Concepts of Cryptography

Brief Introduction

- The word "**cryptography**" derives from the Greek word for "**secret writing**".
- **Cryptography** is the science of communication over untrusted communication channels.
- **Cryptography** is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it.
- Historically, cryptography has been associated with spies, governments, military, and has been used in warfare for thousands of years.
- Over the past 50 years, cryptography has acquired a sound mathematical foundation, and has moved from military application to commercial applications.

Basic Terms

- Plaintext – the format (usually readable) of data before being encrypted
- Cipher text – the "Scrambled" format of data after being encrypted
- Cipher - an algorithm which is applied to plain text to get cipher text
- Encryption – the method of transforming data (plaintext) into an unreadable format.
- Decryption – the method of turning cipher text back into plain text
- Key – (crypto variable) values used in the encryption process to encrypt and decrypt

Terminologies

- Cryptanalysis - science of studying, breaking, and reverse engineering algorithms and keys.
- Cryptology - the study of secret codes or ciphers and the devices used to create and decipher them
- Cryptosystem – A system or product that provides encryption and decryption

Terminologies

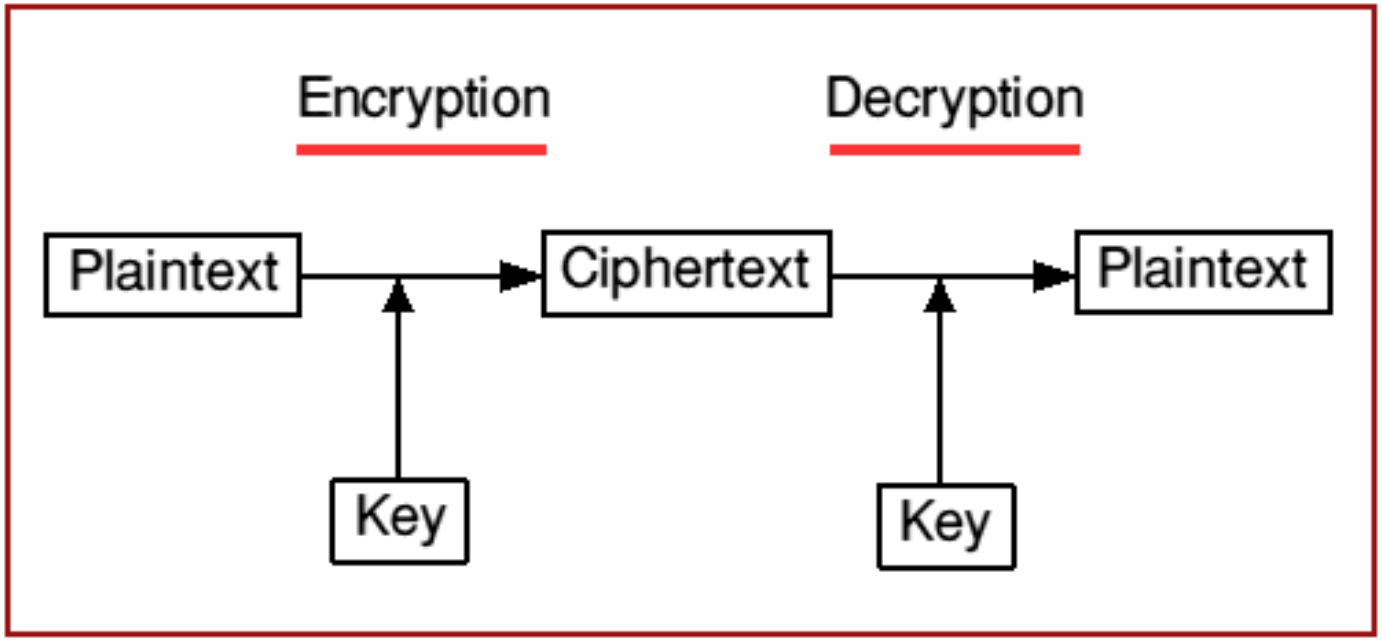
- KeySpace
  - range of values that can be used to construct a key
    - Larger keyspace provides for more possible keys so is stronger
    - 512 bits provides 2 to the power of 512 possible combinations

- The strength of the encryption comes not from keeping the algorithm secret – but from the size , randomness, and secrecy of the key!!!!!

Terminologies

- **Work factor**
  - Estimated time and resources to break a cryptosystem

Basic Process



How Does It Work?

- An ordinary message (the **plaintext**) is processed by an encryption algorithm to produce a scrambled message (the **ciphertext**).
- The receiver then uses a matching decryption algorithm to recover the plaintext from the ciphertext.
- There would be no security if these algorithms were known to everyone.
- Hence, there is an additional piece of input data called a **key**.
- The key is secret, even though many people may know the algorithms.
- The idea is the same as that of combination locks: Many people may use locks with the same design, but each one chooses a different combination (i.e., a different key).

A Motivating Example

- Consider an e-commerce scenario where Alice, a purchasing agent, wants to order some products from Bob, her supplier.
- Requirements for the transaction:

1. Alice wants to be sure that she is really dealing with Bob and not an impostor (**authentication**).
2. Bob wants to know that Alice is really Alice and not an impostor (**authentication**), because Alice gets special prices as negotiated.
3. Alice wants to keep the order secret from her competitors; and Bob does not want other customers to see Alice's special prices (**privacy**).
4. Alice and Bob both want to be sure that crackers cannot change the price or quantity (**integrity**).
5. Bob wants to ensure that Alice cannot later claim that she did not place the order (**non-repudiation**).

