

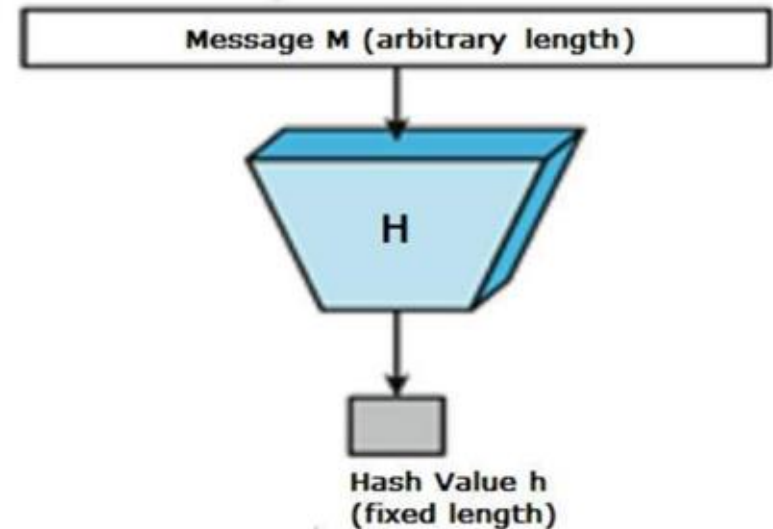


Basics of Cryptography for Blockchain (part 2)

Cryptographic hashing

A cryptographic hash function H is a mathematical function that converts any input message (M) into a hash value (h) of fixed length

$$h = H(M)$$



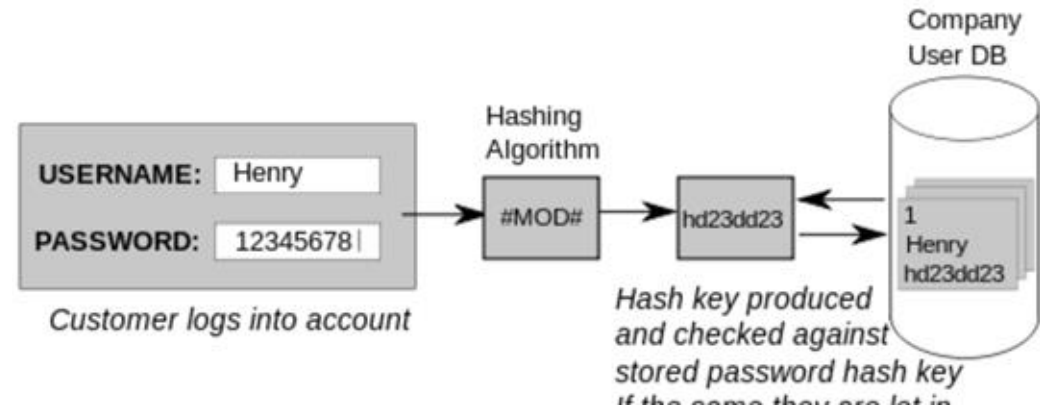
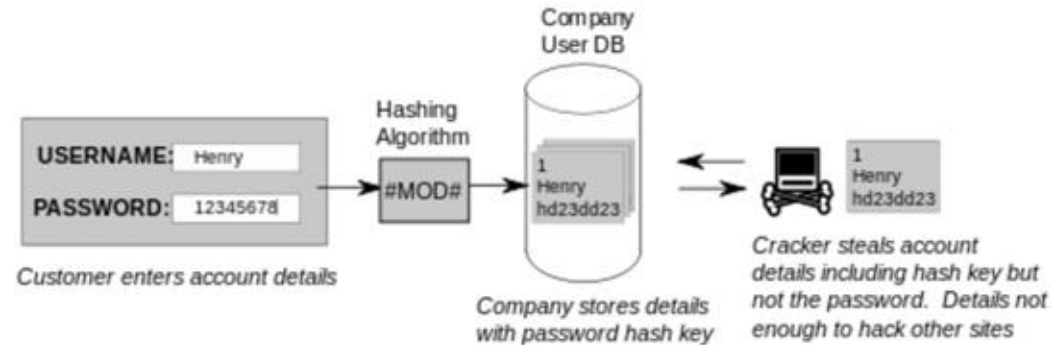
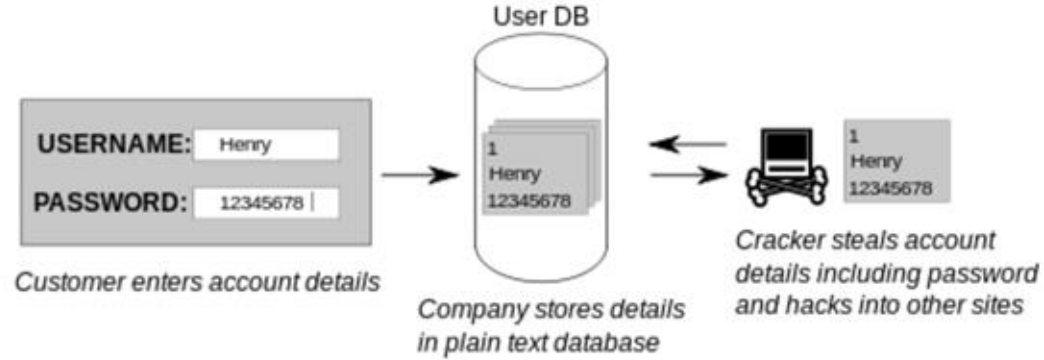
Encrypted passwords


