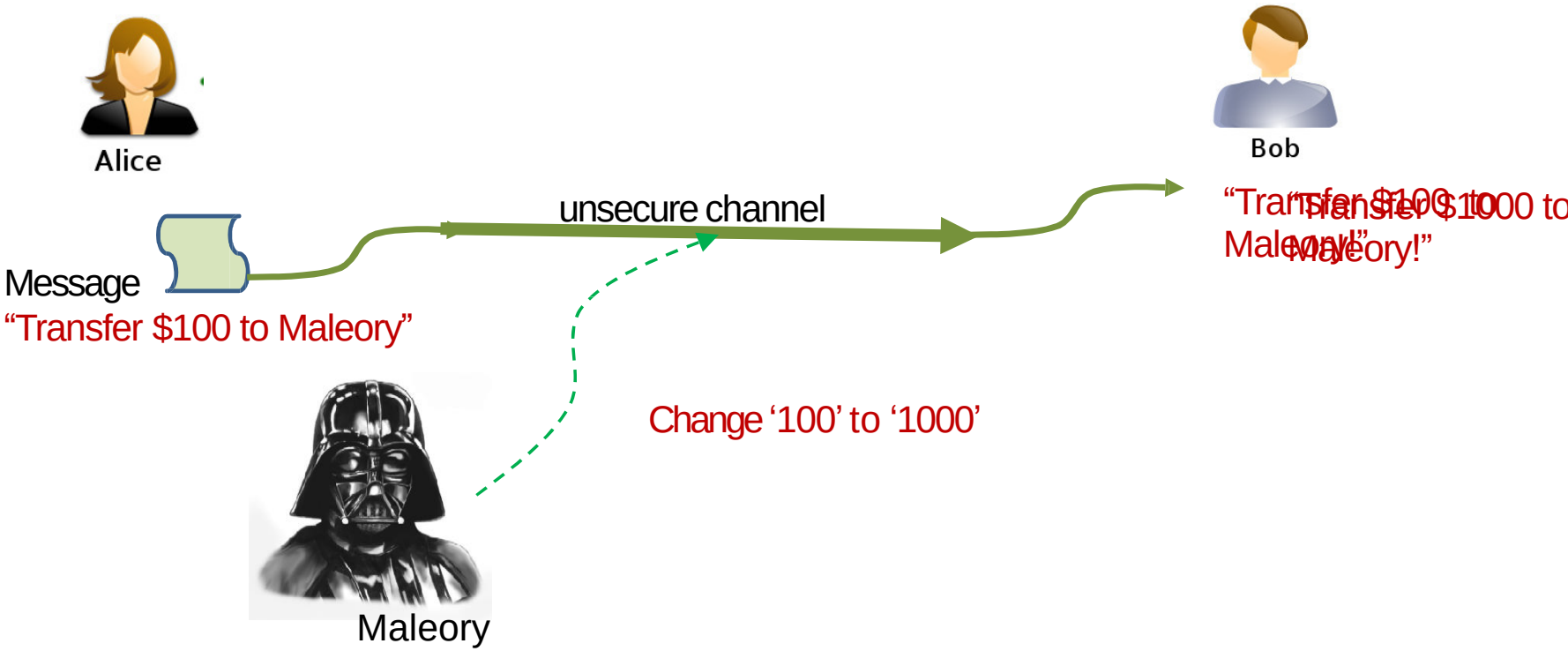
CS557: Cryptography

Cryptographic Hash Function II

S. Tripathy
IIT Patna

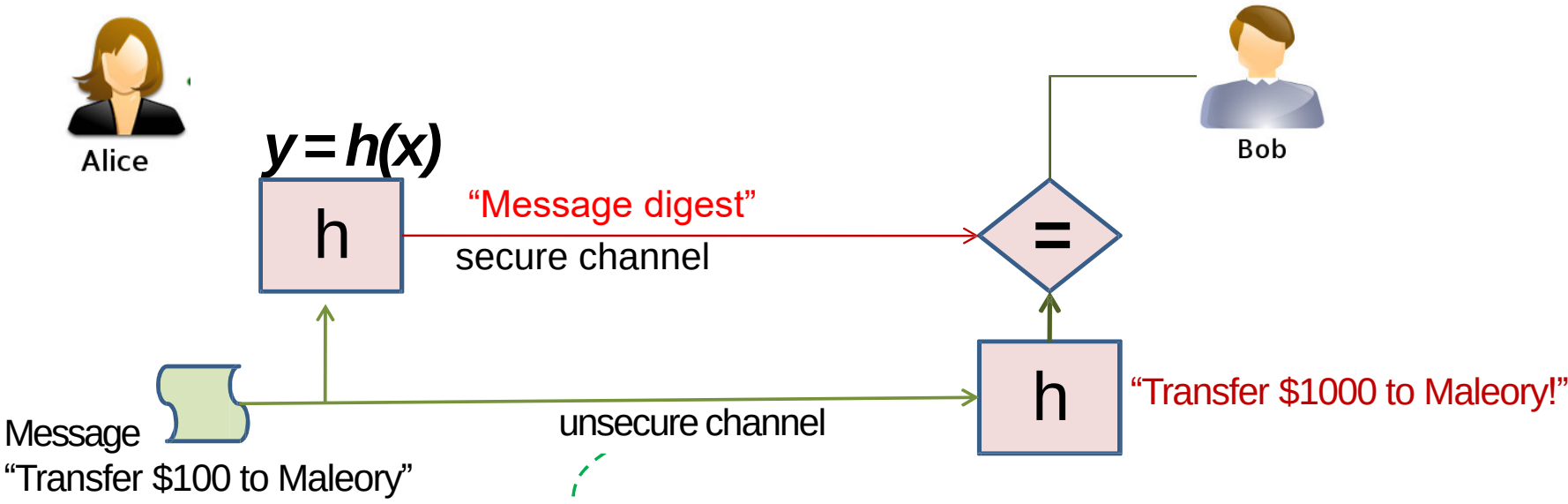
Issues with Integrity

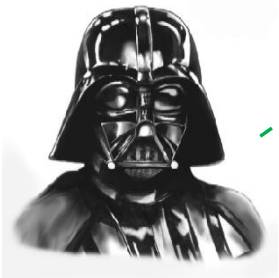
Note…. We are not concerned with confidentiality Now



