	1110	00	1111	11

$H : \{0,1\}^{\leq 4} \longrightarrow \{0,1\}^2$

- (00,1000) is a **collision**.
- 1001 is a **preimage** of 01.
- 10 is a **second preimage** of 1011.

© *Alfred Menezes*

Some applications of hash functions

- Hash functions are used in all kinds of applications, including some that they were not designed for.
- One reason for this widespread use of hash functions is speed.

© Alfred Menezes

Preimage resistance (PR)

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is preimage resistant if, given a hash value $y \in_R \{0,1\}^n$, it is computationally infeasible to find (with non-negligible success probability) *any* $x \in \{0,1\}^*$ with $H(x) = y$. (x is called *a* preimage of y.)

Password protection on a multi-user computer system:

- The server stores [userid, H(password)] in a password file.
- If an attacker obtains a copy of the password file, she does not learn any passwords.
- This application requires preimage resistance.

 © *Alfred Menezes*

2nd preimage resistance (2PR)

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is 2nd preimage resistant if, given $x \in_R \{0,1\}^*$, it is computationally infeasible to find (with non-negligible success probability) *any* $x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$.

Modification Detection Codes (MDCs):

- To ensure that a message m is not modified by unauthorized means, one computes $H(m)$ and protects $H(m)$ from unauthorized modification.
- This is useful in malware protection.
- This application requires 2nd preimage resistance.

 © *Alfred Menezes*

Collision resistance (CR)

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is collision resistant if it is computationally infeasible to find (with non-negligible success probability) $x, x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$. Such a pair (x, x') is called a collision for H.

Message digests for digital signature schemes:

- For reasons of efficiency, instead of signing a (long) message x, the (much shorter) message digest $h = H(x)$ is signed.
- This application requires preimage-resistance, 2nd preimage resistance, and collision resistance.
- To see why collision resistance is required, suppose that the legitimate signer Alice can find a collision (x_1, x_2) for H. Alice can sign x_1 and later claimed to have signed x_2.

© *Alfred Menezes*

Some other applications of hash functions

1. Message Authentication Codes: HMAC.
2. Pseudorandom bit generation:
   Distilling random bits $s = H(x_1, x_2, \ldots, x_t)$ from several "pseudorandom" sources $x_1, x_2, \ldots, x_t$.
3. Key derivation functions (KDF):
   Deriving a cryptographic key from a secret.
4. Proof-of-work in cryptocurrencies (Bitcoin).
5. Quantum-safe signature schemes.

© *Alfred Menezes*

V3b
Relationships between PR, 2PR and CR

HASH FUNCTIONS

CRYPTO 101: Building Blocks

©Alfred Menezes

cryptography101.ca

Typical cryptographic requirements

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is preimage resistant if, given a hash value $y \in_R \{0,1\}^n$, it is computationally infeasible to find (with non-negligible success probability) *any* $x \in \{0,1\}^*$ with $H(x) = y$.

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is 2nd preimage resistant if, given $x \in_R \{0,1\}^*$, it is computationally infeasible to find (with non-negligible success probability) *any* $x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$.

Definition: A hash function $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is collision resistant if it is computationally infeasible to find (with non-negligible success probability) $x, x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$.