It is not desirable that providers store our passwords in plain text (can be stolen)

Providers store hash of passwords

–Attackers will steal, but stolen info is unusable
(H function cannot be inverted!)

Hashing functions used each time user logs in to verify login information

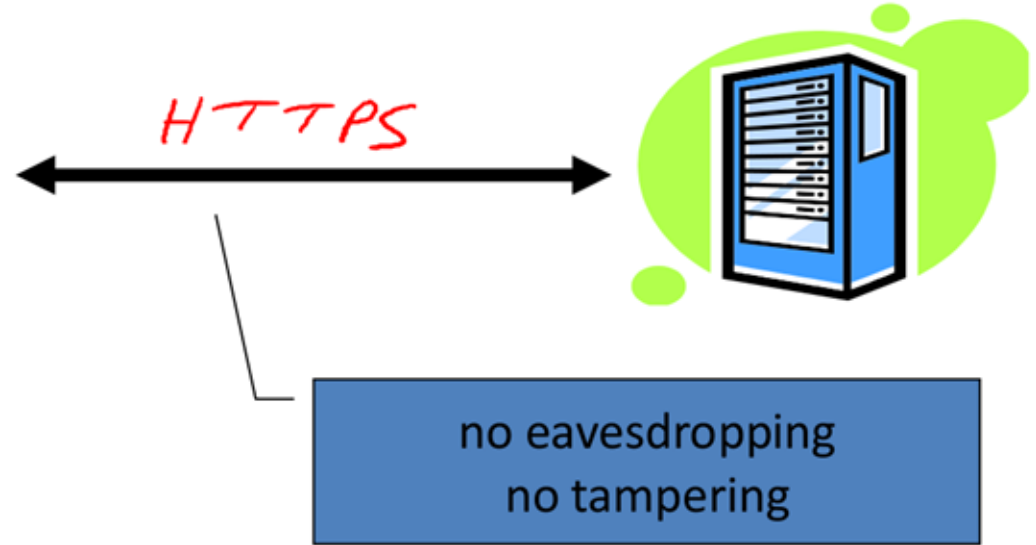
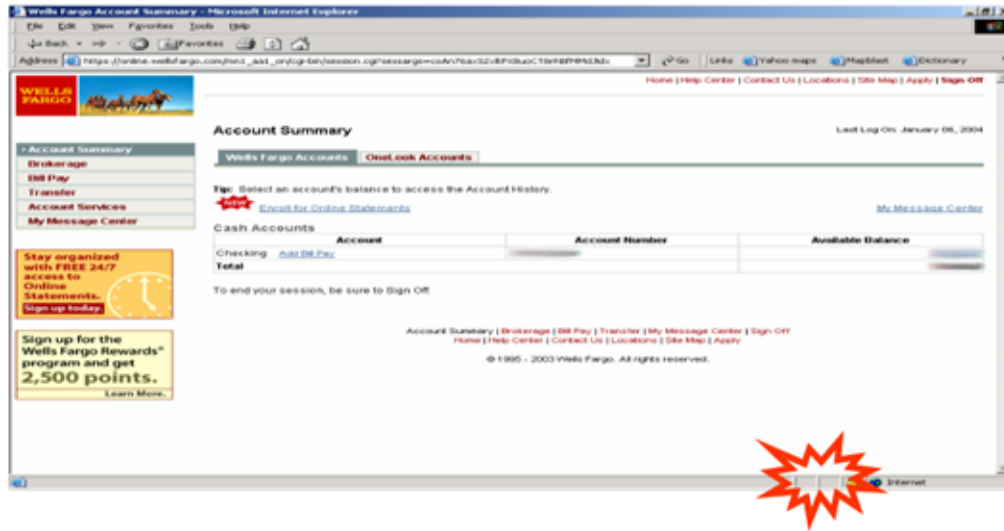