General Requirements

- **Authentication**: The sender knows that the message is going to the intended recipient; and the recipient knows that the message was sent by the proper sender.
- **Privacy**: The message is secret: only the sender and the intended recipient know its contents.
- **Integrity**: The message was not modified (intentionally or accidentally) while in transit.
- **Non-repudiation**: The author of the message cannot later deny having sent the message.
- Cryptographic techniques can be used to satisfy the above requirements.

Key Ideas-Confusion/Diffusion

Strong Ciphers have the following attributes

- **Confusion –** commonly carried out through substitution
- **Diffusion –** commonly carried out through transposition (mixing up characters in message)

Types of Encryption Ciphers

Substitution

– Replaces one letter with another

Transposition

- Move letters around

Substitution Ciphers

Eg: Caesar Cipher

It is a method in which each letter in the alphabet is rotated by three letters as shown

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

↓

D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Transposition

Transposition

Eg: Columnar Transposition

Plain Text

```
T H I S I
S A M E S
S A G E T
O S H O W
H O W A C
O L U M N
A R T R A
N S P O S
I T I O N
W O R K S
```

Cipher Text

```
T S S O H
O A N I W
H A A S O
L R S T O
I M G H W
U T P I R
S E E O A
M R O O K
I S T W C
N A S N S
```

Non Encryption Ciphers

- **Running Cipher** – doesn't use encryption, example. Find a certain book, turn to a certain page, then pick the letter from word 50 character 5.. An on and on to build a message.
- **Concealment Cipher** – a message within a message. Similar to running cipher but delivered in a single message.

Non Encryption Ciphers

Steganography

- The act of hiding data in plain site (in another form). Such that nobody knows the secret data is there.

Does NOT encrypt data.

Example: Gif image, every 100 pixels are altered such they represent a number. This number is a value to be combined with every other 10 pixel values to be a message. (Your eyes wouldn't detect the change in pixels)

Key Management

- Key lengths should be long enough to provide the necessary level of protection
- Keys should be stored and transported in a secure means.
- Keys should be extremely random and use the full spectrum of the key space.
- Keys lifetime should correspond with the sensitivity of the data to be protected.
- The more the key is used the shorter it's lifetime should be.
- Keys should be destroyed when their lifetime is at an end.

Cryptography history

Historical encryption algorithms

–Caesar cipher – just shift a few characters (A->F, B->G)
Eg: CipherText- GTTP, Key =5 , PlainText-?

–Vigenere Table – 2x2 matrix used for substitution

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Eg:
PlainText- HELLOGUYS
Key- SONG
KeyStream- SONGSONGS
CipherText- ZSYRGUHEK

Methods of Encryption

Methods of Encryption Overview

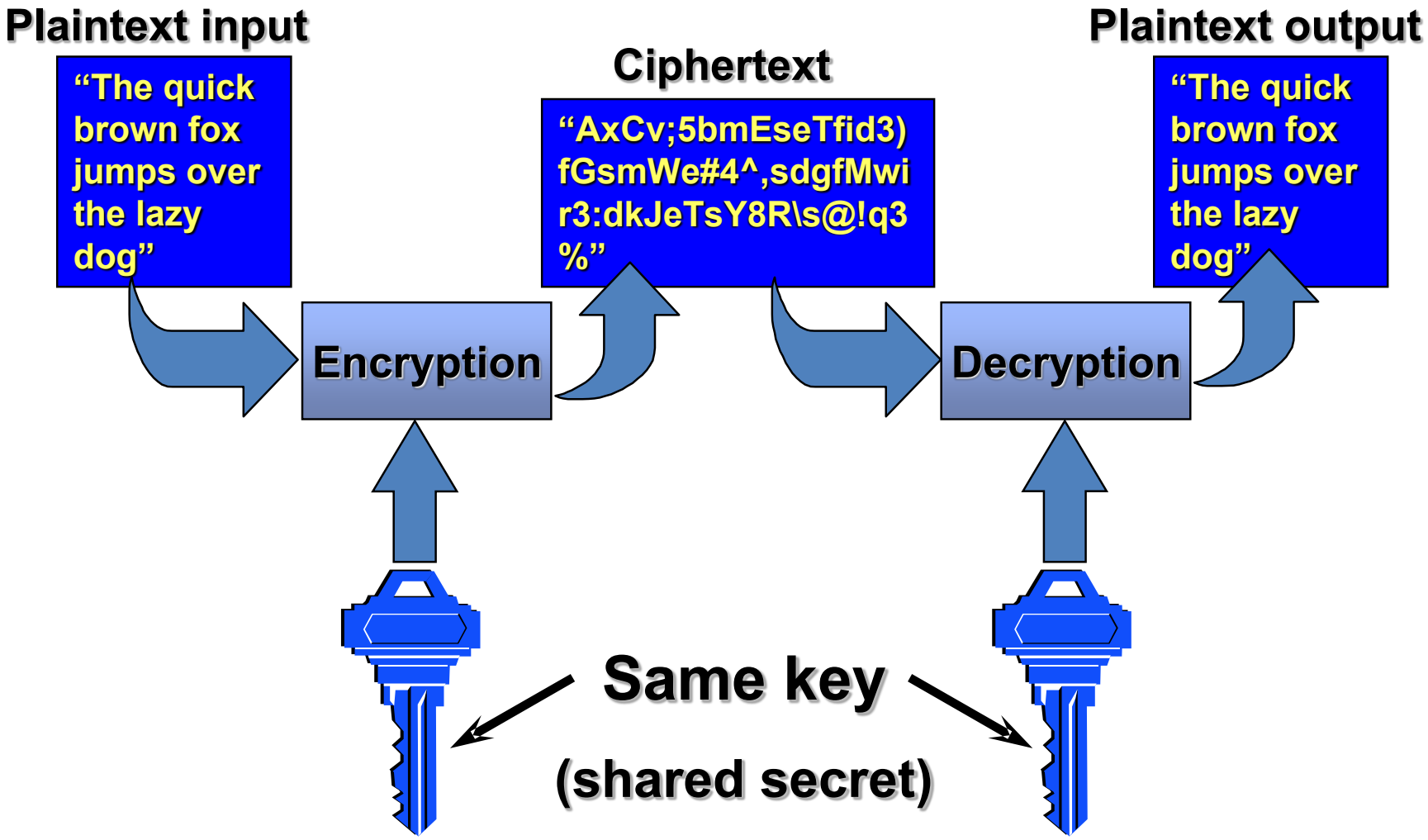
There are multiple “methods” of encryption

