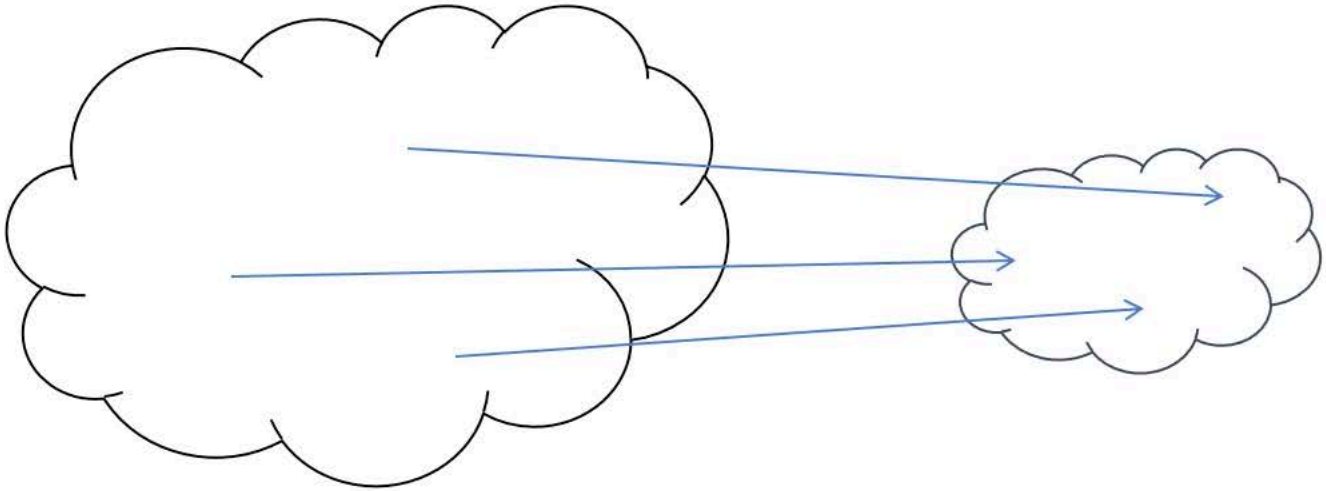
Introduction to Cryptography
Outline

- Hash functions
- Cryptographic hash functions
  - Property 1: Collision-resistance
  - Property 2: Hiding
  - Property 3: Puzzle-friendliness

Hash functions
Example

- Maps a large input space to a small output space

$$H(x) = x \bmod 8$$



Input: set of all integers

Output: set of integers in [0, 7]

