University of Colorado **Boulder**

CSCI 3403 INTRO TO CYBERSECURITY

Lecture: 3-2

Topic: Hashing

Presenter: Matt Niemiec

Hashing

What's a Hash?

- One-way encryption
- No key
- Uniform distribution
- Small changes in input result in large changes in output

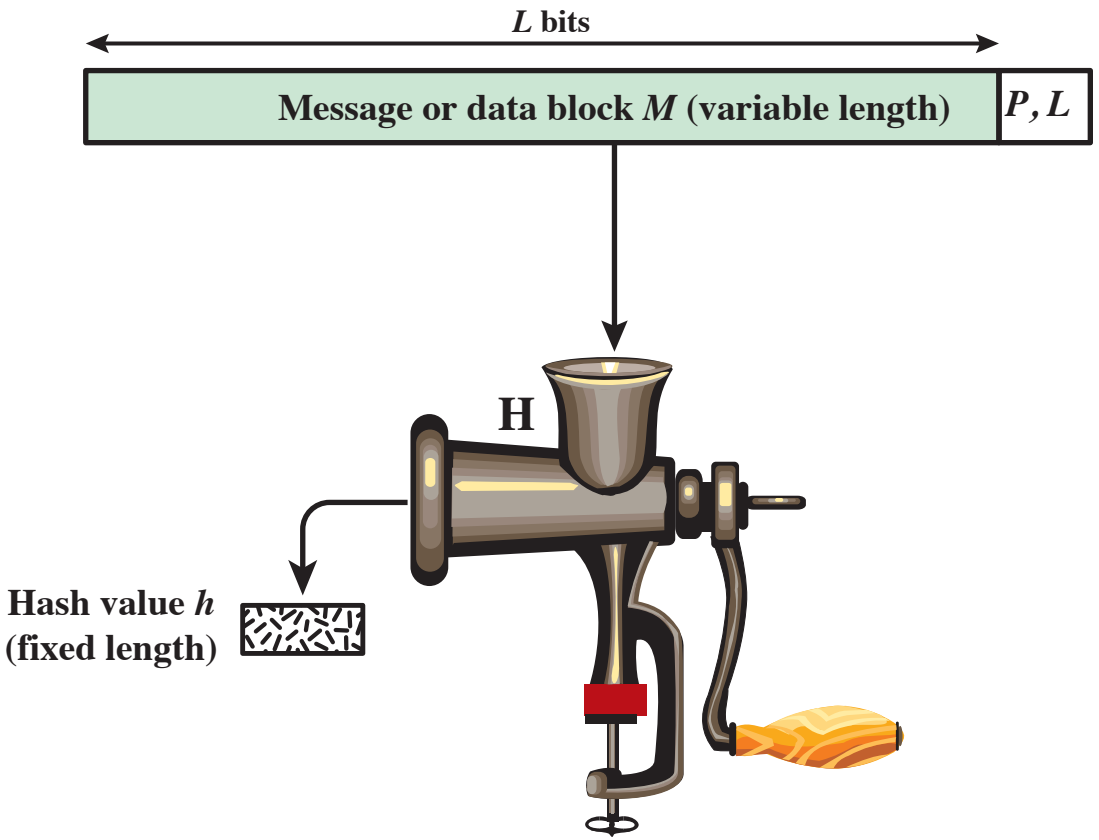


Figure 2.4 Cryptographic Hash Function; $h = H(M)$

Why a Hash?

- Store passwords
- Verify checksums
- Useful data structure
- Application signatures
- Guarantee message integrity
- To name a few!

Security Properties

A Useful Cryptographic Hashing Function Has the Following Properties

- Hashing function H can be applied to a block of data of any size
- H produces a fixed-length output
- H(*x*) is relatively easy to compute for any given *x*, making both hardware and software implementations practical
- For any given code h, it is computationally infeasible to find x such that H(x) = h

A Useful Cryptographic Hashing Function Has the Following Properties

- For any given block x, it is computationally infeasible to find $y \neq x$ with $H(y) = H(x)$
- It is computationally infeasible to find any pair (x, y) where $x \neq y$ such that $H(x) = H(y)$

Happy Birthday!