Encryption

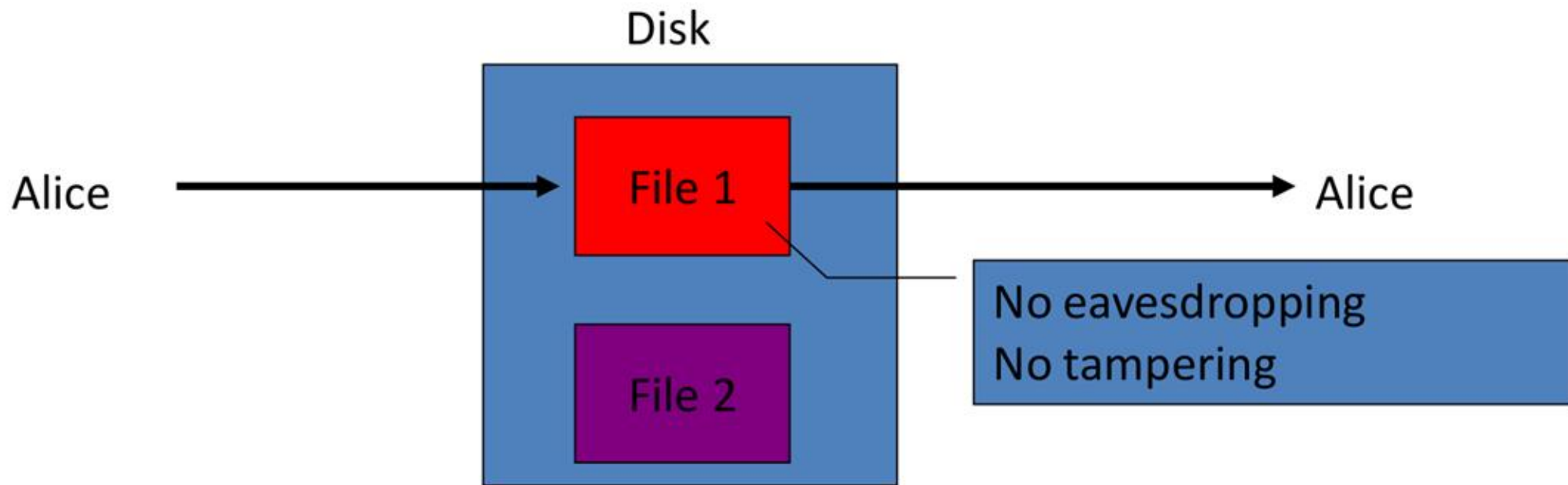
(symmetric, asymmetric)

Secure communication



Protected files on disk

Same as secure communication
(Alice sending a message to her future self)



What is encryption?

