Singapore Management University 2 September 2005

Is a Mathematical Theory of Cryptography Possible?

James L. Massey

Prof.-em. ETH Zürich, Adjunct Prof., Lund Univ.,
Adjunct Prof., Tech. Univ. of Denmark

JLM Consulting
Trondhjemsgade 3, 2TH
DK-2100 Copenhagen East

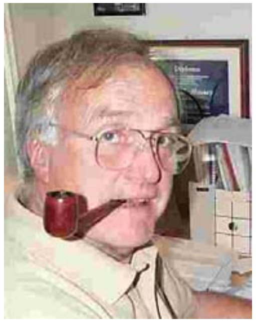
JamesMassey@compuserve.com

Cryptology

("hidden word")

Cryptography

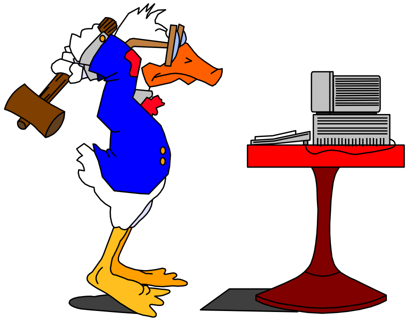
(code making)



The "**good guys**"

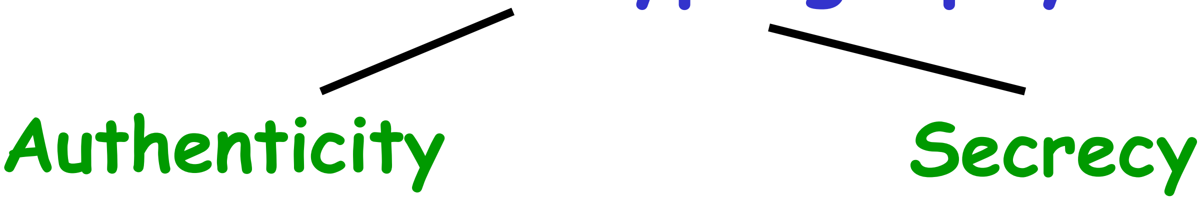
Cryptanalysis

(code breaking)



The "**bad guys**"

Goals of cryptography

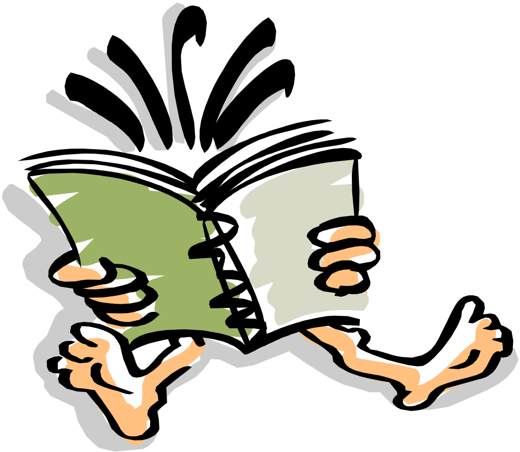


Authenticity

Secrecy

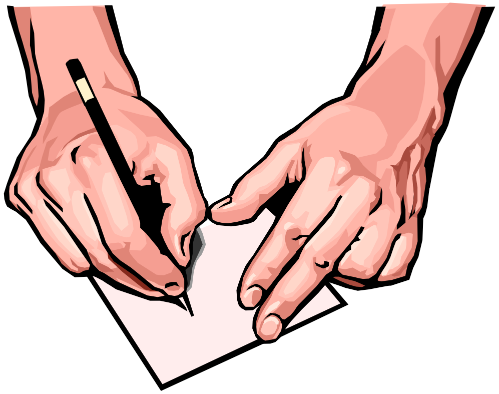
Xuejia Lai has given a useful **razor** for deciding whether something is a matter of secrecy or a matter of authenticity.

Secrecy - concerned with who has **access** to (or can **read**) a legitimate message.



Secrecy deals with **safeguarding the future** by ensuring that **only authorized recipients will be able to gain access** to (or read) a legitimate message.

Authenticity - concerned with who can **create** (or **write**) a legitimate message.