Hash functions
Properties

1. Input can be any string of any size

2. Output is fixed-size

3. Efficiently-computable

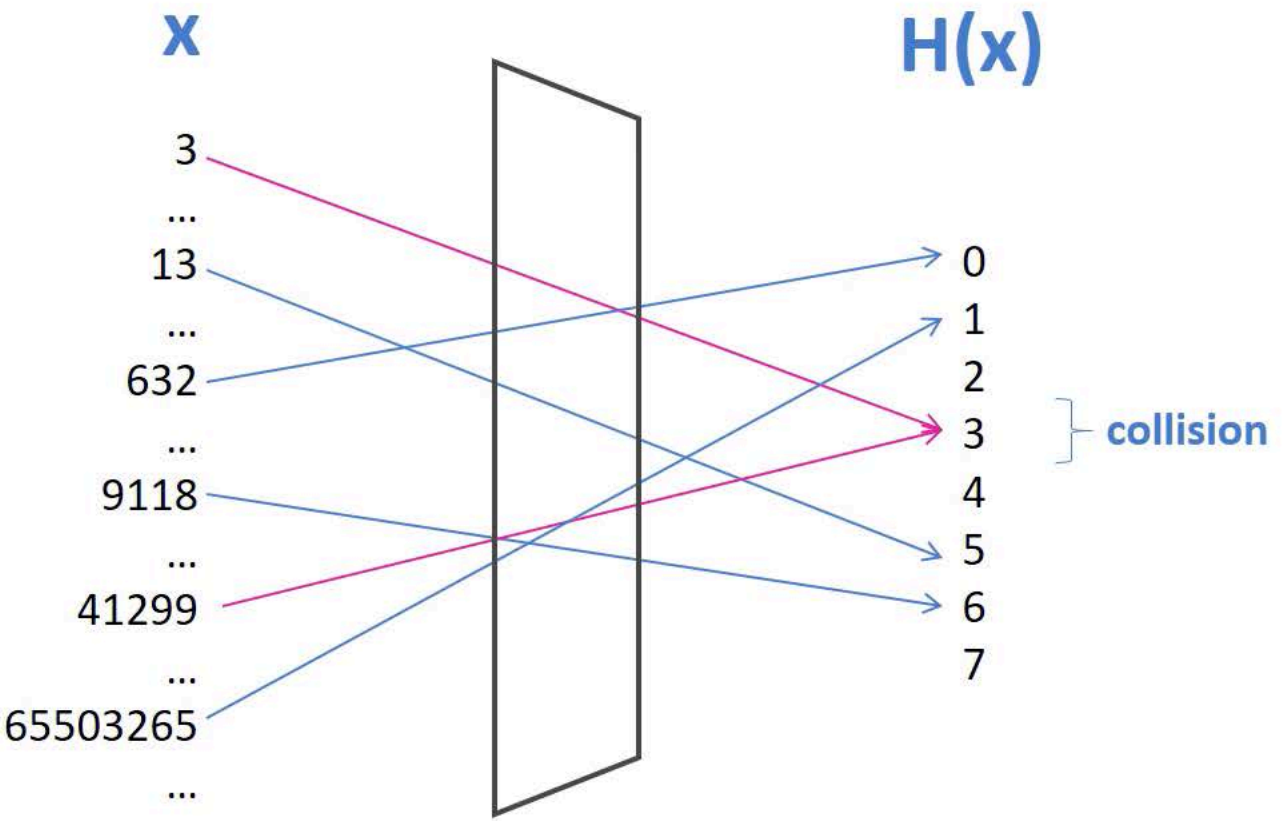
Cryptographic hash function
Definition

- A hash function is cryptographic if it has these three additional properties:

1. Collision-resistance
2. Hiding
3. Puzzle-friendliness

Collision-Resistance
Example

$H(x) = x \bmod 8$



Observation: There must be at least two x-values that map to the same H(x) value

Pigeon Hole Principle
Example

- Prove: There must be at least two people in New York City with the same number of hairs on their heads.

  Proof:

  1. Typical human head has an average of around 150,000 hairs.
  2. Reasonable to assume that no one has more than 1,000,000 hairs.
  3. There are more than 8,500,000 people in New York City
  4. Using the pigeon-hole principle, there must be at least two people with same number of hair on their heads.
     a. Pigeonholes: Each number of hairs on a person's head [1-1,000,000]
     b. Pigeons: People [1-8,500,000]
     c. Collision: There must be two people "mapped" to a specific number of hairs.

Collision-Resistance

Definition

infeasible ≠ impossible

- A hash function H is said to be collision-resistant if it is **infeasible to find** two input values that have the same output value.
- Infeasible to find
  - There are no **known, practical** method to find collisions.
  - There may be known, theoretical methods to find collisions.
- Example: **H(x) = x mod 8**. Collision-resistant?
  - No, because can easily find a collision
  - Collision: x = 3 and y = 11
- Example: **H(x) = x mod 2^{256}**. Collision-resistant?
  - No, because can easily find a collision:
  - Collision: x = 3 and y = 2^{256}+3

Collision-Resistance
Example

- Example: **H(x) = MD5(x)**
  - Collision-resistant?
    - Believed to be until March 2005.
    - Researchers from Sandong University in China published an article on an algorithm that finds collisions in MD5.
  - Collision:

```
x = d131dd02c5e6eec4693d9a0698aff95c 2fcab58712467eab4004583eb8fb7f89
    55ad340609f4b30283e488832571415a 085125e8f7cdc99fd91dbdf280373c5b
    d8823e3156348f5bae6dacd436c919c6 dd53e2b487da03fd02396306d248cda0
    e99f33420f577ee8ce54b67080a80d1e c69821bcb6a8839396f9652b6ff72a70

y = d131dd02c5e6eec4693d9a0698aff95c 2fcab50712467eab4004583eb8fb7f89
    55ad340609f4b30283e4888325f1415a 085125e8f7cdc99fd91dbd7280373c5b
    d8823e3156348f5bae6dacd436c919c6 dd53e23487da03fd02396306d248cda0
    e99f33420f577ee8ce54b67080280d1e c69821bcb6a8839396f965ab6ff72a70
```