 © *Alfred Menezes*

Breaking PR, 2PR, CR

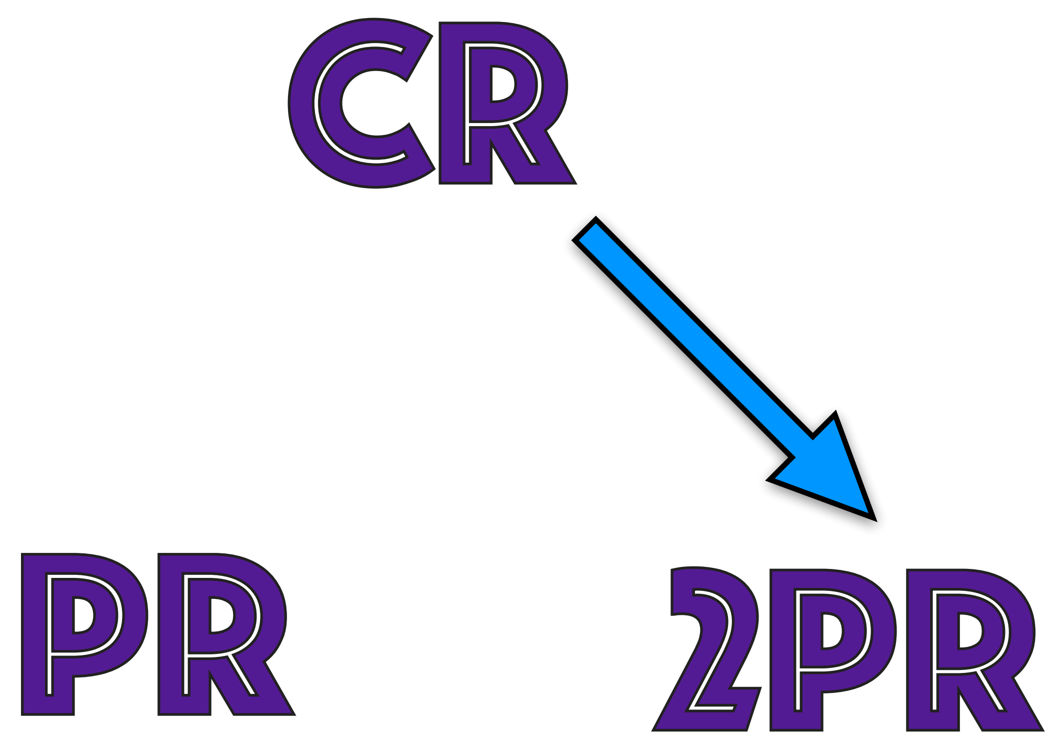
Breaking PR:
Given: $y \in_R \{0,1\}^n$.
Required: $x \in \{0,1\}^*$ with $H(x) = y$.

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

Breaking 2PR:
Given: $x \in_R \{0,1\}^*$.
Required: $x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$.

Breaking CR:
Given: $-$.
Required: $x, x' \in \{0,1\}^*$ with $x' \neq x$ and $H(x') = H(x)$.

Claim 1: If H is CR, then H is 2PR



Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is not 2PR.

We'll show that H is not CR.

Select $x \in_R \{0,1\}^*$. Since H is not 2PR, we can efficiently find $x' \in \{0,1\}^*$, $x' \neq x$, with $H(x') = H(x)$.

Thus, (x, x') is a collision for H that we have efficiently found, showing that H is not CR. $\square$

Note: The proof established the *contrapositive* statement.

© *Alfred Menezes*

Claim 2: CR does not guarantee PR

Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is CR.

Consider the hash function $\overline{H} : \{0,1\}^* \longrightarrow \{0,1\}^{n+1}$ defined by

$$\overline{H}(x) = \begin{cases} 0\|H(x), & \text{if } x \notin \{0,1\}^n \\ 1\|x, & \text{if } x \in \{0,1\}^n. \end{cases}$$