Encryption is the science of establishing and performing secure communications among computer systems

Symmetric encryption

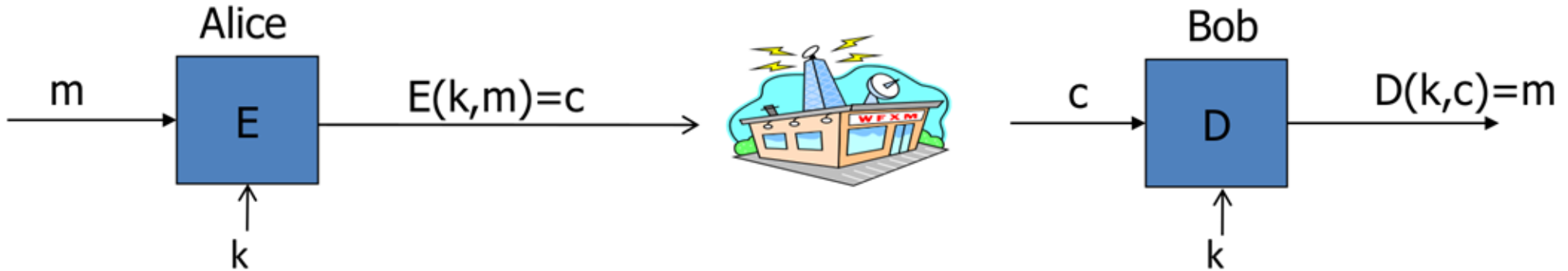
Sender and receiver use the same “key” to encrypt/decrypt messages

Asymmetric encryption

Different keys to encrypt/decrypt

Symmetric encryption

- .The same key k used for encryption and decryption of m
- .The algorithms used for E/D are usually called “cypher”



E, D : cipher k : secret key (e.g. 128 bits)

m, c : plaintext, ciphertext

What is a cypher?

A cypher defined over (K, M, C) is a pair of “efficient” algorithms (E, D) , where:

$$E: K \times M \rightarrow C$$

$$D: K \times C \rightarrow M$$

such that, $D(k, E(k, m)) = m$, for all m in M and k in K

E is often “randomised”, D is (obviously) always deterministic

M, C, K are symbol sets (for messages, cyphered-messages, and keys, respectively)

$\{a, b, c, \dots, x, w, z\}$

$\{0, 1\}^n$ [the set of all bit sequences of length n]

Stream v. Block cyphers

Stream cypher

Convert one symbol of plaintext directly into one symbol of cyphertext

Block Cypher

Convert groups (blocks) of plaintext symbols into groups (blocks) of cyphertext symbols

(Modern encryption mechanisms mostly use block cyphers, stream cyphers are easier to exemplify in this lecture)

Caesar cypher

- An example of “substitution cypher” (stream cypher)
- Used by Julius Caesar to secure war strategy communications
- Shift symbols of n (= the key) positions in the alphabet
- Example: $n = 3$ on English alphabet

-A → D

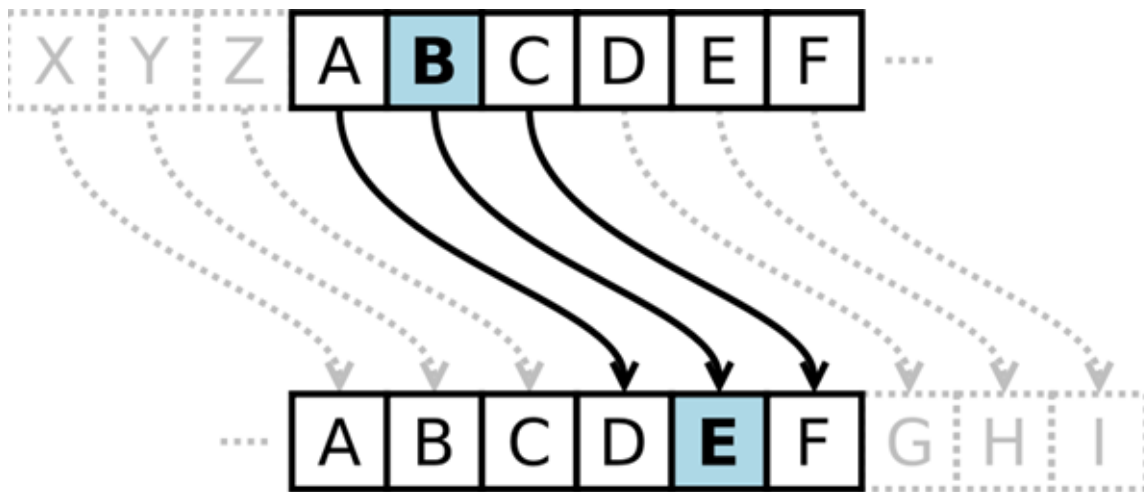
-B → E

-C → F

-...

