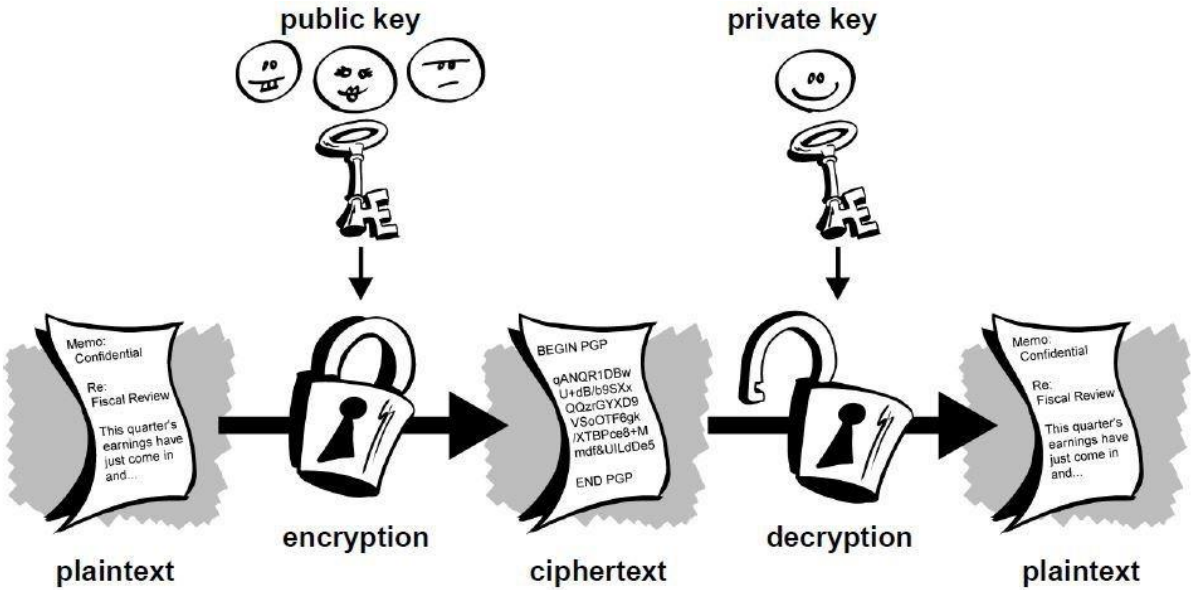
Introduction to Cryptography

By

Vipul Goyal



A Secure Cipher: One-Time Pad

Some Basics: XOR function

XOR Gate:

$0 \oplus 0 = 0$

$0 \oplus 1 = 1$

$1 \oplus 0 = 1$

$1 \oplus 1 = 0$

$$\begin{array}{r} 010110 \\ \oplus\ 110011 \\ \hline 100101 \end{array}$$

Key property: $S \oplus S = 0$

Key property: $S \oplus 0 = S$

Key property: Given $A \oplus B$, and A, easy to find B (and vice versa)

One-time pad

M = message K = key C = encrypted message

(everything in binary)

Key Generation: simply choose a random string

Encryption:

$$
\begin{array}{rl}
M = & 01011010111010100000111 \\
\oplus \quad K = & 11001100010101111000101 \\
\hline
C = & 10010110101111011000010
\end{array}
$$

$C = M \oplus K$ (bit-wise XOR)

One-time pad

M = message K = key C = encrypted message

(everything in binary)

Decryption:

$$\begin{array}{rl} C = & 100101101011111011000010 \\ \oplus \quad K = & 110011000101011111000101 \\ \hline M = & 010110101111010100000111 \end{array}$$

Encryption: $C = M \oplus K$

Decryption: $C \oplus K = (M \oplus K) \oplus K = M \oplus (K \oplus K) = M$

(because $K \oplus K = 0$)

One-time pad security

$$
\begin{aligned}
M &= 010110101111010100000111 \\
\oplus\ K &= 110011000101011111000101 \\
\hline
C &= 100101101011110110000010
\end{aligned}
$$

One-time pad is perfectly secure:

Say adversary sees ciphertext C = 0
Maybe M = 0 and K =0
Maybe M = 1 and K =1

No way to tell even if adversary has an infinitely powerful computer!