The Birthday Paradox

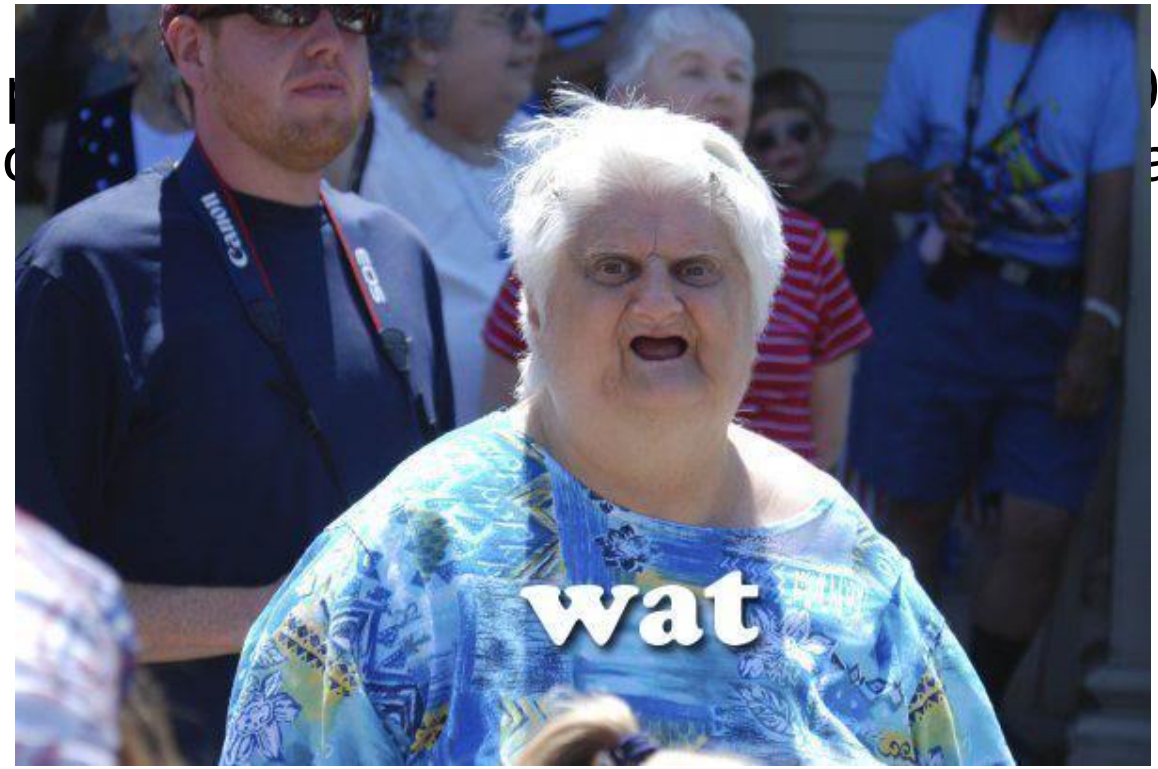
- If 23 people are in a room, then there is a [redacted] chance that at least two people have the same birthday

The Birthday Paradox

- If 23 people are in a room, then there is a >50% chance that at least two people have the same birthday!

The Birthday Paradox

- If 23 p
  chanc



0%
ay!