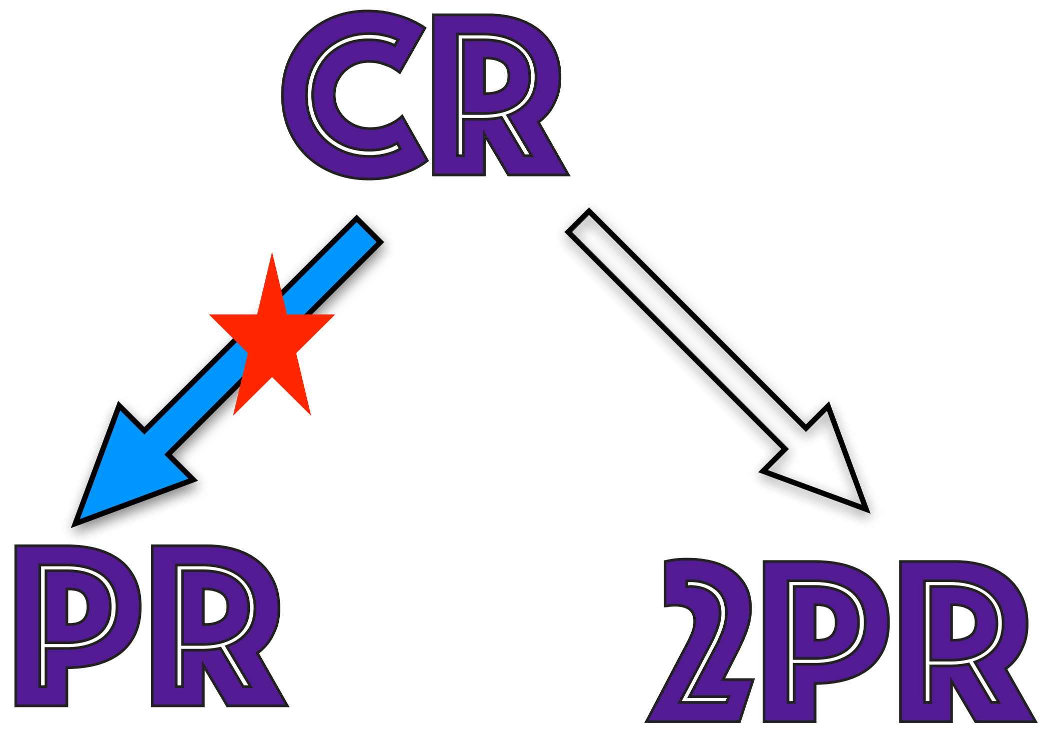
Then $\overline{H}$ is CR (since H is).

And, $\overline{H}$ is not PR since preimages can be efficiently found for at least half of all $y \in \{0,1\}^{n+1}$, namely the hash values that begin with 1. $\square$

Note: The hash function $\overline{H}$ is rather contrived. For *somewhat uniform* hash functions, i.e., hash function for which all hash values have roughly the same number of preimages, CR does indeed guarantee PR.

© *Alfred Menezes*

Claim 2*: Suppose H is somewhat uniform. If H is CR, then H is PR.



Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is not PR.

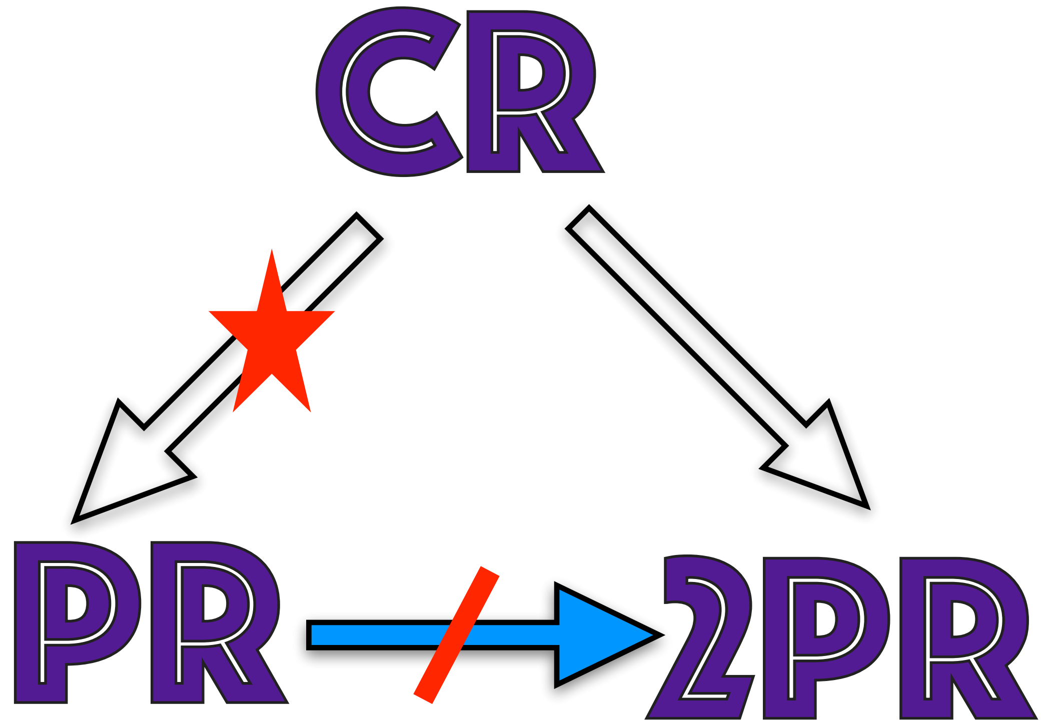
We'll show that H is not CR.