- Symmetric
- Asymmetric
- Hybrids
- Hashes (not really encryption, but no better place to put this)

Symmetric Encryption

Same key is used to BOTH encrypt and decrypt data

Symmetric Cryptography



Symmetric Pros

- Fast
- Hard to break if using a large key size
- Provides Confidentiality

Symmetric Cons

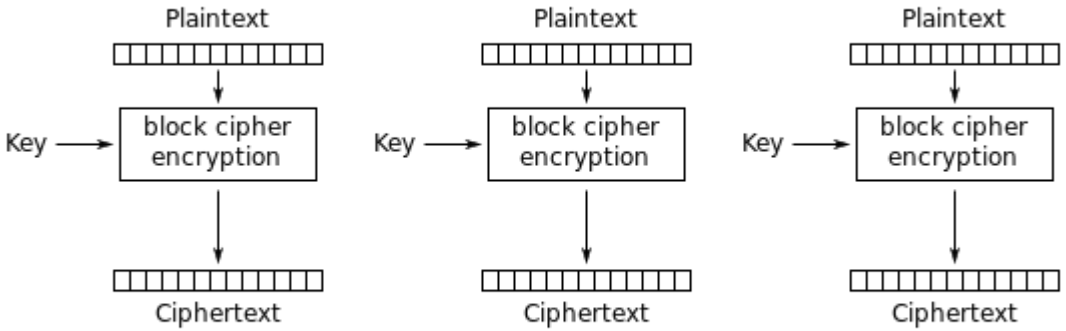
- Keys must be shared
  - This is difficult to really do? How do you get a key to someone you want to talk to?
  - Requires secure mechanism to deliver keys
  - Number of keys becomes needed becomes crazy large as number of people involved increases
  - Does Not provide Authenticity or Non-repudiation

Symmetric Cryptography

- Types:
  1. **Block Ciphers**
  2. **Stream Ciphers**

1.Block Ciphers

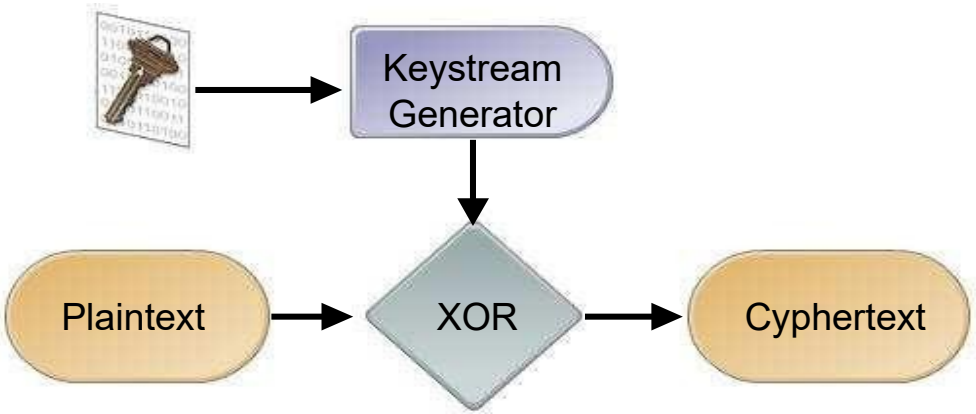
- Break down message into fixed sized blocks, equal to the size of the key.
- Encrypt each block with the key.



Electronic Codebook (ECB) mode encryption

2. Stream Ciphers

- Do not break into blocks, instead take one character of the message at a time.
- The“key” is used with a “key stream generator” to create a stream of bits.
- These bits are XORed with the plaintext to create cipher text



Specific Symmetric Key Cryptosystems (algorithms)

Specific Symmetric Cryptosystems

- DES
- Triple DES
- AES
- IDEA
- Blowfish
- RC4
- RC5
- RC6

DES general info

- Read history (on your own)
- Understand that DES is the "Standard" DEA is the actual algorithm.
- Retired when it became it was too easy to break.

DES

DES (Data Encryption Standard)

- Symmetric algorithm
- Block based algorithm
- 64 bit key, but only 56 of these are actually used by the algorithm.
- Divides the data into blocks and operates on them one at a time. These blocks are put through *16 rounds** (called an "S-box) of transposition (re- arranging) and substitution (changing) the order and type depends on the key.
- There are 5 "modes" of DES

DES Modes (overview)

- Electronic Code Book
- Cipher Block Chaining
- Cipher Feedback
- Output Feedback
- Counter Mode

ECB

- Electronic Code Book – “regular” type of encryption, straight forward block by block encryption.
- Given the same plain text and the same key, the resulting cipher text will always be the same