How can Bob ensure that Alice's message has not been modified ?

Hash (Manipulation Detection code





To suceed

Maleory needs to modify x to x', such that

$h(x) = h(x')$

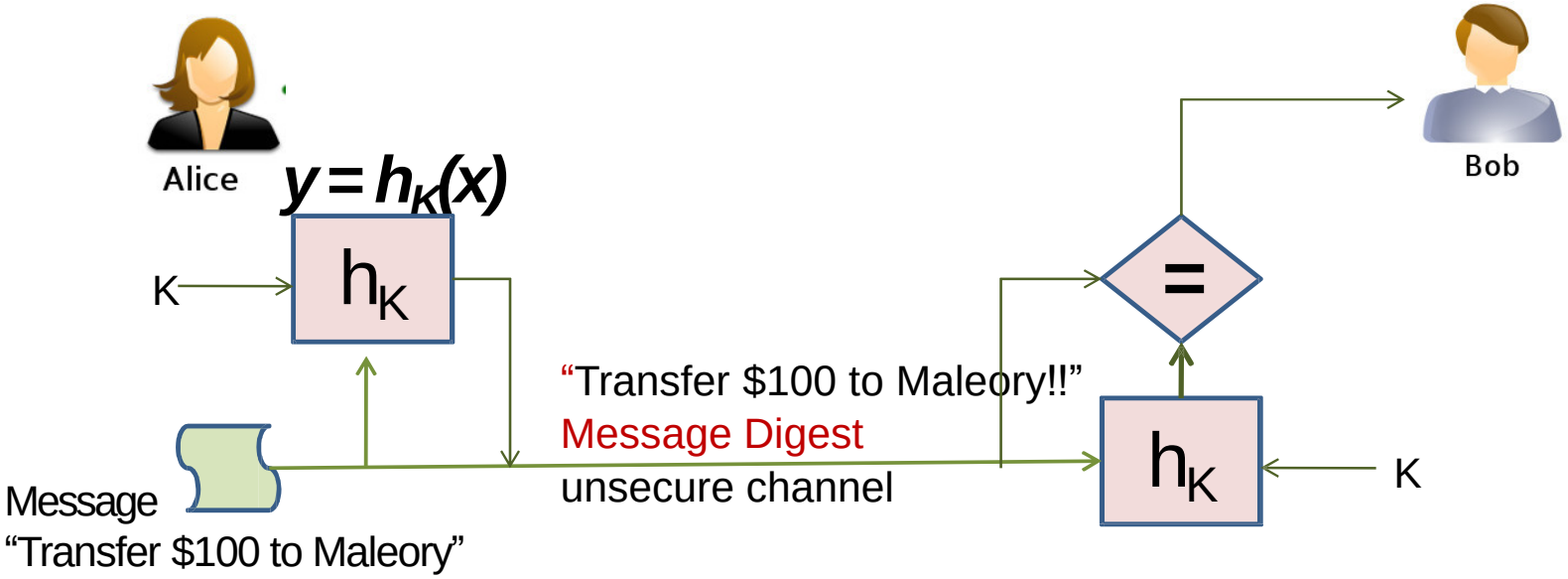
Goal of Hash functions is to resist such collisions