Select $x \in_R \{0,1\}^*$ and compute $y = H(x)$. Since H is not PR, we can efficiently find $x' \in \{0,1\}^*$ with $H(x') = y$. Since H is somewhat uniform, we expect that y has many preimages, and thus $x' \neq x$ with very high probability. Thus, (x, x') is a collision for H that we have efficiently found, so H is not CR. $\square$

Note: *For the remainder of the course we'll assume that hash functions are somewhat uniform.*

 © *Alfred Menezes*

Claim 3: PR does not guarantee 2PR



Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is PR.

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

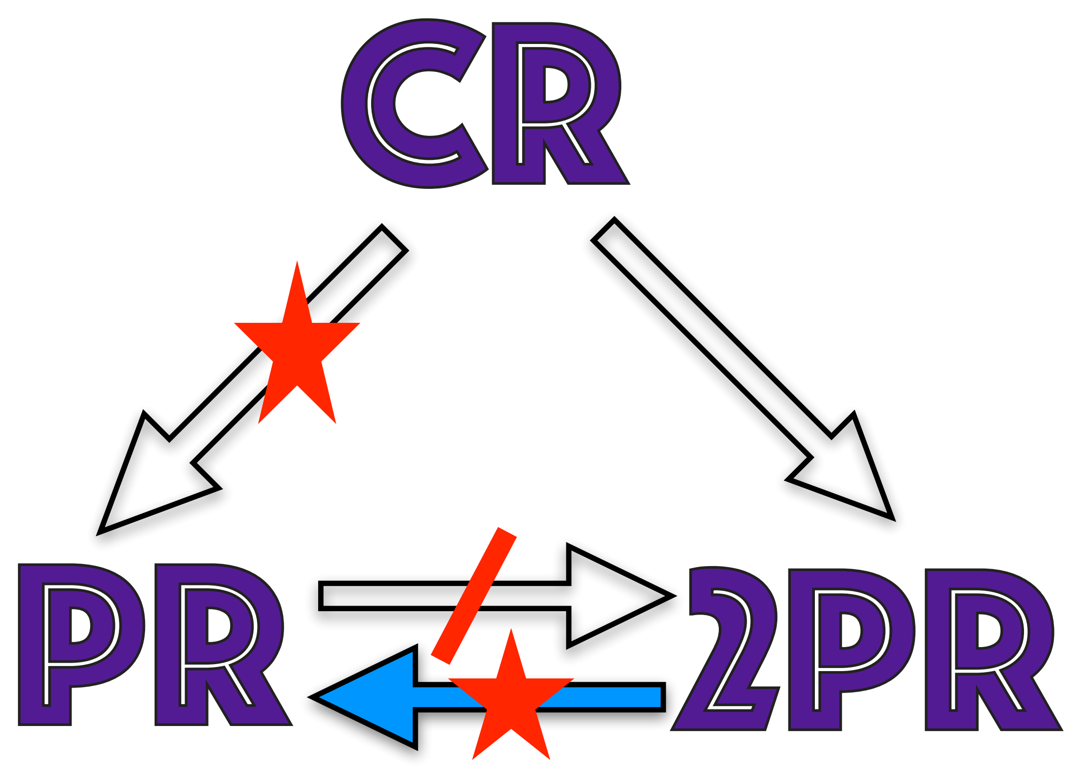
$\overline{H}(x_1, x_2, \ldots, x_t) = H(0, x_2, \ldots, x_t)$ for all $(x_1, x_2, \ldots, x_t) \in \{0,1\}^*$.

Then $\overline{H}$ is PR [Why?].

However, $\overline{H}$ is not 2PR [Why?]. $\square$

© *Alfred Menezes*

Claim 4: Suppose H is somewhat uniform. If H is 2PR, then H is PR.



Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is not PR.

We'll show that H is not 2PR.