Cipher Block Chaining

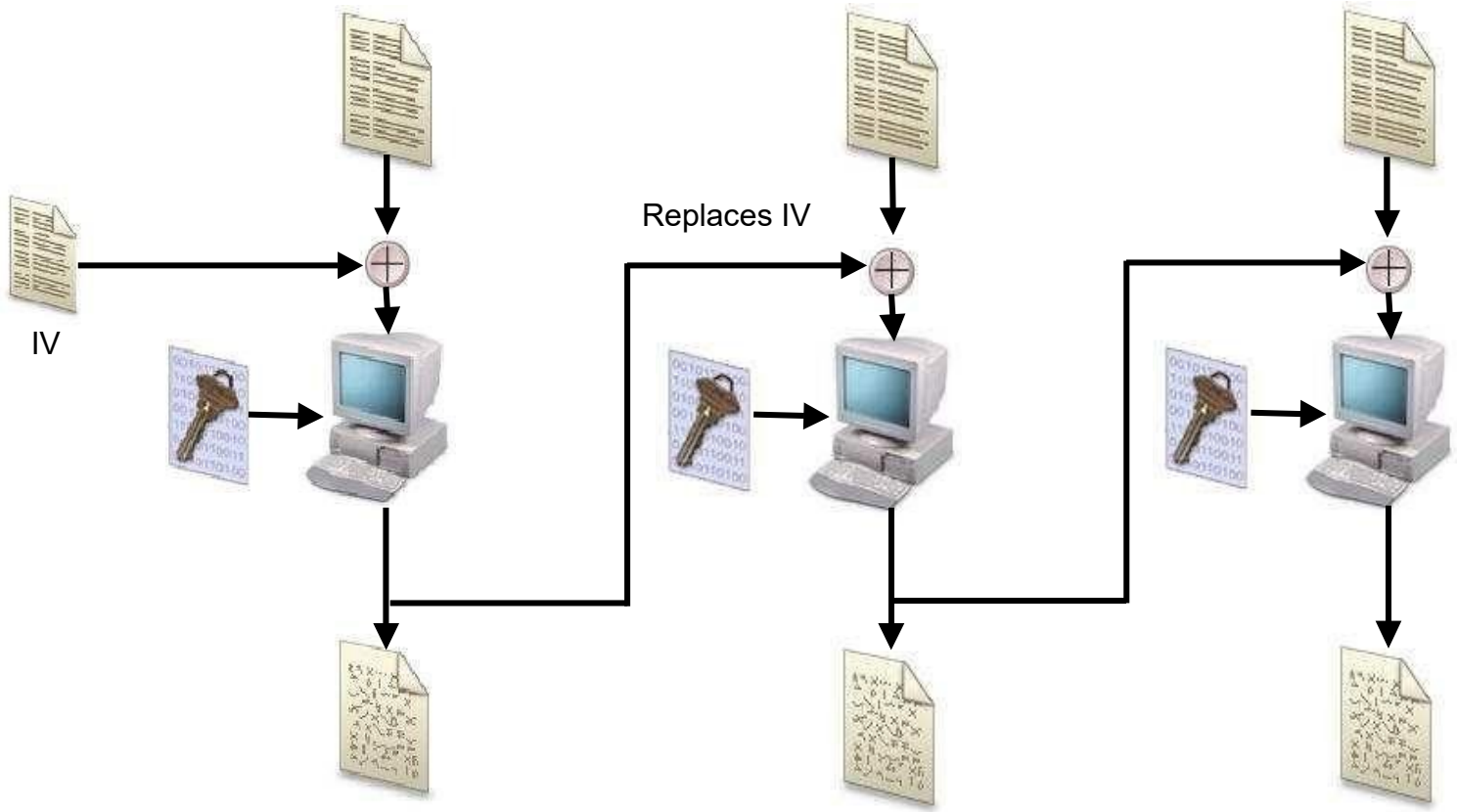
Tries to solve the problem of ECB mode.

- For each block of data to encrypt, CBC uses not only the key but the results from the previous block.
- For the first block (since we don't have results from a previous block) we use an "Initialization Vector"

(see diagram on next page)

CBC diagram

CBC



Other DES Modes

Cipher Feedback, Output Feedback, Counter

Triple DES

Like DES but uses *48 rounds* rather than DES's 16 rounds.

Has 4 variants

- DES-EEE3 – 3 different keys: data is Encrypted, Encrypted, Encrypted (one key for each)
- DES-EDE3 – 3 keys: Encrypted, Decrypted, Encrypted
- DES- EEE2 – 2 keys, first and last operation use the same key
- DES- EDE2 – 2 keys, first and last operation use the same key

AES

Advanced Encryption Standard – Developed to replace DES. There were multiple algorithms proposed to become "DES" the one chosen was called *Rijndael*.

- Block cipher
- Works well in software or hardware
- Low memory requirements
- Replaces DES
- Supports block sizes of 128, 192, 256 bits

IDEA

International Data Encryption Algorithm

- Block cipher
- 64 blocks of data
- 128 bit key
- IDEA is faster than DES when implemented in software
- Used in PGP (later)
- patented

Blowfish

- Block cipher
- 64 blocks of data
- Key length can be 32 – 448
- 16 rounds of cryptographic functions
- Unpatented, anyone can use it

RC4

- Owned by “RSA”
- Stream cipher
- Variable key size (40– 2048 bits)
- Used in SSL and in WEP (wireless) encryption
- Simple, fast and efficient
- Also called ARC4

RC5

- Owned by RSA
- Block cipher
- Block sizes of 32, 64, or 128 bits
- Key size can go up to 2048 bits
- “rounds” are not fixed, can be up to 255

RC6

- Same attributes at RC5, but modified to be faster
- Owned by RSA
- Block cipher
- Block sizes of 32, 64, or 128 bits
- Key size can go up to 2048 bits
- “rounds” are not fixed, can be up to 255
- faster than RC5

Asymmetric Encryption

Asymmetric Encryption

Rather than use the same key for encryption and decryption, you use a different key for encryption and decryption

