

A close-up photograph of a person's hands in a dark suit jacket. The person is holding a magnifying glass over an open notebook and several sheets of paper. The papers contain cryptograms, which are grids of letters and symbols. The person's right hand is holding a black pen, poised to write in the notebook. The scene is dimly lit, with a strong light source from the left, creating a dramatic effect. The background is dark and out of focus, showing a desk with a small white object and some papers.

Cryptography Algorithms

Topics

- What is **cryptography**?
- What are the **basics of cryptographic algorithms**?
 - What are **cryptographic hashes**?
 - What is a **secret key encryption**?
 - What is **public-key encryption**?

Cryptography:

The absolute basics

Context

- **Cryptography**
 - A very broad area.
 - We'll focus on how to use cryptography.
 - We just touch on the basics!
- **Reference Material**
 - Coursera course: Cryptography I
<https://www.coursera.org/learn/crypto>

The CIA Model

- CIA model: the classic security model.

- Confidentiality:

- ..

- Integrity:

- ..

- and only by authorized parties.

- Availability:

- ..

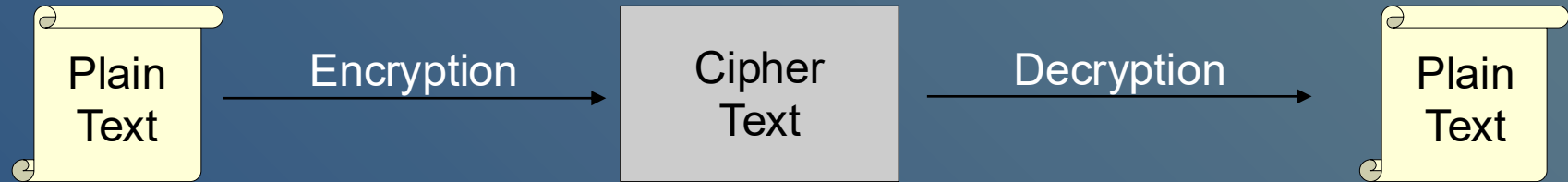
- Threat examples

- Against confidentiality: classified information leak

- Against integrity: fake images/videos

- Against availability: Denial-of-Service (DoS) attacks

General Cryptography Process



- Cryptographers invented secret codes to hide messages from unauthorized observers.
- **Challenges:**
 - How can you **hide** a message from everyone but the intended recipient?
 - How can the recipient know the message is **authentic**?

ABCD: Traditional Cryptography

- **Traditional Cryptography:**
 - Secret codes, which are secret algorithms.
 - **E.g., Caesar Cipher:** ..

For +1 'A' becomes 'B'.

- **ABCD:** Which of the following is the **cipher text** from using a **3-letter shift Caesar Cipher** on the plain text “**Hello world**”?

- (a) EBIL TLOIA
- (b) KHOOR ZRUOG
- (c) IFMMP XPSME
- (d) LOWOR LDHEL

- **What is the problem with a secret algorithm?**
 - When your algorithm (or code book) is compromised,
 - ..

Modern Encryption

- Algorithms are Public

- ..

- May be symmetric (secret key) or asymmetric (public key).

- Why is this better?

- If algorithm or code is secret,
then if it falls into the wrong hands it means code is useless.

- If only key is private,
then if it falls into the wrong hands then

- ..

Crypto Algorithm Goals

- Choose an encryption algorithm such that:
 - Given a key, it should be
 - ..
 - Without a key, it should be
 - ..
- Strength of security often based on length of key:
Longer key is more difficult to guess (by brute-force).

Window of Validity

- **Window of Validity**

..

- Must only use algorithm that have not been compromised.

- **Problem:**

Window of validity of your crypto function

..

- Design systems so you can replace the crypto function easily.

- **Example Windows of Validity**

- **1993**: **SHA-0** was published.

- **1995**: Possible weakness was found in the **SHA-0** algorithm; replaced with **SHA-1**.

- **2004**: Published way to compromise **SHA-0**

- **2017**: Published way to compromise **SHA-1**

- **????**: Published way to compromise **SHA-256?**

Three Types

- Types of cryptography algorithms based on their keys:
 - Zero keys: ..
 - One key: ..
 - Two keys: ..