-Y → B

-Z → C



Caesar cypher - exercise

C = “wklv phvvdjh lv vhfuhw”, K=3

M =?

Caesar cypher?

Is it secure?

Caesar Cypher - Is it secure?

- Not really
- Can be cracked using cyphertext-only attacks
- Simple case – we know C is encrypted using Caesar Cypher
 - Try all possible keys (how many?)
 - One of them will turn C into M that makes sense :)
- What if we do not know that C is encrypted using Caesar Cypher?
 - We can figure it out by looking at distribution of letters

OTP – One Time Pad

$$M, K, C = \{0,1\}^n$$

Given a message M of size n and a key K of size n , C is the bit-by-bit XOR of M and K

XOR

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

$M = 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0$

$K = 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0$

$C = ??$

OTP – is it a cypher?

• Yes, $D(k, E(k, m)) = m$

ENCRYPT

$$\begin{array}{rcl} & 00110101 & \text{Plaintext} \\ \oplus & 11100011 & \text{Secret Key} \\ \hline = & 11010110 & \text{Ciphertext} \end{array}$$

DECRYPT

$$\begin{array}{rcl} & 11010110 & \text{Ciphertext} \\ \oplus & 11100011 & \text{Secret Key} \\ \hline = & 00110101 & \text{Plaintext} \end{array}$$

OTP – is it secure?

- Yes, it is actually impossible to recover M from C only
 - In other words, no cyphertext-only attacks to OTP are possible
 - Applying Shannon's "Perfect secrecy" theorem
- OTP seems good, but has one major practical problem, which one?

OTP

- K must have same length as M
 - Alice and Bob should securely exchange K before communicating...
 - ...but if they can do that, why don't they exchange M directly?
- (Shannon's perfect secrecy can be guaranteed only if $|K| > |M|$)