Attacks against MDC

OWHF: given y find x s.t. h(x)=y; or given (x,h(x)) find $x' \neq x$ s.t. h(x')=h(x)

CRHF: find any two inputs $x' \neq x$ s.t. h(x')=h(x) (birthday attack)

Message Authentication Codes (MAC)



MACs can allow the message and the digest to be sent over an insecure channel

However, it requires Alice and Bob to share a common key

Attacks against MAC

without <u>knowing k</u> compute $(x, h_k(x))$ given $(x_i, h_k(x_i))$ with $x_i \neq x$

Applications of Hash functions in Security

- Digital signatures
- Random number generation
- Key updates and derivations
- One way functions
- MAC
- Detect malware in code
- User authentication (storing passwords)

Hash Function

- Hash Function produces a fingerprint of some file/message/data

  $h = H(M)$

Requirements for Hash Functions

Compression:

Efficiency:

Oneway:

Weak collision resistance

Strong collision resistance:

Simple but Insecure Hash Function

- bit-by-bit XOR of every block
  - $C_i = b_{i1}$ *xor* b_{i2} *xor . . . xor* b_{im}
  - reasonably effective as data integrity check
  - in most normal text files, the high-order bit of each octet is always zero
- H(x,y) = ax + by mod m
  - Not Secure

Bruteforce Attacks on Hash Functions

- Attacking one-wayness
  - Goal: given $h:X \rightarrow Y$, $y \in Y$, find x such that $h(x)=y$
  - Algorithm:
    - pick a random value x in X,
    - if h(x)=y, return x; otherwise iterate
    - after failing q iterations, return fail
  - The average-case success probability is

$$\varepsilon = 1 - \left(1 - 1/|Y|\right)^q \approx \frac{q}{|Y|}$$

  - Let $|Y|=2^m$, to get ε to be close to 0.5, $q \approx 2^{m-1}$

Bruteforce Attacks on Hash Functions

- Attacking collision resistance
  - Goal: given h, find x, x′ such that h(x)=h(x′)
  - Algorithm: pick a random set X_0 of q values in X
    for each $x \in X_0$, computes $y_x=h(x)$
    if $y_x=y_{x'}$ for some $x' \neq x$ then return (x,x′)
    else fail
  - The average success probability is
  - Analogous to that of Birthday paradox

Birthday Paradox

- The probability that in a group of people two will share the same birthday

Event A :atleast two people in the group have the same birthday

Event A': no two people in the group have the same birthday

$$\Pr[A] = 1 - \Pr[A']$$

$$\Pr[A] = 1 - \prod_{i=1}^{Q-1}\left(1 - \frac{i}{365}\right)$$

- If 23 people are there in a room, the probability that two people would share the same birthdays is 1/2

Hash Functions

Effort Required for length = n-bit

One Way / Pre-image	2^n
Weak Collision/ 2^{nd} Pre-image Resistance	2^n
(Strong) Collision Resistance	$2^{n/2}$

- Finding collisions is easier than solving pre- image or second preimage

- Thanks