Cryptographic Hash Functions (Zero Keys)

Cryptographic Hash Functions

- Suppose we have a cryptographic hash function $h()$
 - It takes a message m of arbitrary length as input and
 - ..

- Toy example:
 $h(m) = (m^2) \% 4321$

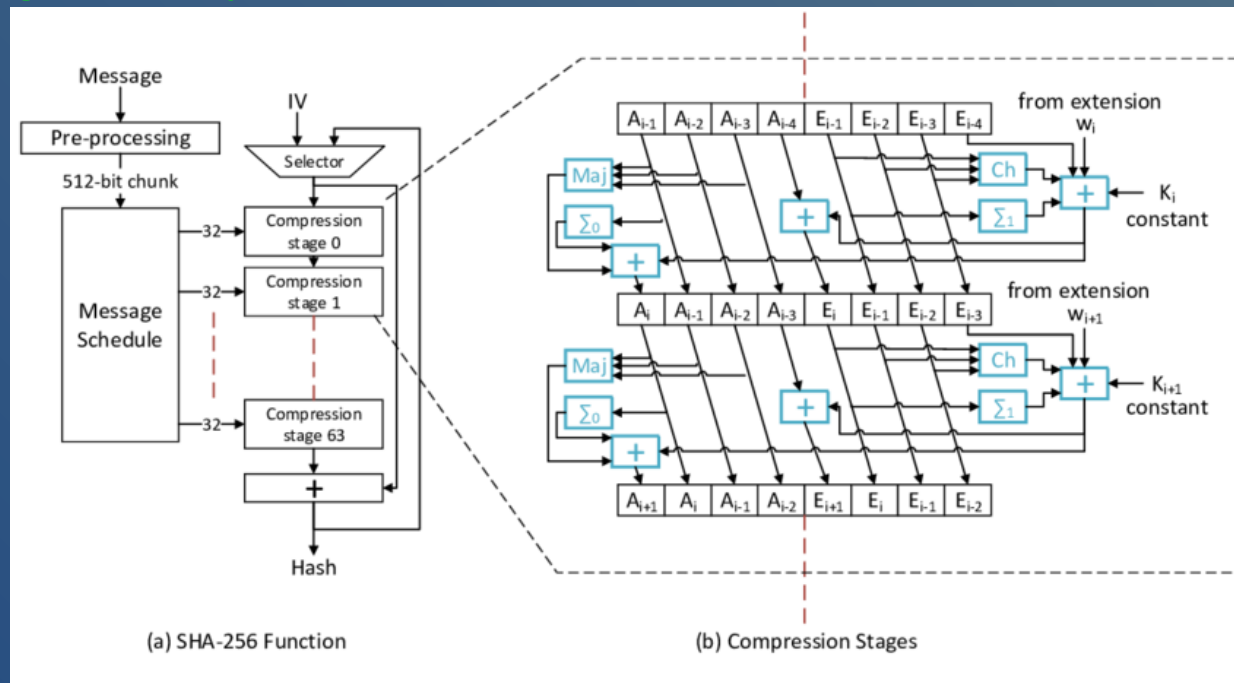
m	m in hex	h(m)
AAAA	(0x41414141) -->	2242
BBBB	(0x42424242) -->	893
CCCC	(0x43434343) -->	2558
DDDD	(0x44444444) -->	2916
EEEE	(0x45454545) -->	1967
FFFF	(0x46464646) -->	4032
GGGG	(0x47474747) -->	469
HHHH	(0x48484848) -->	4241
IIII	(0x49494949) -->	2385
JJJJ	(0x4A4A4A4A) -->	3543
KKKK	(0x4B4B4B4B) -->	3394
LLLL	(0x4C4C4C4C) -->	1938

Hash Function Properties

- ..
 - It should be **easy to compute $h(m)$**
- ..
 - Given $h(x)$, it should be difficult to find x .
 - i.e., the reverse of $h()$ should be difficult to compute.**
- ..
 - Given x , it should be difficult to find x' where $h(x') == h(x)$
 - i.e., Given a value and a hash function, it should be difficult to find another value that produces the same hash.**
- ..
 - It should be difficult to find two messages x and x' where $h(x) == h(x')$
 - i.e., given a hash function, it should be difficult to find two values that produce the same hash.**