- These keys are mathematically related to each other

  These keys are called

  - Private Key
  - Public Key

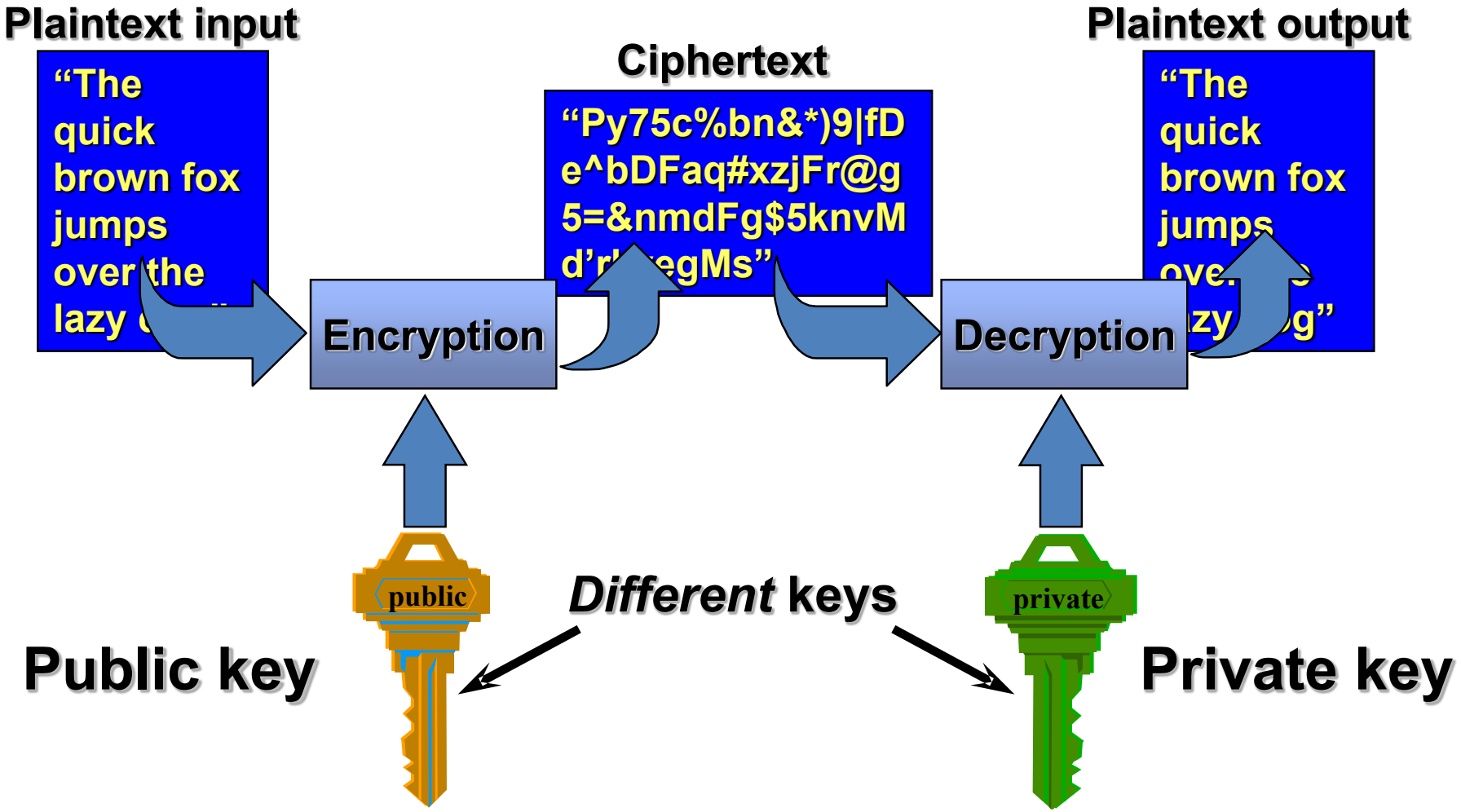
Asymmetric Encryption

Public Key – given to everyone

Private Key – stays secret

Asymmetric Cryptography

Use public key to encrypt a message, private key can decrypt

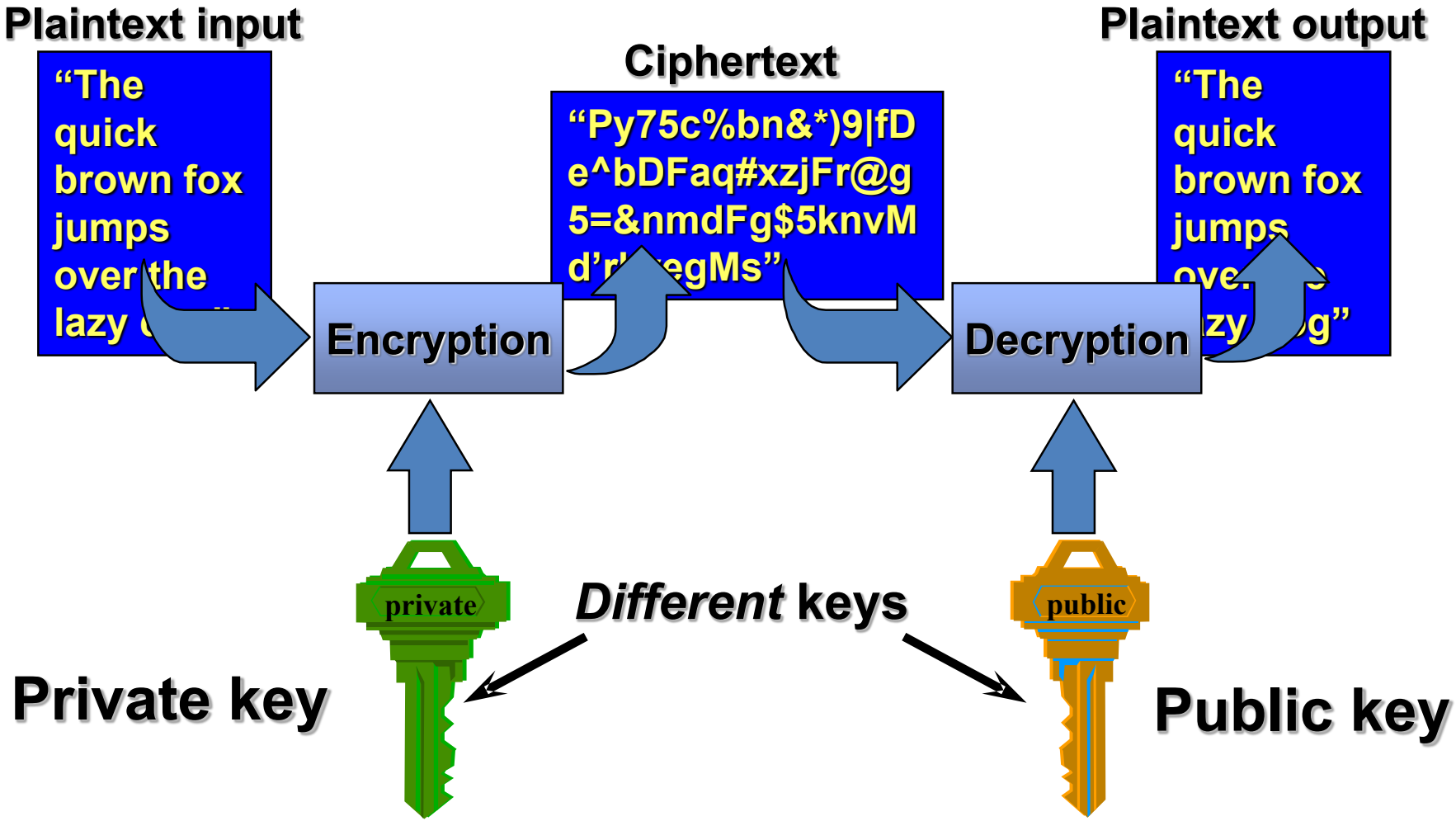


Asymmetric Encryption

Private and Public keys can actually do the reverse, you can use the private key to encrypt plaintext then the resultant cipher text can only be decrypted by the corresponding “public key”

Asymmetric Cryptography

Use private key to encrypt a message, public key can decrypt



Signing

This process of using a private key to encrypt something that can only be decrypted with your public key is call “signing” and is used for authentication and non-repudiation

- If someone can read something you signed it proves that your private key was used.

One way function

An important concept in symmetric encryption is a “One way function”

A one way function is an operation that is faster to complete in one direction than the other.

Example: if you drop a glass it breaks instantly to “undo” this would take much more time.

Asymmetric algorithms use this concept

One way functions

- With Asymmetric encryption, a message is encoded with a one way function. This function supplies a trapdoor* (knowledge of how to undo the one way function faster). The private key can be used to retrieve this “trapdoor” and then use the trapdoor to put things back in order.
- Asymmetric algorithms use mathematical operations that are easier to do in one direction, than the other.