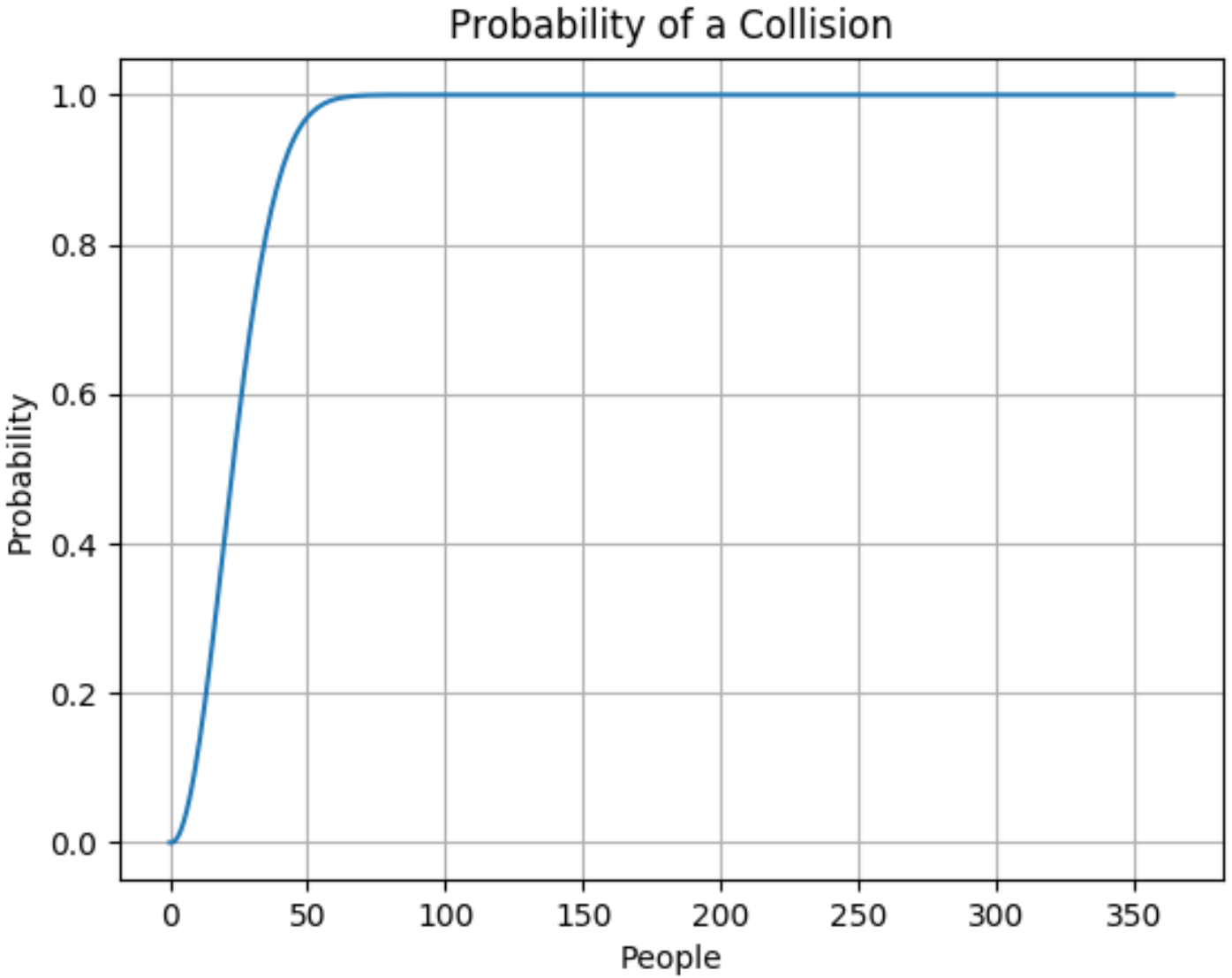
Some Math


ONE DIE
TWO
THREE
FOUR
FIVE
SIX
virtualdiceroll.com
VirtualDiceRoll on Facebook
THROW DICE!
Decision maker

But There's a Lot of Birthdays!

- Let's try it!
- Assume equal distribution*

* This is wrong, wrong, wrong!!!


Probability of a Collision
Probability
1.0
0.8
0.6
0.4
0.2
0.0
0
50
100
150
200
250
300
350
People

How Many Collisions Are Here?

- Some math later...
- The expected number of collisions in a room of *n* people is:

$$n(1-(1-1/365)^{n-1})$$

- Let's try it!

So What?

- You know a cool, geeky fun fact
  - Share it at a party
- More importantly... Hashing is hard!

Modern Algorithms

Not Too Much To Say

- Don’t use MD5
  - It’s really fast! Danger!
- SHA-1 slightly better and slightly slower than MD5
  - Not really recommended any more
- SHA-2 is even better
  - Provides SHA-256, SHA-384, SHA-512
- SHA-3 came from NIST competition
  - Based on SHA-2: Based on SHA-1: Similar to MD4/MD5?

So... Which One To Use?

- Check hardware compatibility
- Almost all are “pretty good”
- SHA-256 is typically a good option