H(x) = H(y) = 79054025255fb1a26e4bc422aef54eb4

1. Wang, X., Hongbo, Y. *How to break MD5 and other hash functions.* EUROCRYPT'05.
2. 2. http://www.mathstat.dal.ca/~selinger/md5collision/

Collision-Resistance
Example

- Example: **H(x) = MD5(x)**
  - Collision-resistant?
    - Believed to be until March 2005.
    - Researchers from Sandong University in China published an article on an algorithm that finds collisions in MD5.
  - Collision:

```
x = d131dd02c5e6eec4693d9a0698aff95c 2fcab58712467eab4004583eb8fb7f89
    55ad340609f4b30283e488832571415a 085125e8f7cdc99fd91dbdf280373c5b
    d8823e3156348f5bae6dacd436c919c6 dd53e2b487da03fd02396306d248cda0
    e99f33420f577ee8ce54b67080a80d1e c69821bcb6a8839396f9652b6ff72a70

y = d131dd02c5e6eec4693d9a0698aff95c 2fcab50712467eab4004583eb8fb7f89
    55ad340609f4b30283e4888325f1415a 085125e8f7cdc99fd91dbd7280373c5b
    d8823e3156348f5bae6dacd436c919c6 dd53e23487da03fd02396306d248cda0
    e99f33420f577ee8ce54b6708080d1e c69821bcb6a8839396f965ab6ff72a70
```

H(x) = H(y) = 79054025255fb1a26e4bc422aef54eb4

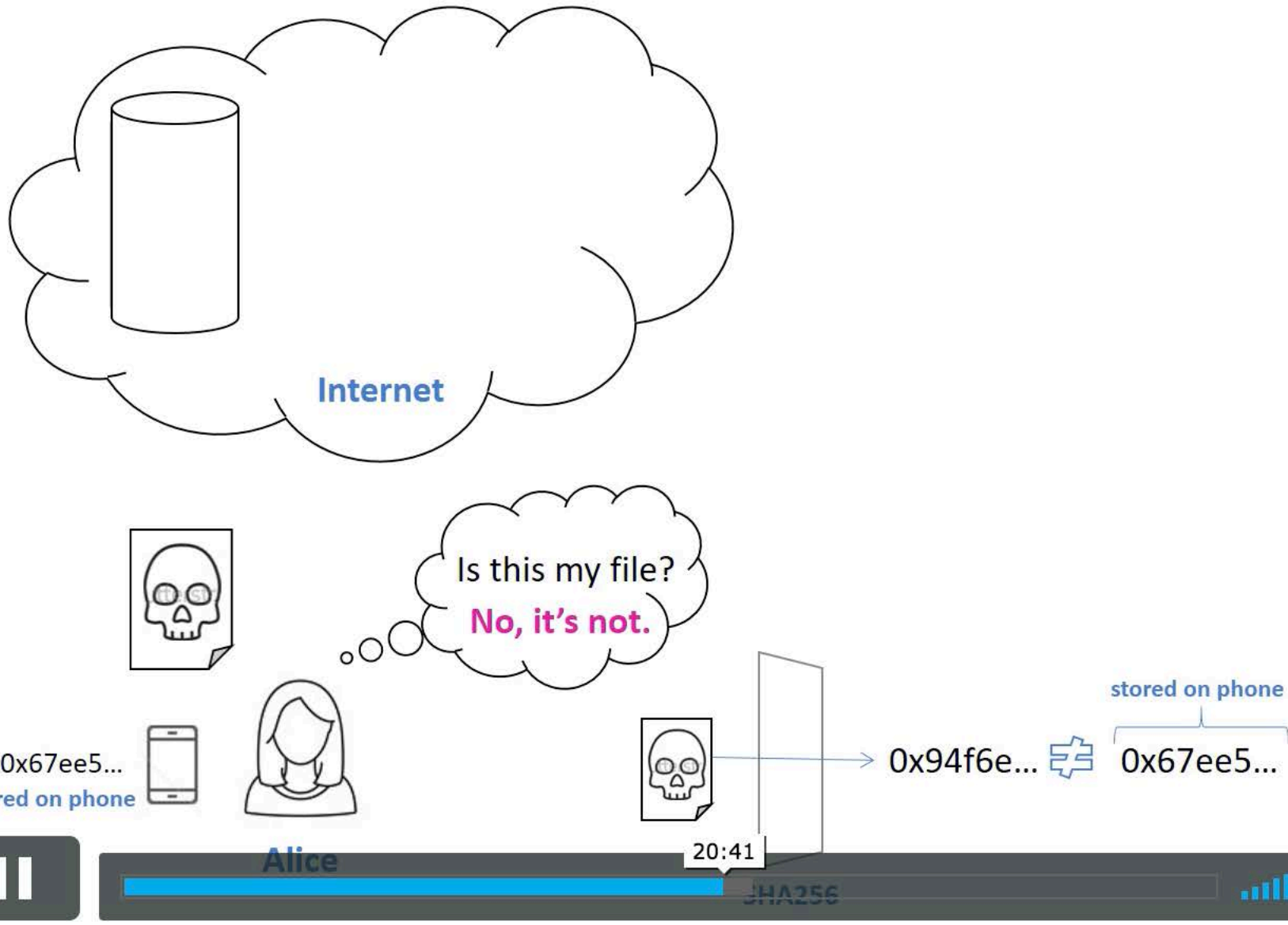
1. Wang, X., Hongbo, Y. *How to break MD5 and [illegible]ash functions.* EUROCRYPT'05.
2. http://[illegible]lision/