Say adversary sees ciphertext C = 1
Maybe M = 0 and K =1
Maybe M = 1 and K =0

One-time pad security

$$
\begin{aligned}
M &= 010110101111010100000111 \\
\oplus\ K &= 110011000101011111000101 \\
\hline
C &= 100101101011110110000010
\end{aligned}
$$

One-time pad is perfectly secure (another view):

For any M, if K is random, then C is also random

So adversary learns nothing about M by seeing C

One-time pad Limitation

$$
\begin{aligned}
M &= 01011010111010100000111 \\
\oplus\ K &= 11001100010101111000101 \\
\hline
C &= 10010110101111011000010
\end{aligned}
$$

The shared key has to be as long as the message!

Could we reuse the key?

One-time pad limitation Example

Alice sends a message to Bob everyday encrypted using the same key. Say messages are either ATTACK or DEFEND.

Day 1: Adv sees C_1 = ATTACK $\oplus$K

At the end of the day: Adv learns it was ATTACK,

Then: recover the key (by key property)

Day 2: if Alice uses the same key, recover message immediately

One-time pad limitation Example

A general attack:

Suppose you encrypt two messages M_1 and M_2 with K

$C_1 = M_1 \oplus K$

$C_2 = M_2 \oplus K$

Then $C_1 \oplus C_2 = M_1 \oplus M_2$ (by key property)

This is still non-trivial information about the two messages

Shannon's Theorem

Is it possible to have a secure system like one-time pad with key reuse or with key size less than the message size?

Shannon proved "no": given "sufficient time", can break any system like that

However in real world: nobody has infinitely time or infinite computation. Maybe only 10-20 years.

Later: will see good SKE encryption with key reuse based on "Hard Problems"

Hard Problems

- Almost all of Cryptography is based on “hard problems”
- Examples: assume factoring is hard, assume discrete log problem is hard
- These problems are not impossible to solve, but it just takes very long time for even a supercomputer to solve (e.g. 1 million years). No good algorithms known.
- However in future if someone finds a good way of factoring numbers, a lot of crypto (like RSA) will be broken
- Quantum computers (if built) can do factoring and discrete log efficiently

A Limitation of SKE

Alice and Bob should first meet in person to exchange the secret key K

What if you can't meet in person? What if you only have internet?

Example: suppose you want to establish a secure session with google.com or qq.com

(think https protocol used by your browser)

Public Key Encryption
(Only Definition)

Public Key Encryption

public

private

Can be used to encrypt.
But can't be used to decrypt.

Public key Encryption (PKE)

C

M

K_{pub}

K_{pri}

(M, K_{pub})

Enc

C

(C, K_{pri})