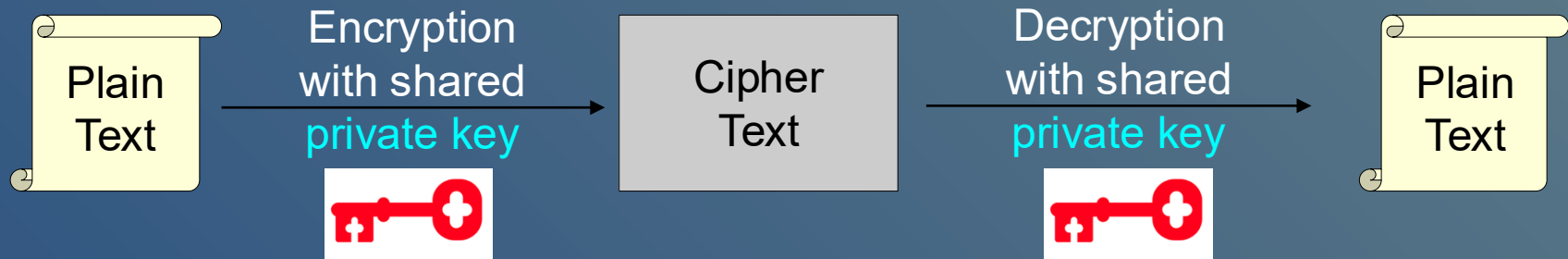
Ideal Hash

- Ideally, we want all these properties for a strong cryptographic hash function.
 - However, not all hash functions provide all these properties.
- Example good crypto hash function: SHA-256.



Private Key Cryptography
or
Symmetric Key Cryptography
(One key)

Private (Symmetric) Key Crypto



- One key:

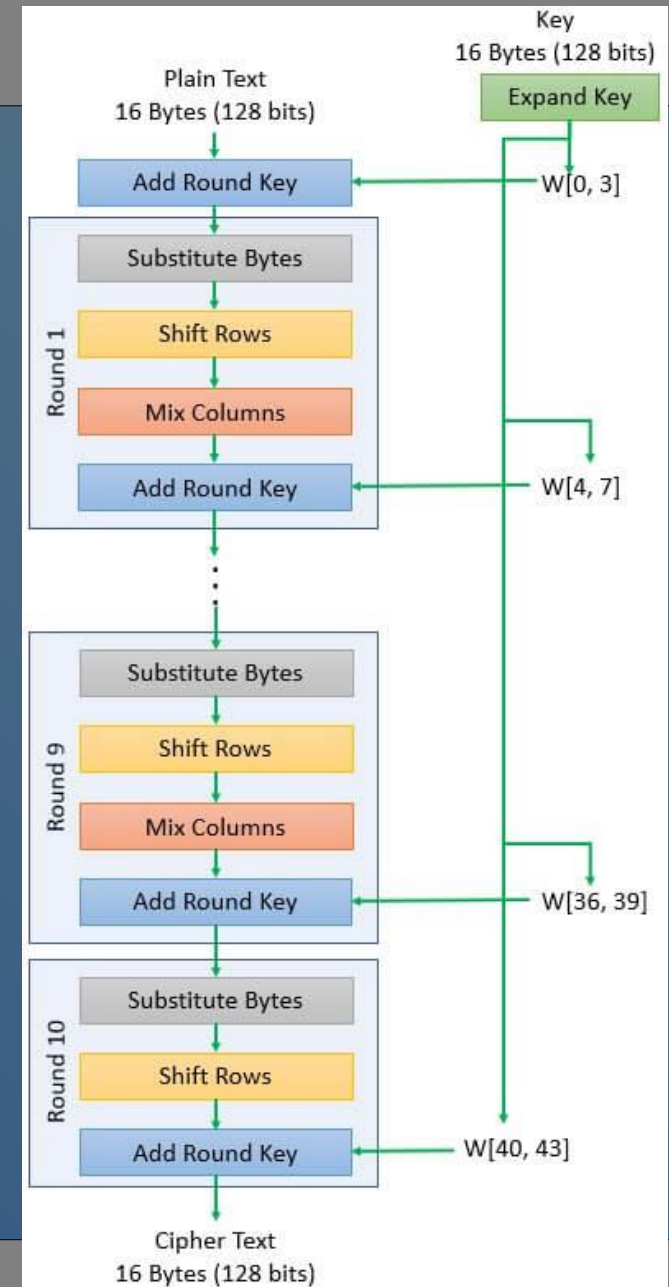
..