Block cyphers

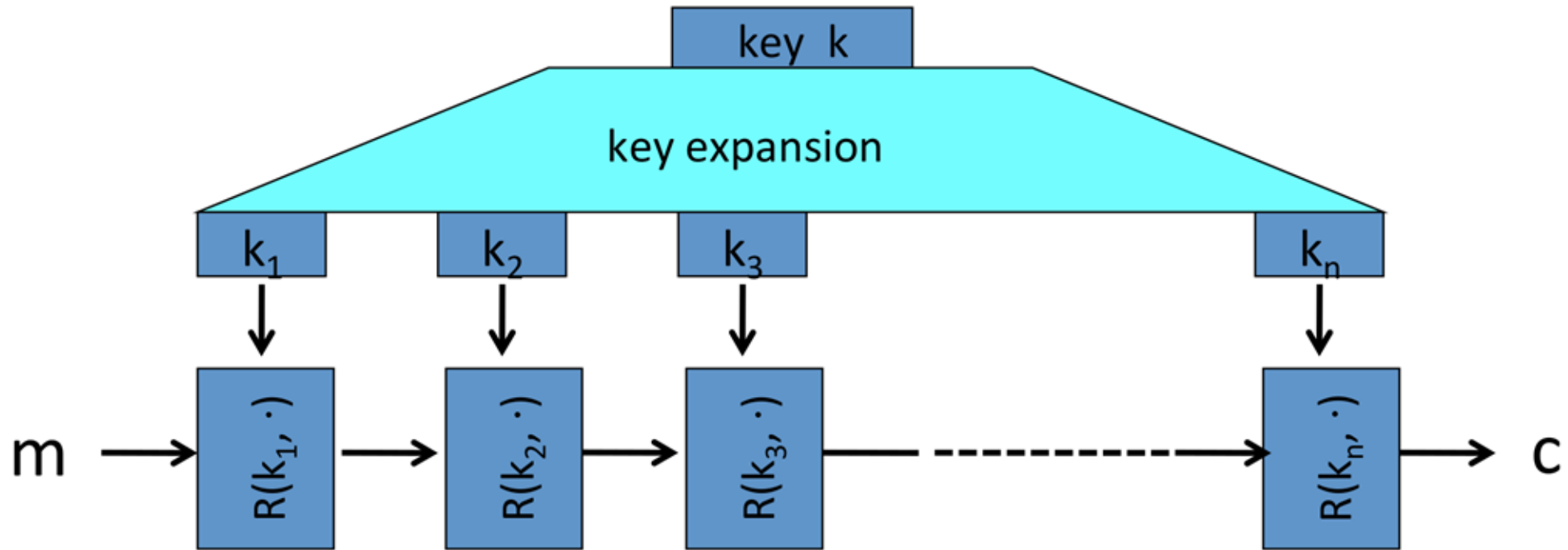
- .General architecture

- .Examples

 - 3DES

 - AES

Block cyphers – built by iteration



for 3DES ($n=48$), for AES-128 ($n=10$)

Stream cyphers

.Advantages

- Speed

- Low error propagation (error in encrypting one symbol does not affect next symbols)

.Disadvantages

- Low diffusion (all information about one plaintext symbol contained in its cypher)

- Easy to tamper with (an interceptor who breaks the algorithm can easily include new symbols)

Block cyphers

.Advantages

- High diffusion (information about one plaintext symbol diffused into several cyphertext symbols)
- Immunity to tampering (difficult to insert new symbols without knowing the past)

.Disadvantages

- Slow (an entire block must be accumulated before encryption can start)
- Error propagation (an error in one symbol may corrupt an entire block)

What did we learn?

- .In symmetric encryption the same key K is used to encrypt and decrypt M
- .Combination of E/D algorithms is called “cypher”
- .Stream cyphers and block cyphers
- . $|K| > |M|$ guarantees perfect secrecy, but does not work in practice :)



Asymmetric encryption

Asymmetric encryption

.Until now, in a cypher the same key K used for encrypting and decrypting M

.What if we could define a cypher that used two different keys K_1 and K_2 somehow mathematically related such that

- One key used for encrypting
- One key used for decrypting

(let's call the 2 keys "public" and "private")

RSA Cryptosystem

- Invented by Ron **R**ivest, Adi **S**hamir, and Len **A**dleman
- Still widely adopted
- Specified by two separate procedures
 - Generation of key pairs
 - Encryption and Decryption

RSA – Generation of Key Pair

.Each agent who desires to communicate using asymmetric encryption needs a pair of keys (public and private)

.In RSA, the public key is generated as follows:

- Select 2 large prime numbers, p and q

- Calculate $n=p*q$
(for strong encryption, let n be large, typically at least 512 bits)

- Find e , such that $e > 1$, $e < (p-1)*(q-1)$, with e and $(p-1)*(q-1)$ “coprime”

- .No common factors (except 1) between e and $(p-1)*(q-1)$

- (n,e) is the RSA public key, which is made public

RSA – Generation of key pair

Private key is (n,d) , where d is calculated from p,q , and e as follows:

$$-e*d = 1 \text{ [mod } (p-1)(q-1)]$$

--- the remainder of $(e*d)/((p-1)*(q-1))$ must be 1 ---

– d is unique for given (n,e)

• Extended Euclidean Algorithm can be used to find d

– (n,d) is the private key

RSA key pair generation – example

•Public Key

– $p = 7$, $q = 13$, then $n = p * q = 7 * 13 = 91$

– $e = 5$ is a valid choice, because there is no common factor between 5 and $(p-1)*(q-1) = 6*12 = 72$

– **$(n, e) = (91, 5)$** is the public key

•Private key

– $d = 29$

– $e * d = 1 \pmod{(p-1)(q-1)}$
 $5 * 29 = 145$; $145 / 72 = 2$; $145 \bmod 72 = 1$