Dec

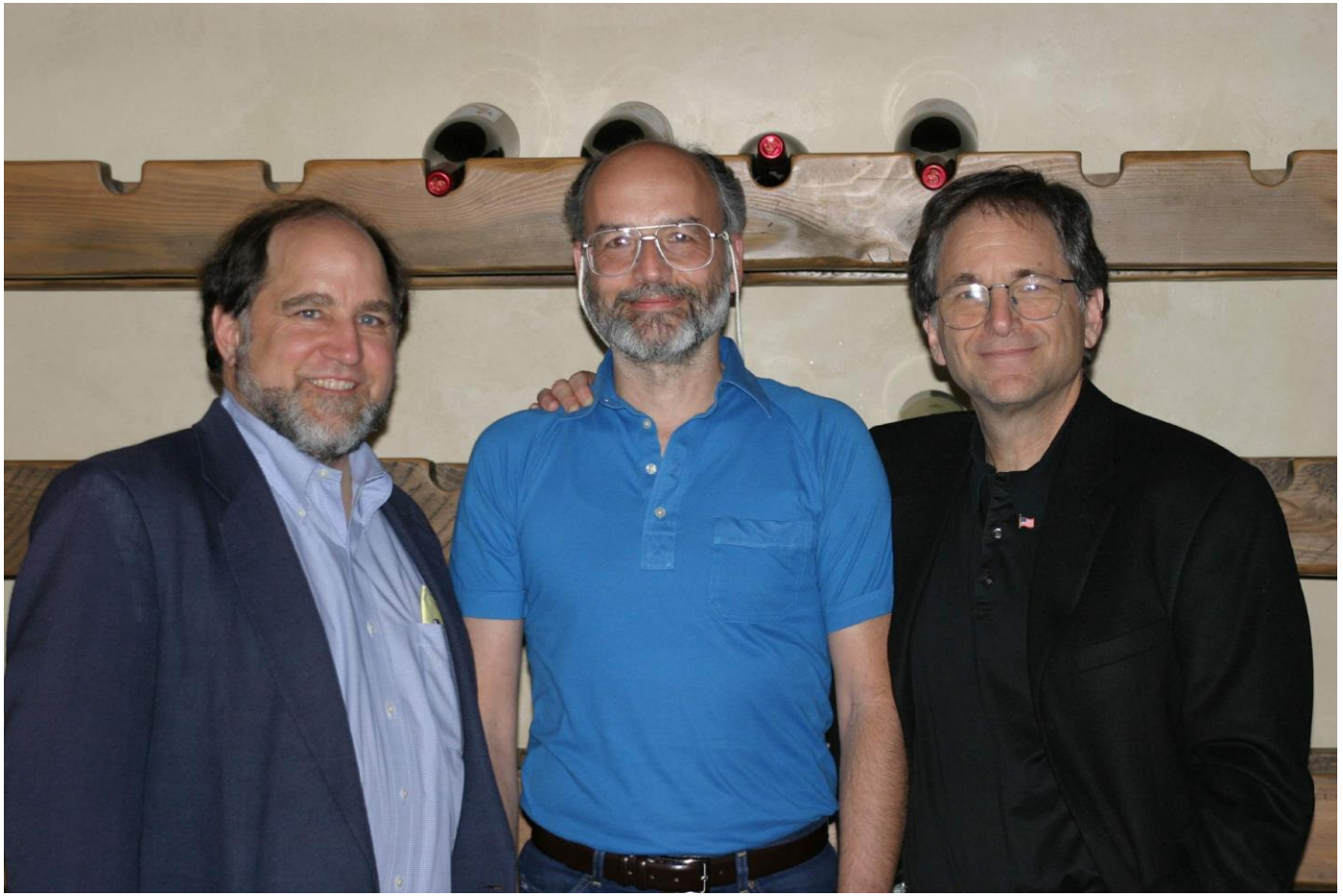
M

Defining PKE

1) Gen: takes no input. Outputs PK and SK (also denoted as K_{pub} and K_{priv})
2) Enc: Takes input *PK* and *M*. Outputs *C*.
3) Dec: Takes input *C* and SK. Outputs M.

RSA crypto system (1977)

2002 Turing Award



Ron Rivest Adi Shamir Leonard Adleman

An Interesting Question

Adversary intercepts the encrypted message. He wants to change it s.t.:

- If the original message was ATTACK, it becomes DEFEND
- If the original message was DEFEND, it becomes ATTACK

Is this possible?

- Encryption (such as one-time pad) only hides the message
- It doesn’t guarantee that adversary can’t change the message
- But if you can’t decrypt the message, how can you change it?

The Answer

Yes this attack is possible on one-time pad!!! (very counterintuitive). Say attacker knows that the message is either ATTACK or DEFEND. Given C, attacker changes it to C' as follows:

Let C be the ciphertext. What is C' = C $\oplus$ ATTACK $\oplus$ DEFEND?

Say C = K $\oplus$ ATTACK

C' = K $\oplus$ ATTACK $\oplus$ ATTACK $\oplus$ DEFEND
= K $\oplus$ DEFEND

Moral of the story: hiding doesn't imply non-tamperability!
(To prevent anyone from changing documents: we use signatures. So need "digital" signatures)

Digital Signatures
(Only Definition)

Digital Signatures

VK

M, σ

SK M

sign

σ (signature)

VK, M, σ

verify

0/1

Note: we don't care about hiding M

Defining Digital Signatures

1) Gen: An algorithm which outputs VK and SK
2) Sign: An algorithm which takes as input *SK* and *M*.

   Outputs σ.
3) Verify: Takes input (M, σ, VK). Outputs 0/1.

Security

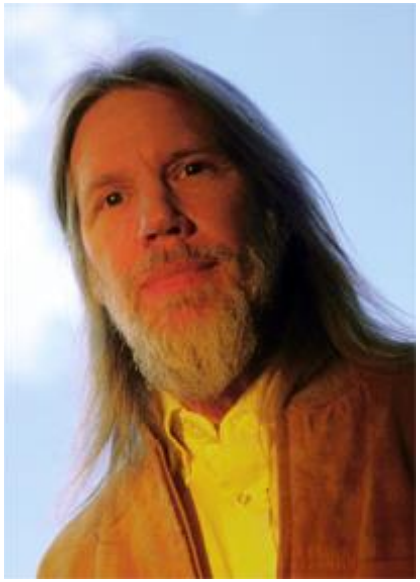
Adv is given:

1) Verification key VK
2) Signatures ($\sigma_1, \sigma_2, \ldots, \sigma_q$) on messages ($M_1, M_2, \ldots, M_q$) chosen by him