Asymmetric Pros/Cons

Pros

- Key distribution is easy
- Scalable due to that
- Can provide authentication and non- repudiation

Cons

- Very mathematically intense
- Slow due to that

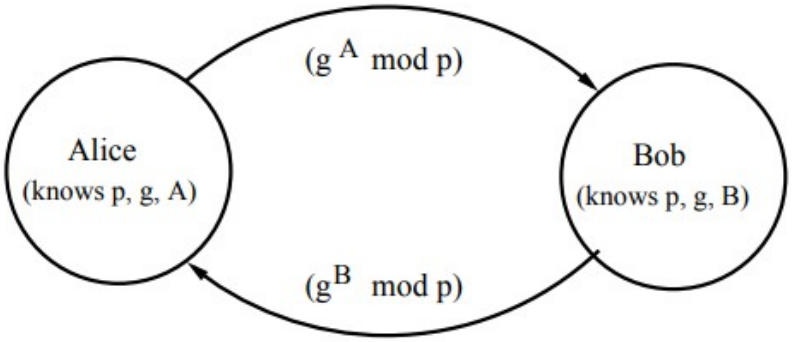
Specific Asymmetric Cryptosystems

Specific Asymmetric Cryptosystems

- Diffie-Hellman
- RSA
- Elliptic Curve Cryptosystem

Diffie-Hellman Key Exchange

- Developed to address shortfalls of key distribution in symmetric key distribution.
- The Diffie-Hellman key agreement protocol (1976) was the first practical method for establishing a shared secret over an unsecured communication channel.
- The point is to agree on a key that two parties can use for a symmetric encryption, in such a way that an eavesdropper cannot obtain the key.



RSA

- Can be used for digital signatures, key exchanges, and encryption
- Security comes from the difficulty of factoring larges numbers.
- Private and Public keys are functions (results of mathematical operations) of large prime numbers.

Elliptic Curve Cryptosystem

- Used for digital signatures, encryption and key distribution
- Uses one-way function with discrete logarithms of elliptic curve.
- Requires fewer computational sources as shorter keys can be used
- Often used on lower-power devices

Hybrids

Hybrids cryptosystems use both Asymmetric and symmetric key cryptosystems.

- Use a Asymmetric system to encrypt a key to a symmetric key system. (i.e. to distribute the key).
- The Symmetric key is used to actually perform the encryption.
- This key is called a “session key” and is only used for the current conversation.