Message Integrity

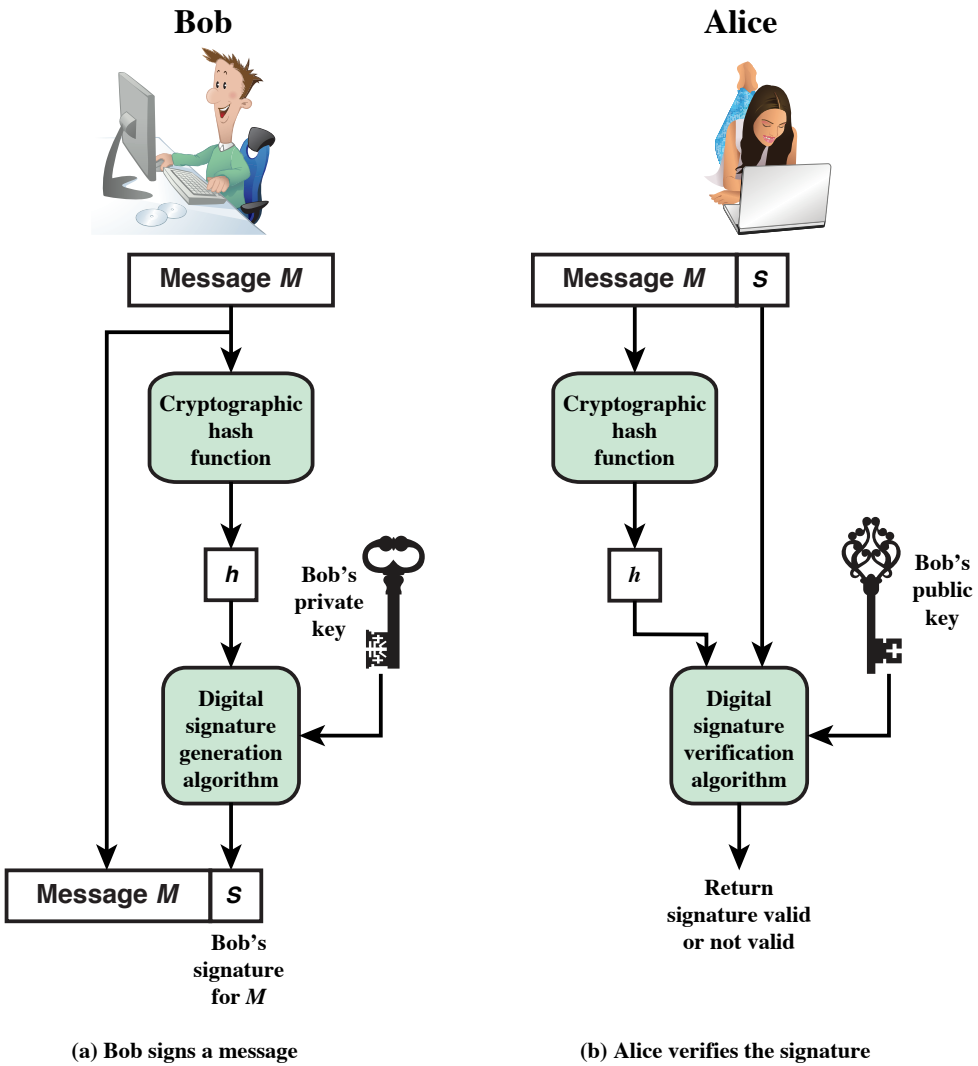


Figure 2.7 Simplified Depiction of Essential Elements of Digital Signature Process