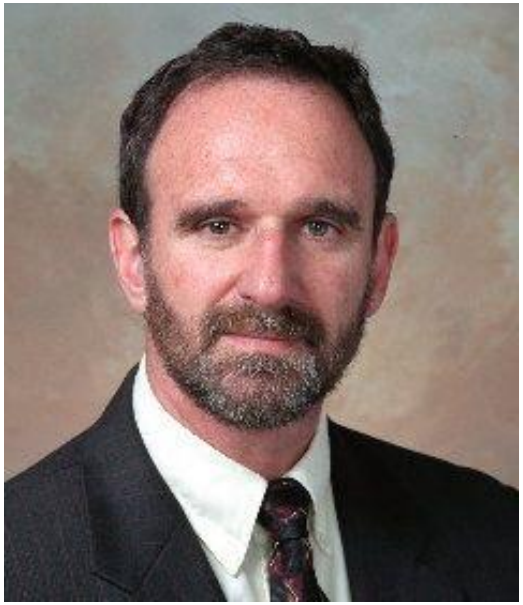
Adv still can't output a valid signature on a new message

(That is, can't output (σ, M) s.t. Verify(M, σ, VK) = 1 and M is different from all M_i)

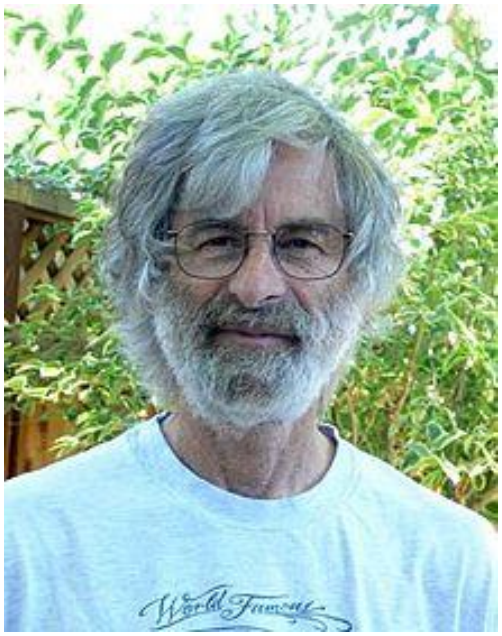
Digital Signatures (1976)



Whitfield Diffie



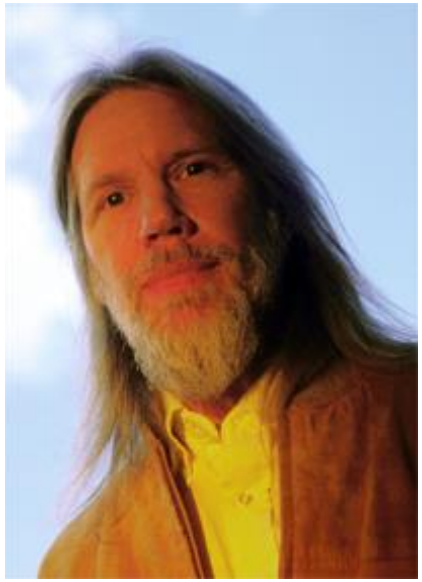
Martin Hellman

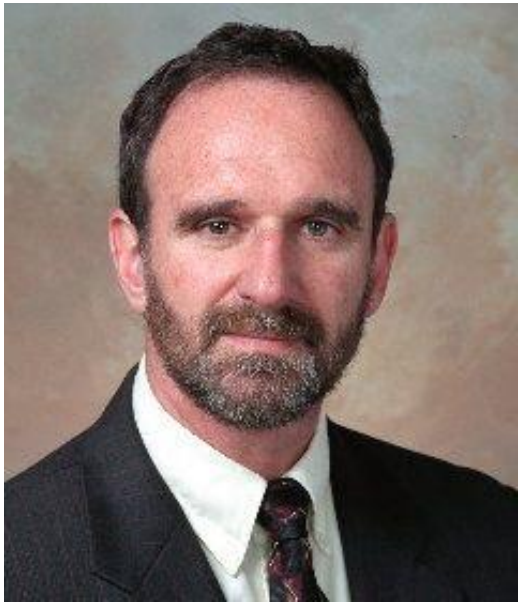


Leslie Lamport

Interesting Story

A Homework Assignment





Whitfield Diffie Martin Hellman

Read: New Directions in Cryptography (1976)
Link: https://ee.stanford.edu/~hellman/publications/24.pdf
Introduced: PKE, digital signatures, key exchange…

(Turing Award)

Questions and Discussion?