– **$(n, d) = (91, 29)$** is the private key

RSA Encryption/decryption

•Public key is $(n,e) = (91, 5)$

•Let's say message $m = 9$, what is the cyphertext c ?

$$c = m^e \bmod n = 9^5 \bmod 91 = 59049 \bmod 91 = 81$$

•Private key is $(n,d)=(91,29)$ used for decryption

$$m = c^d \bmod n = 81^{29} \bmod 91 = 2.21 \times 10^{55} \bmod 91 = 9$$

RSA Analysis

.Encryption

- It is considered one-way (not invertible), because it is virtually impossible to invert modulo functions
- Modulo functions, in fact, map many input values to the same output value

.Key generation

- It is very hard to guess the private key from the public key, since it is equivalent to factoring n , which is very complicated if n is obtained from p and q large prime numbers

Elliptic curve cryptosystem

- .Elliptic Curve Cryptography (ECC) is based on the complexity of solving the “discrete logarithm problem”
 - It does not use the modulo function
- .Bitcoin uses ECC
- .More details provided while presenting Bitcoin

Applications of asymmetric encryption

- .Public key encryption

- .Digital signatures

Public key encryption

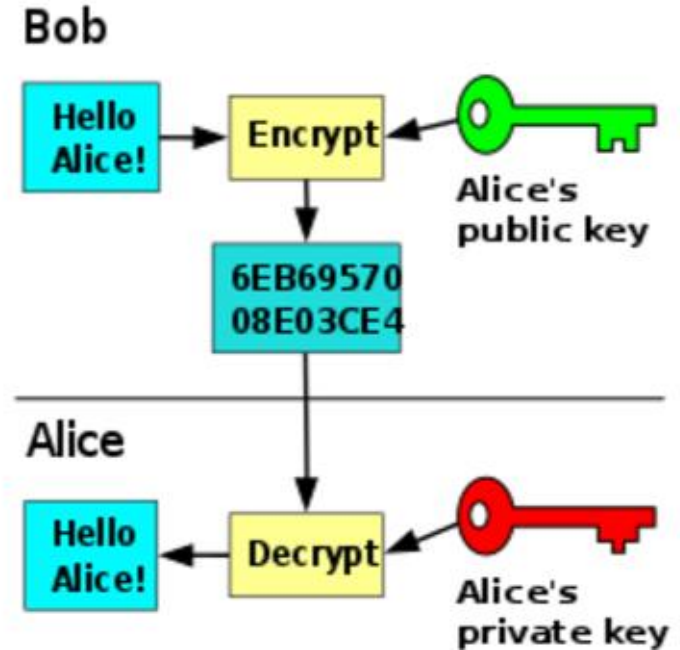
.Public Key for **encrypting** messages

.Private Key for **decrypting** messages

-Alice distributes her public key to the world

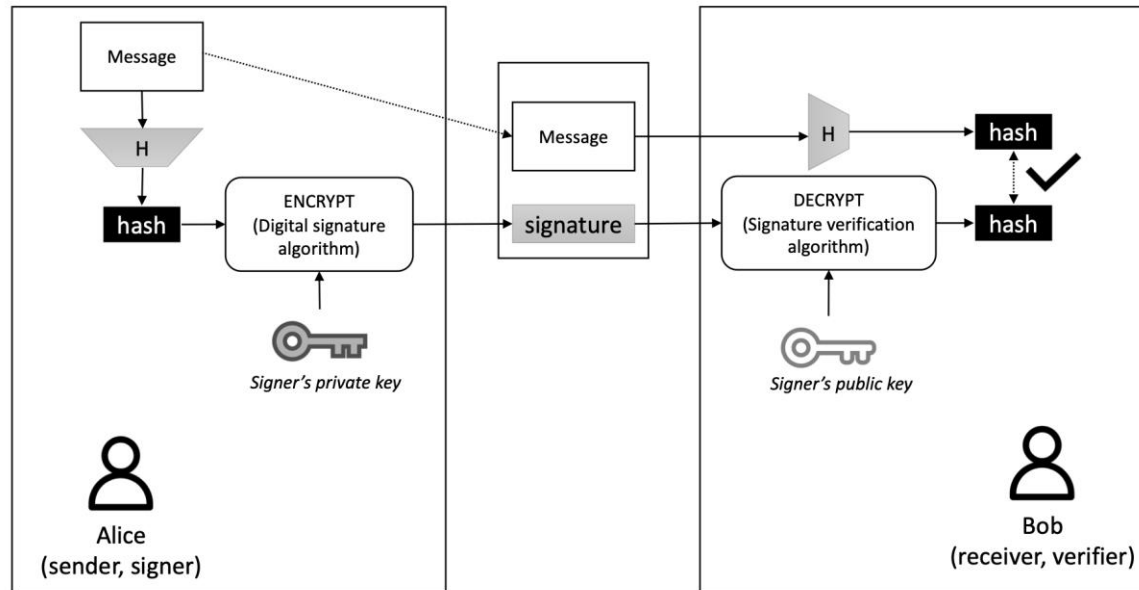
-To send a message to Alice, anyone (such as Bob) encrypts messages with Alice's public key