Collision-Resistance

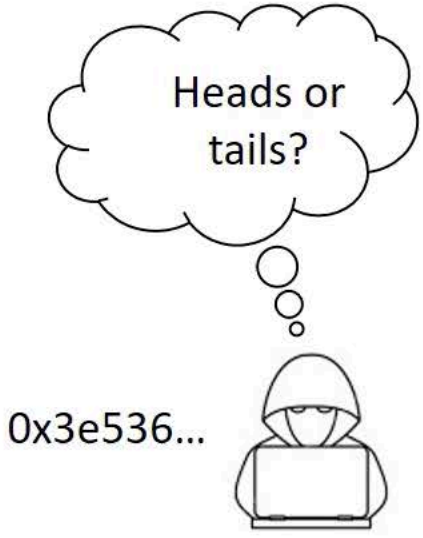
- All hash functions have collisions.
- No hash function has ever been proven to be collision-resistant.
- **Theoretical** method for finding collisions in any hash function:
  - Until a collision is found:
    - Randomly select two input values x, y
    - If H(x)=H(y), then found collision
  - SHA256: would take more than 10^{27} years to find a collision on average
- Cryptographic hash functions simply make it very difficult for collisions to be found.
- Value of collision-resistance:
  - For a collision-resistant hash function H:
  - If we know that two inputs x and y are different, then we are confident that their hashes, H(x) and H(y), are different