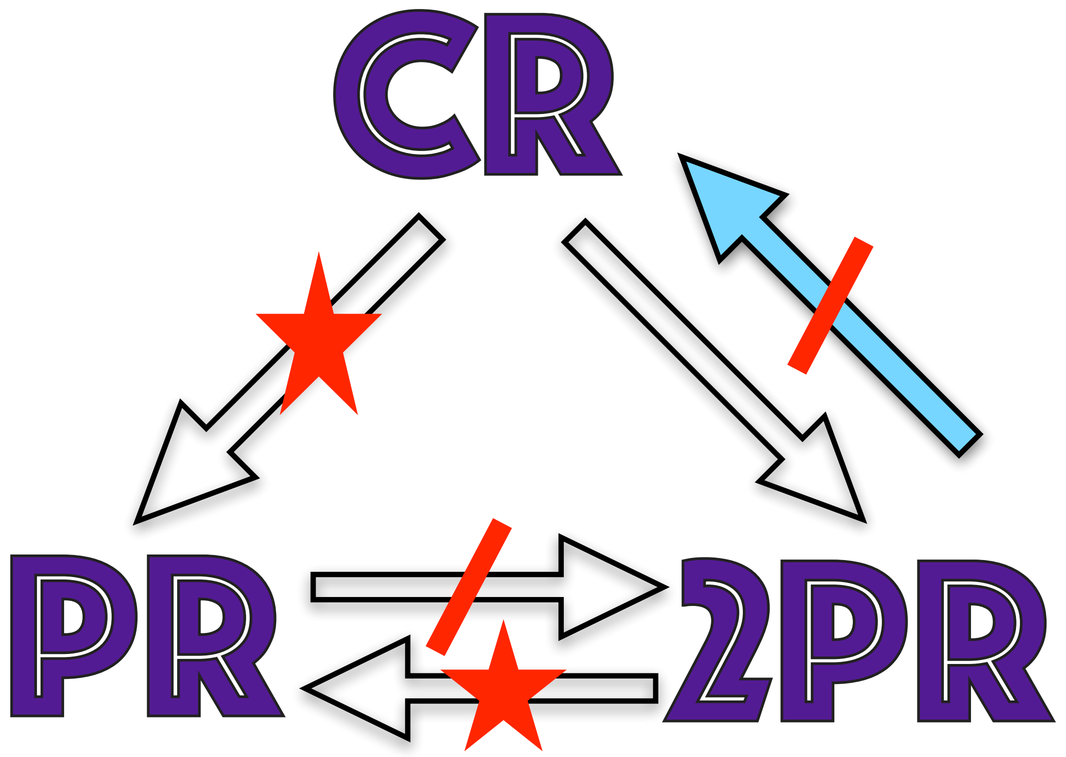
So, suppose we are given $x \in_R \{0,1\}^*$. We compute $y = H(x)$.

Since H is not PR, we can efficiently find $x' \in \{0,1\}^*$ with $H(x') = y$.

Since H is somewhat uniform, we expect that $x' \neq x$ with very high probability. Hence, x' is a second preimage of x that we have efficiently found.

Thus H is not 2PR. $\square$

Claim 5: 2PR does not guarantee CR



Proof: Suppose that $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ is 2PR.

Consider $\overline{H} : \{0,1\}^* \longrightarrow \{0,1\}^n$ defined by $\overline{H}(x) = H(x)$ if $x \neq 1$, and $\overline{H}(1) = H(0)$.

- Then $\overline{H}$ is not CR, since (0,1) is a collision for $\overline{H}$.

- Suppose now that $\overline{H} : \{0,1\}^* \longrightarrow \{0,1\}^n$ is not 2PR. We'll show that H is not 2PR.

So, we are given $x \in_R \{0,1\}^*$. Since $\overline{H}$ is not 2PR, we can efficiently find $x' \in \{0,1\}^*$, $x' \neq x$, with $\overline{H}(x') = \overline{H}(x)$. With probability essentially 1, we can assume that $x \neq 0,1$. Hence, $\overline{H}(x) = H(x)$.

Now, if $x' \neq 1$, then $H(x') = \overline{H}(x') = \overline{H}(x) = H(x)$.