-Only Alice can decrypt messages with her own private key



Digital signatures

- Alice (signer) needs to send a message to Bob (verifier)
- Bob wants to be sure that the message M he receives was sent by Alice
- Alice wants to be sure that Bob cannot repudiate her as the source of M



More notes

•Signature is the hash of the message encrypted using the signer (sender) private key

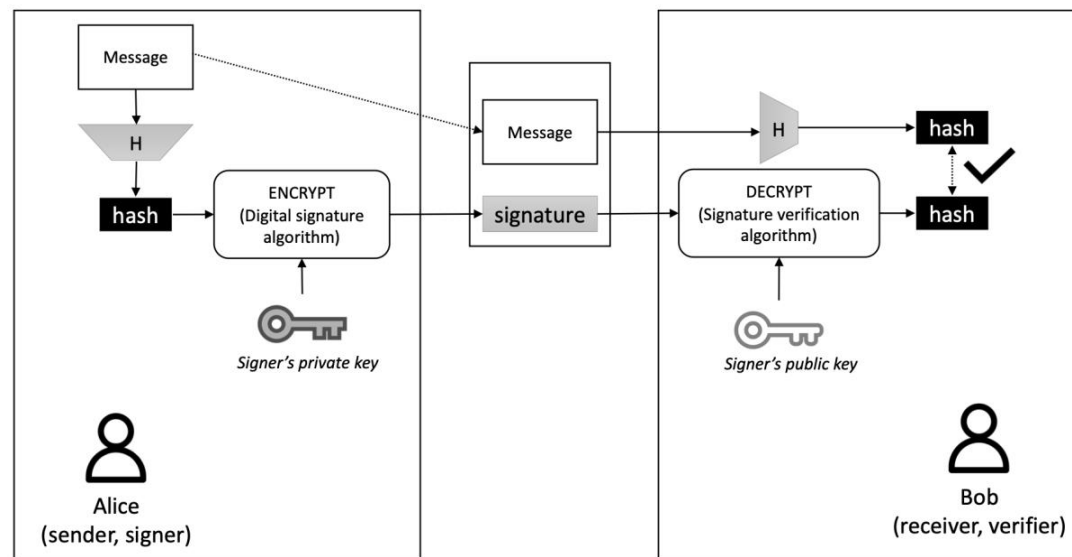
•Generally, message M (data) is not encrypted (but it could be)

-Sign-then-encrypt

-Encrypt-then-sign

•Signer Private key used to “sign”
(=encrypt the hash)

•Signer Public key used by any receiver to
verify the signature



(digital) signatures

Message is “signed” using the wax seal, so that the recipient can verify the identity of the sender

Message inside the envelope is not encrypted



What did we learn?

Asymmetric encryption uses different keys for encrypting/decrypting messages

RSA as an example of asymmetric encryption

Public key encryption and digital signatures as applications of asymmetric encryption

- Digital signature very important in blockchain-based systems

What's next

Thursday 9.8: Exercises

(begin in class and submit on bb once completed)

Next week: NO LECTURE (and no homework), yay!!