Collision-Resistance

Application: Message Digests



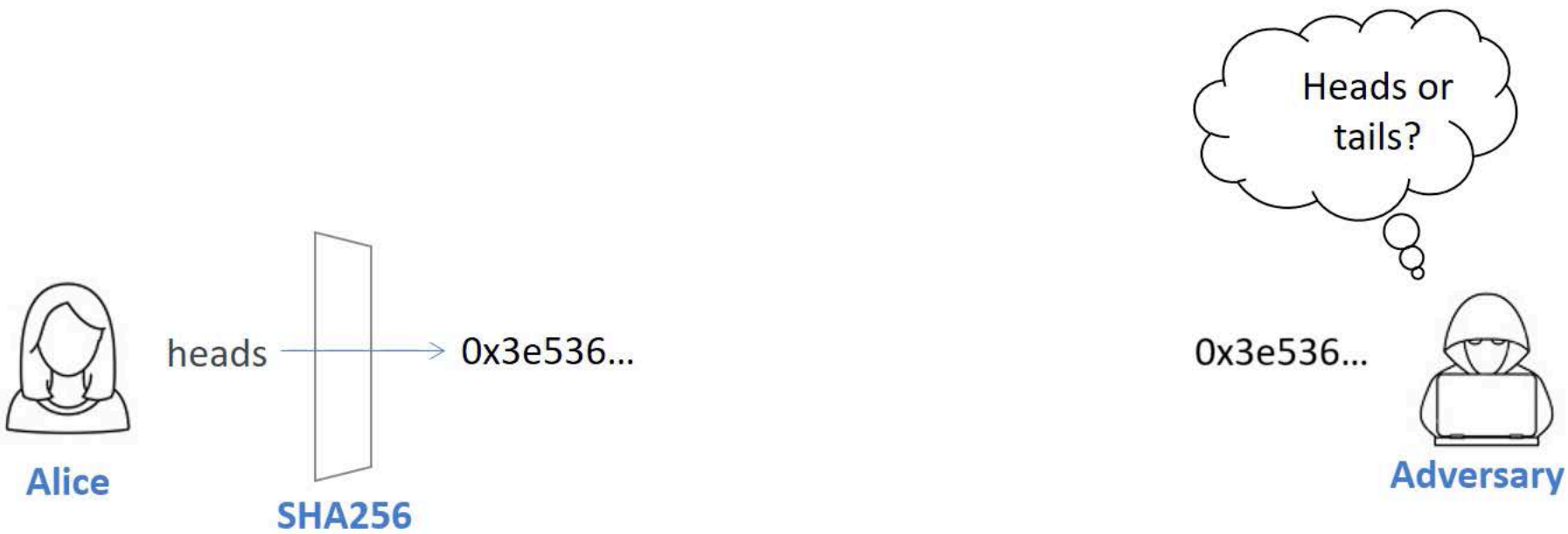
Hiding



Adversary

Hiding
Definition

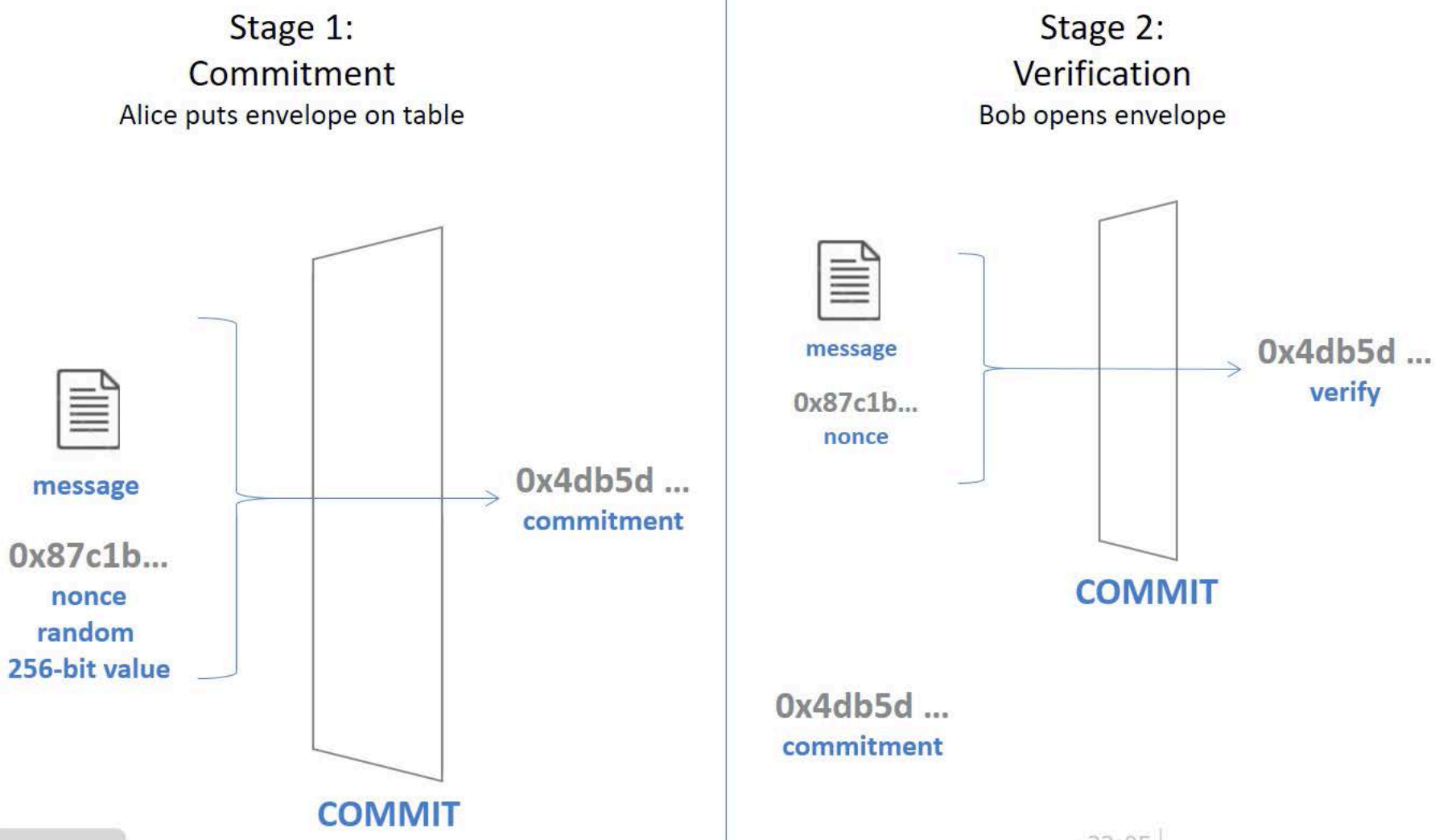
- A hash function H has the *hiding* property if when given the output of the hash function H(x), it is infeasible to find x.
- Non-trivial property to achieve:
  - Suppose Alice flips a coin, computes the hash of the outcome H(x), and publicly advertises H(x)
  - Can an adversary determine the outcome from H(x)?