Hashes

Hashes

A mathematical function that takes variable length input and produces a fixed length string.

Hash

- Since hashes take any length input and produce a fixed output, there will be multiple inputs that produce the same output, this is called a *collision**.
- A good hash function should not make it predictable on how to “force” a collision. Otherwise you could create a message what would generate the same hash as another
  - Collision Attack

    If two inputs a and b such that $H(a) = H(b)$, and $a \neq b$.

Hash

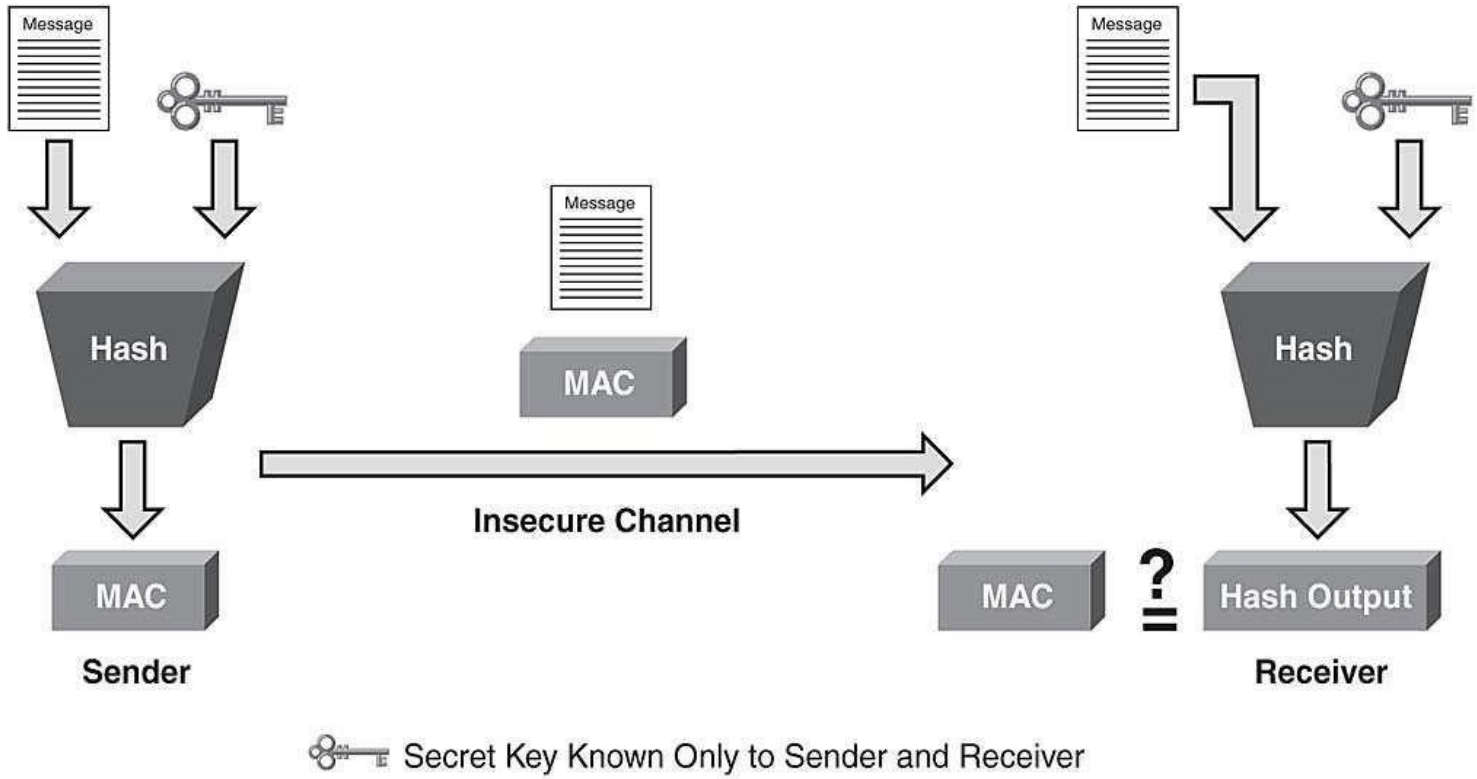
- Provide integrity, not confidentiality or authentication
- Hashes are vulnerable to man in the middle attacks

HMAC

HMAC – uses a secret key in combination to a hash algorithm to verify that a hash is not tampered with.

Rather than just doing the “hash algorithm” on the message, append your secret key to the message to create a new message and run the hash on the new message. The returned value is called a MAC (Message Authenticating Code)

HMAC


Message
Hash
MAC
Sender
Message
MAC
Insecure Channel
Message
Hash
MAC
?
Hash Output
Receiver
Secret Key Known Only to Sender and Receiver

HMAC

- Provide integrity and data original authentication
- Does not provide confidentiality

Specific Hash algorithm

Specific Hash algorithms

- MD2
- MD4
- MD5
- SHA

MD2

- Creates a 128 bit hash value, slower than MD4 and MD5

MD4

- creates 128 bit hash value
- Faster than MD2

MD5

- Creates 128 bit hash value
- More complex than MD2 and MD4
- More secure, harder to determine how to force collisions for a specific message

SHA

- Designed to be used with the Digital Signature Standard, (for use with digital signatures)
- Creates 160 bit hash values
- Modified version of SHA = SHA-1
- Alternate versions
  - SHA-256 = 256 bit hash values
  - SHA-385 = 384 bit hash values
  - SHA-512 = 512 bit hash values