Message Authentication Code

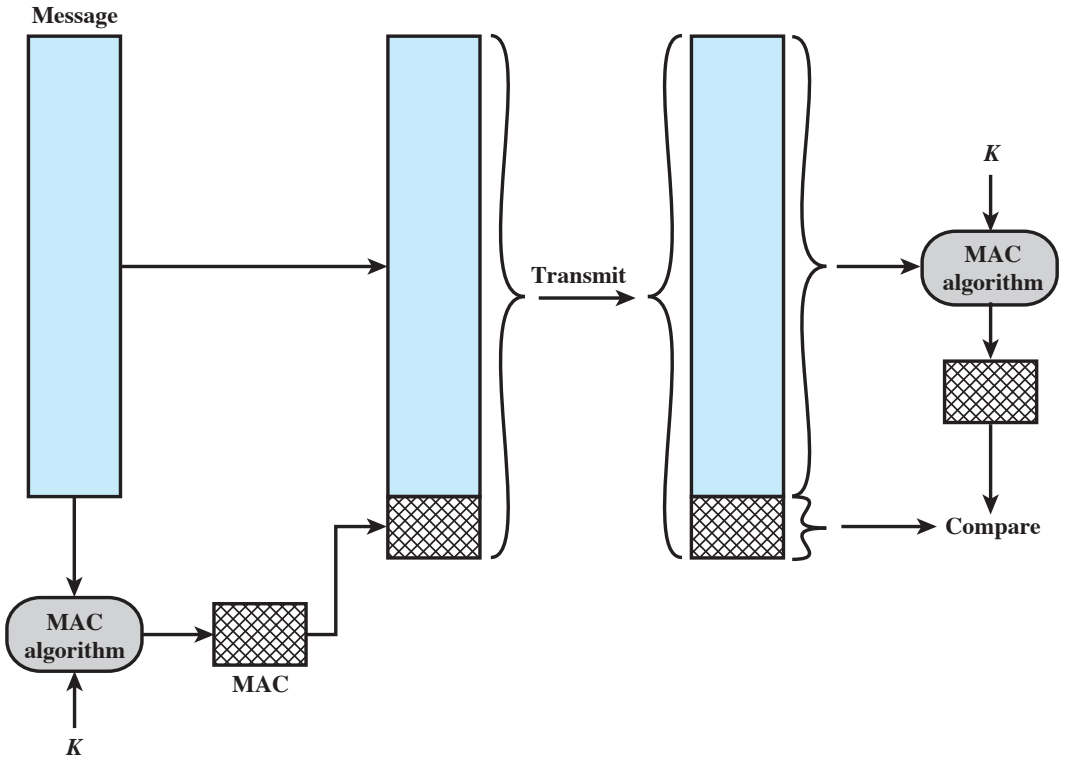


Figure 2.3 Message Authentication Using a Message Authentication Code (MAC).

All Three Methods

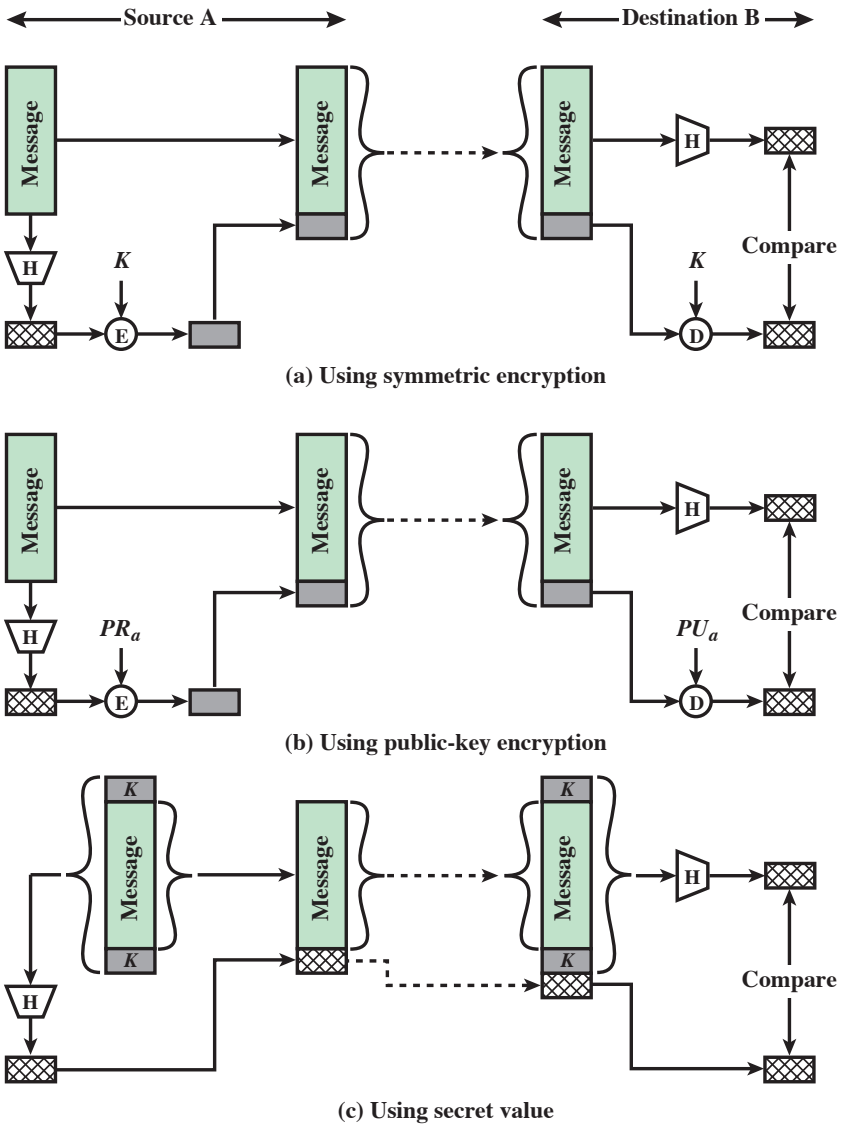


Figure 2.5 Message Authentication Using a One-Way Hash Function.