Hiding
Revised Definition

- A hash function H has the ***hiding*** property if when a secret value r is chosen from a probability distribution that has high min-entropy, then, given $H(x \circ r)$, it is infeasible to find x

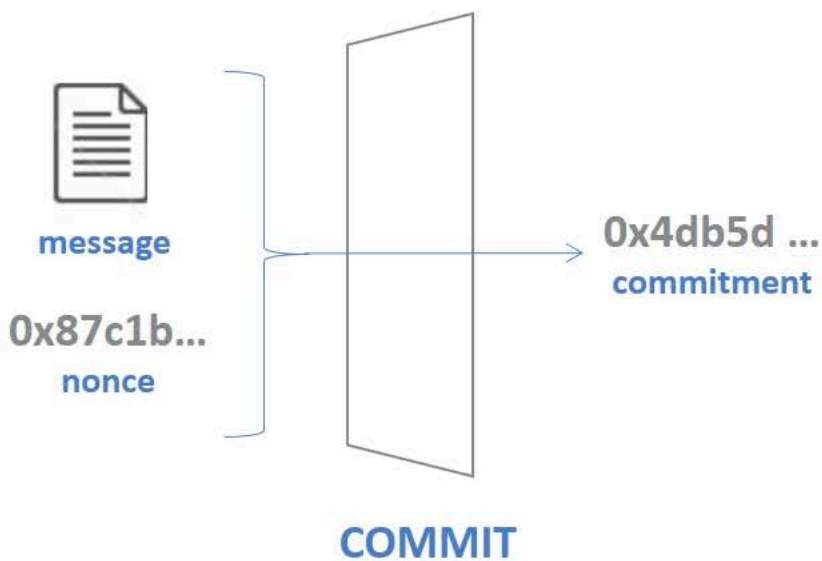
Hiding
Digital Commitment Scheme



Hiding
Commitment Scheme

Two security properties required for **COMMIT** function:

1. Hiding: Given commitment, it is infeasible to find message.
    - Message remains secret in stage 1
2. Binding: It is infeasible to find two pairs (message, nonce) and (message', nonce') that collide.
    - Alice can't commit to a message in stage 1, and then present Bob with a different message in stage 2.

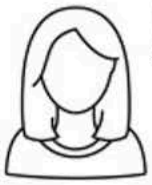


Puzzle-friendliness
Definition

Solve this puzzle:

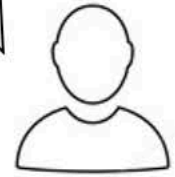
1. Hash function **H**
2. Random value **id**
3. Target set **Y**

Find **x** such that:
$H(id \circ x) \in Y$



Alice

If **H** is puzzle-friendly, then there's no solving strategy for this puzzle that is much better than just trying random values of x.



Bob