—..

—This was the only type of encryption prior to invention of public-key in 1970's.

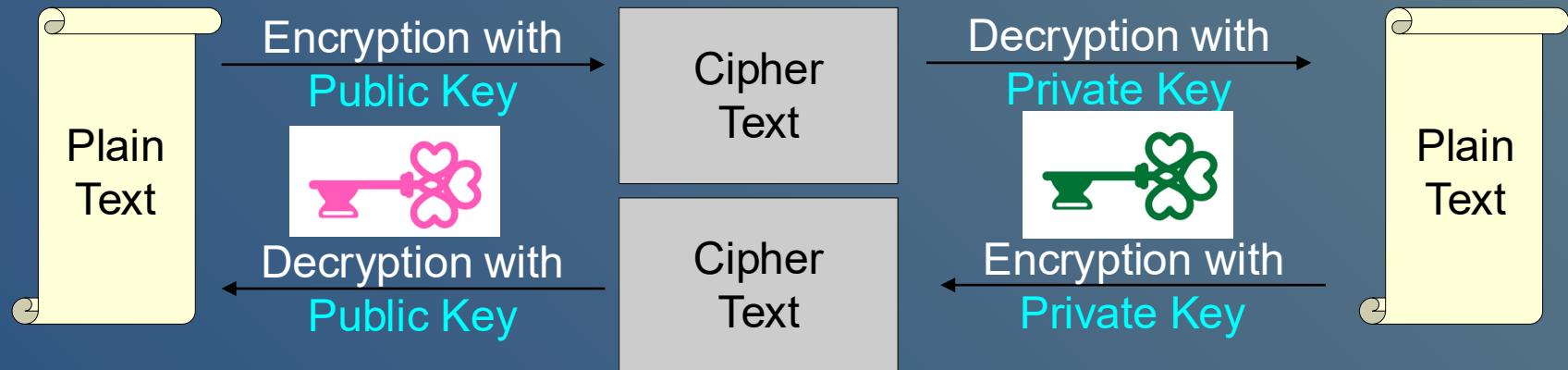
Private Key Crypto: AES

- AES is an example private key crypto algorithm
 - Need the same key to encrypt and decrypt.



Public Key Crypto
or
Asymmetric Crypto
(Two keys)

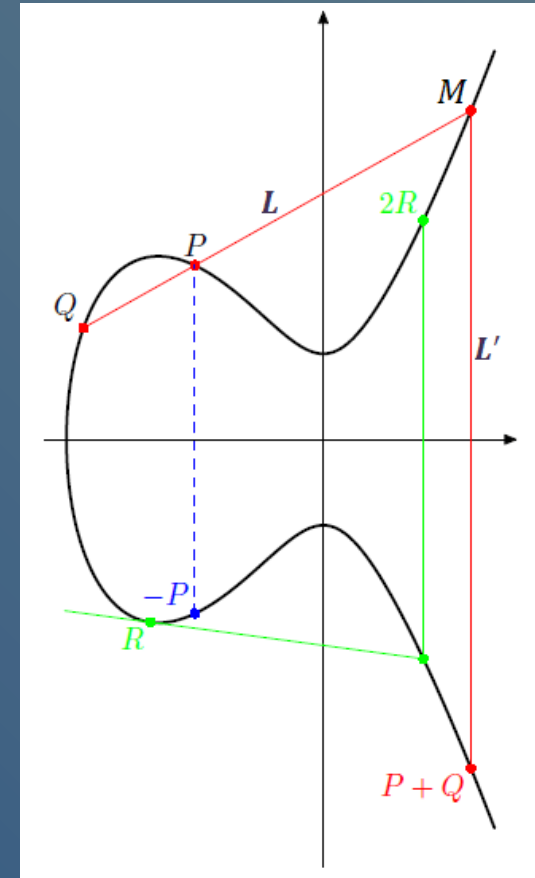
Public Key Crypto (Asymmetric)



- There are two keys:
 - Public key: can be known to anybody
 - Used to encrypt and verify signatures (more below).
 - Private key: ..
 - Used to decrypt and sign signatures (more below).
- Fundamental property of public key encryption:
 - ..