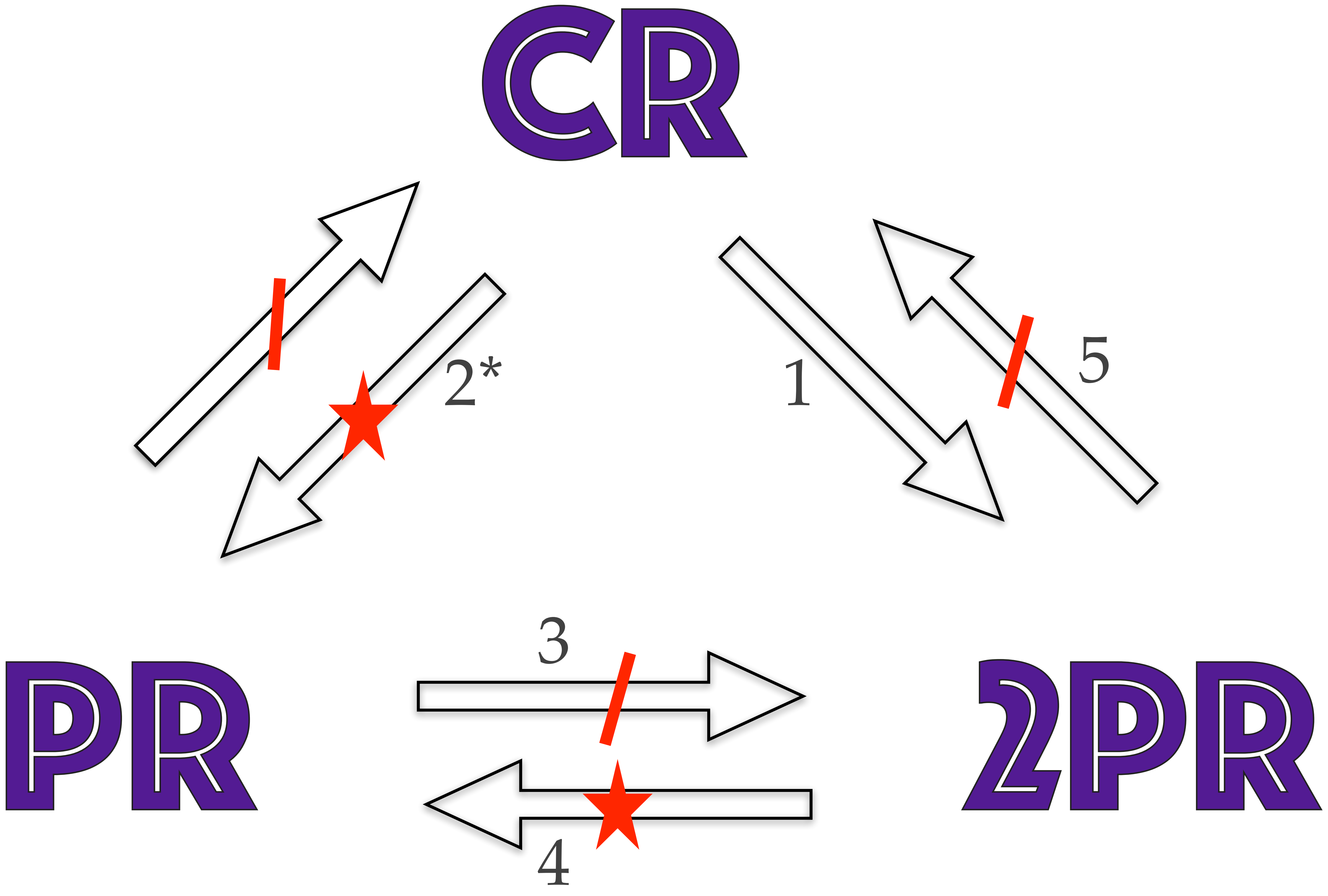
And, if $x' = 1$, then $\overline{H}(x') = \overline{H}(1) = H(0) = H(x)$.

In either case, we have efficiently found a second preimage for x w.r.t. H.

Hence, H is not 2PR, a contradiction. Thus, $\overline{H}$ is 2PR. $\square$

Relationships between PR, 2PR, CR

Let $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ be a hash function.



★ for somewhat uniform hash functions