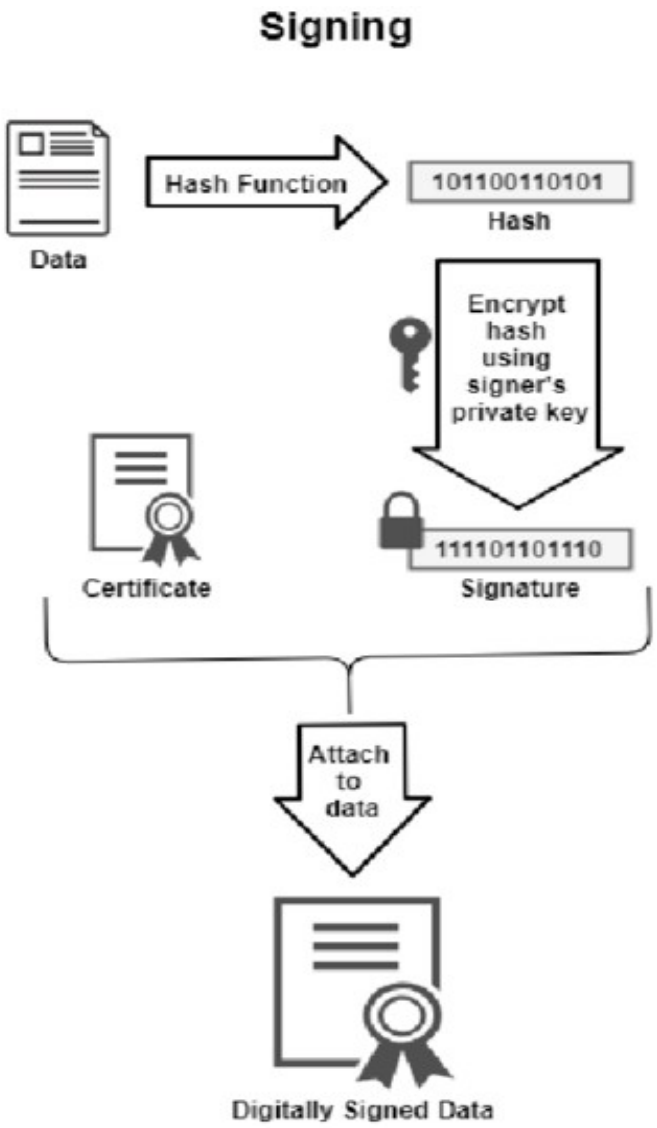
Attacks against Hashes

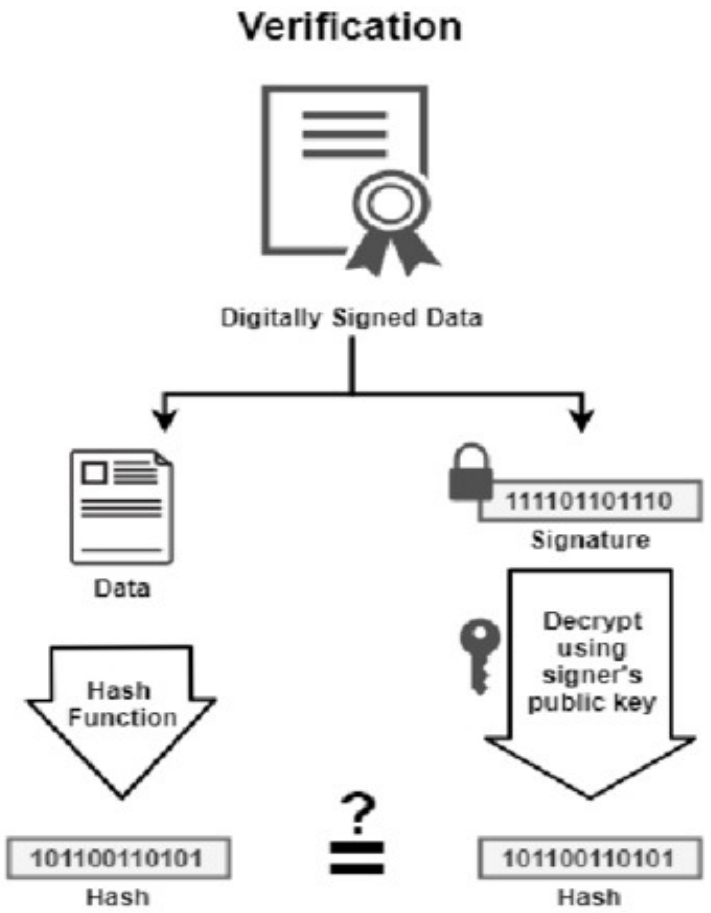
- Collisions – figure out how to create a message with the same hash value (collision)
  - Ex. “I’d like to buy 100 units of the widget” => A3BT
  - What if I could make the messages “I’d like to buy 500 units of the widget” and have the same hash value “A3BT” I can beat the integrity constraint
- This is called a birthday attack

Digital Signatures

- We can use Asymmetric Cryptography and Hashes. To provide message authenticity, and Integrity and Non-repudiation

Digital Signature





If the hashes are equal, the signature is valid.

Services provided by the Algorithm

Algorithm Type	Encryption	Digital Signature	Hashing Function	Key Distribution
Asymmetric Key Algorithms				
RSA	X	X		X
ECC	X	X		X
Diffie-Hellman				X
El Gamal	X	X		X
DSA		X		
LUC	X	X		X
Knapsack	X	X		X
Symmetric Key Algorithms				
DES	X			
3DES	X			
Blowfish	X			
IDEA	X			
RC4	X			
SAFER	X			
Hashing Algorithms				
Ronald Rivest family of hashing functions: MD2, MD4, and MD5			X	
SHA			X	
HAVAL (variable-length hash values using a one-way function design)			X	

Thank You