Generating Keys

- **Generating keys:**
 - The public and private keys are ..
- **Example approaches to generating keys**
 - Factoring very large prime numbers,
 - Solving "Twisted Edwards curves" (ed25519)



Keeping Secrets

- **Example: Keeping Secrets**

- Alice wants to send a secret message to Bob

- ..

- Bob decrypts the cipher-text using his **private key**

- **Analysis**

- Since only Bob knows Bob's private key, only Bob can decrypt the cipher-text.

- Hence Alice and Bob can securely share the message.

Verifying Sender

- **Example: Verifying Sender**

- Bob wants Alice to know that he sent a message and it has not been altered.

- ..

- Alice decrypts the cipher-text using Bob's **public key**.

- **Analysis**

- Since only Bob knows Bob's private key,

- ..

- Alice knows it was Bob who created the message.

Secret and Verified

- **Example: Secret and Verified**

Combine previous two examples.

– Alice wants to send a verified, secret message.

– ..

- Anyone can decrypt it with her public key.

- But only she can encrypt with it; so we know she sent it!

– ..

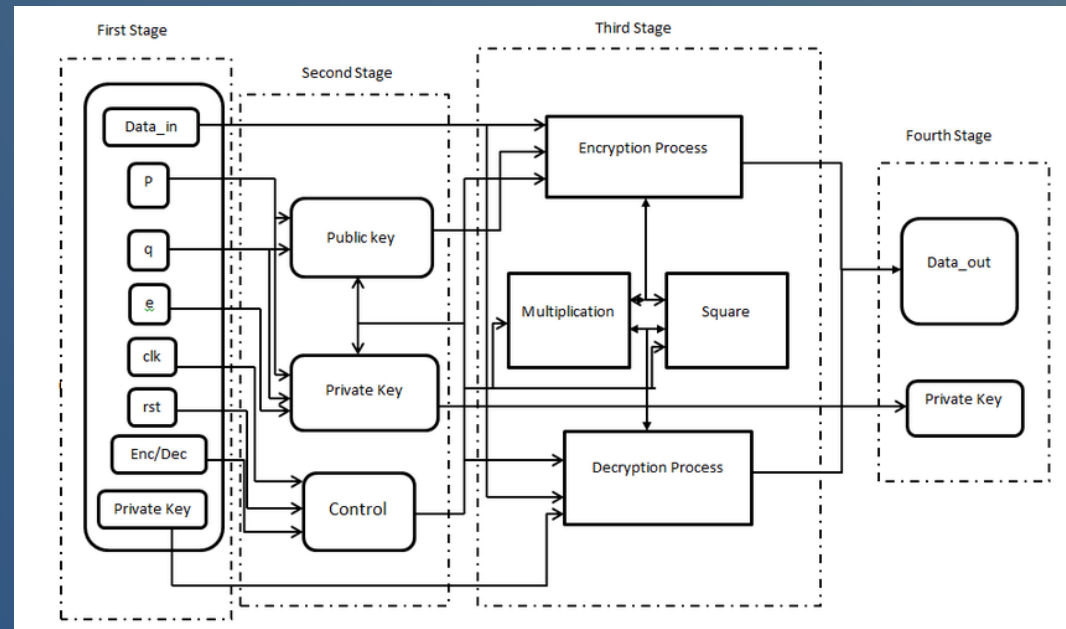
- Only Bob can decrypt it with his private key.

- **Analysis**

– Only Bob can decrypt the message (using his private key), and he'll know that only Alice can create it (using her private key).

Public Key

- **Benefit:**
 - This does not require having ..
 - Lots of other use cases beyond encryption / decryption
- **Example algorithm: RSA.**



Summary

- **Cryptography**
 - From **plain text**, create **cipher text** that others cannot read or change.
- **Types of algorithms**
 - 0 Keys: **Hash function**
 - 1 Key: **Symmetric encryption (private-key)**
 - Both sides know the **same secret key**.
 - 2 Keys: **Asymmetric encryption (public-key)**
 - You share a public key with the world.
 - Anyone can encrypt messages for you using this key.
 - **Only you can decrypt messages using your secret private key which matches the public key.**