Authenticity deals with **protecting the past** by

- ensuring that the creator (or author) was **entitled to create** (or write) the message
- ensuring that the **contents** of the message **have not been altered**

“As a first step in the mathematical analysis of cryptography, it is necessary to idealize the situation suitably, and to define in a mathematically acceptable way what we shall mean by a secrecy system.”

Shannon, 1949.

How did Shannon define a secrecy system in a

“**mathematically acceptable way**”?

He drew a picture!

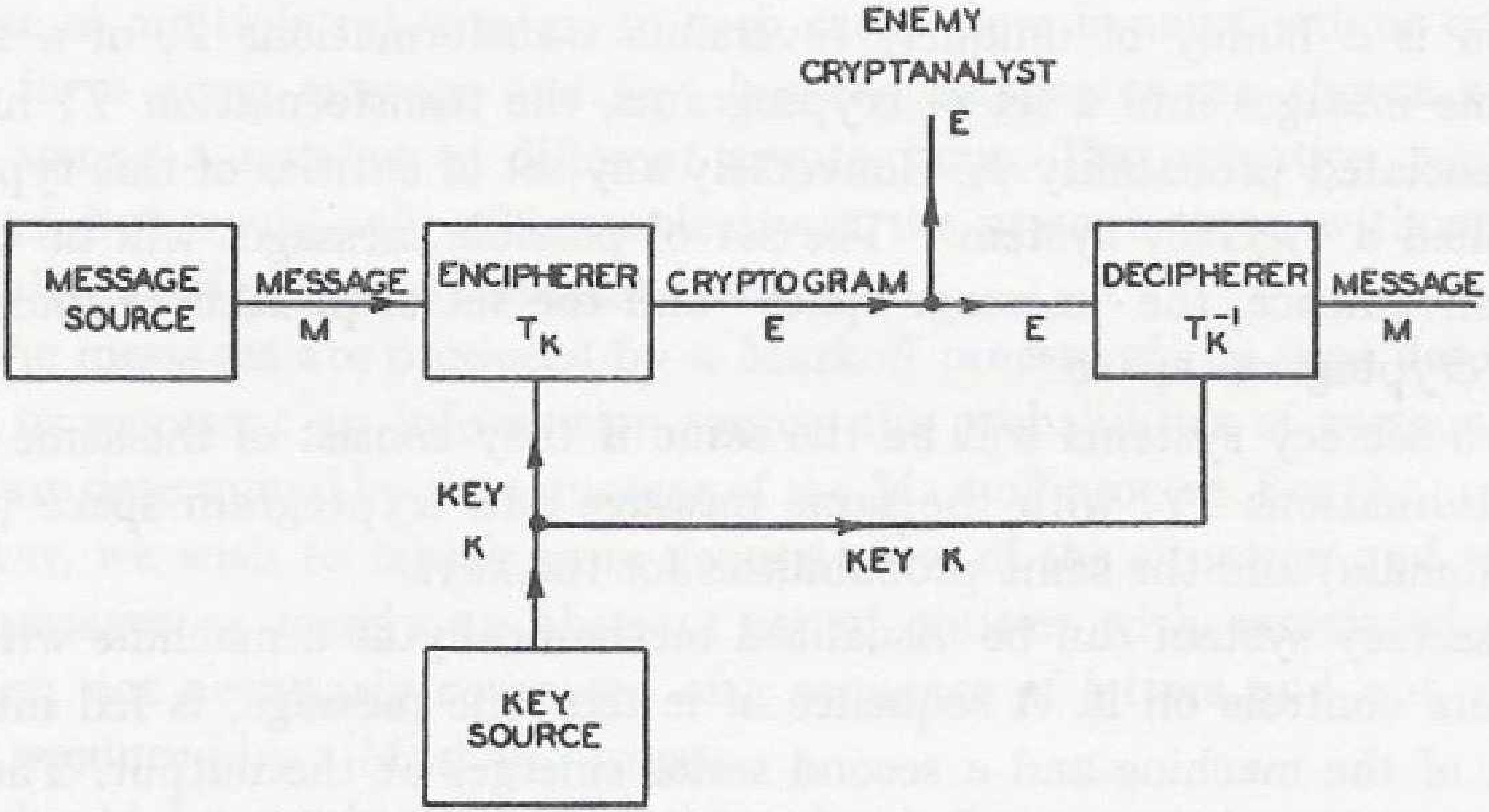
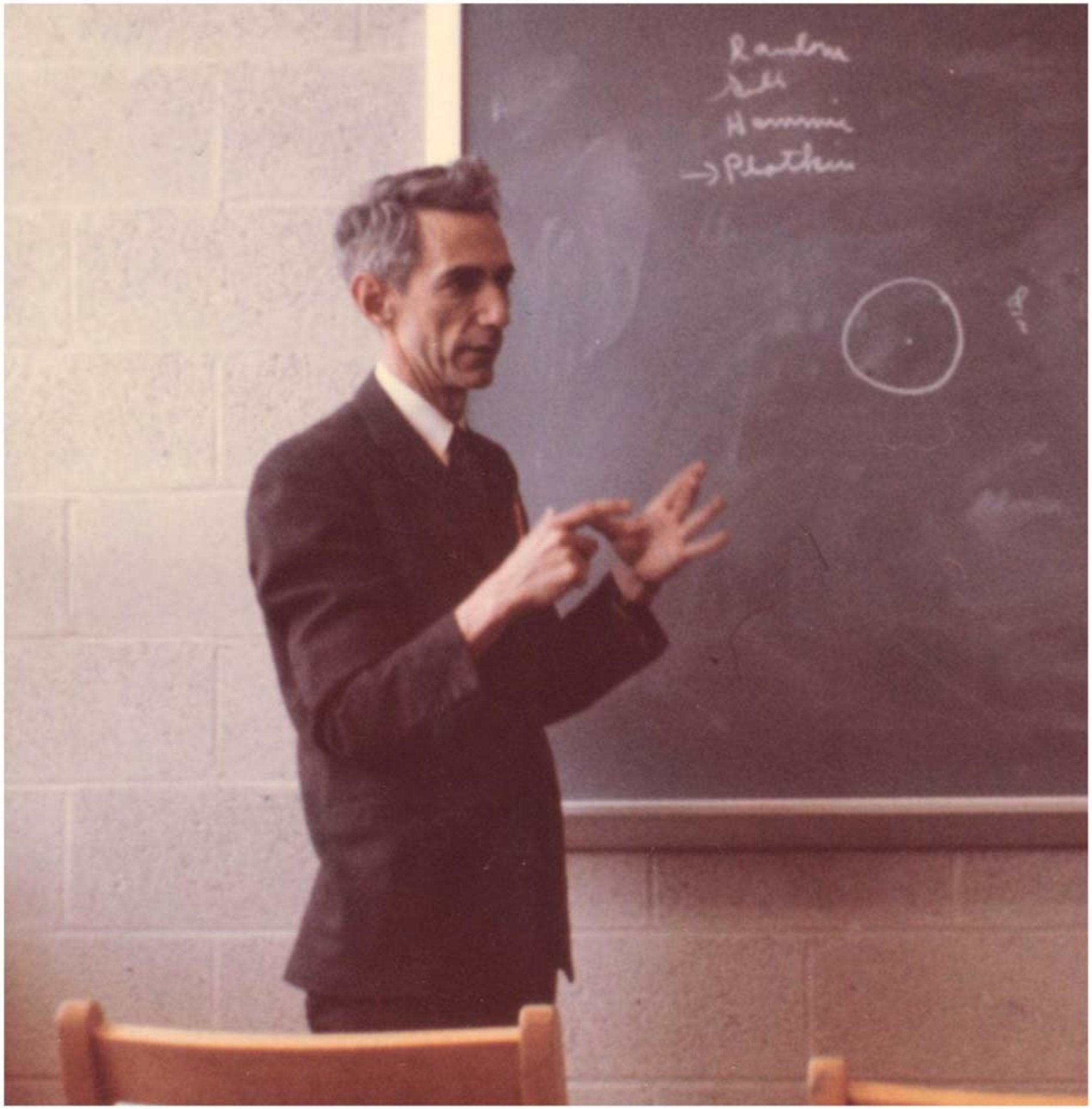


Fig. 1—Schematic of a general secrecy system.

(Shannon, 1949)



Claude Elwood Shannon - 17 April 1961
(photograph by Göran Einarsson)

In Fig. 1, Shannon is stating the following **assumptions**:

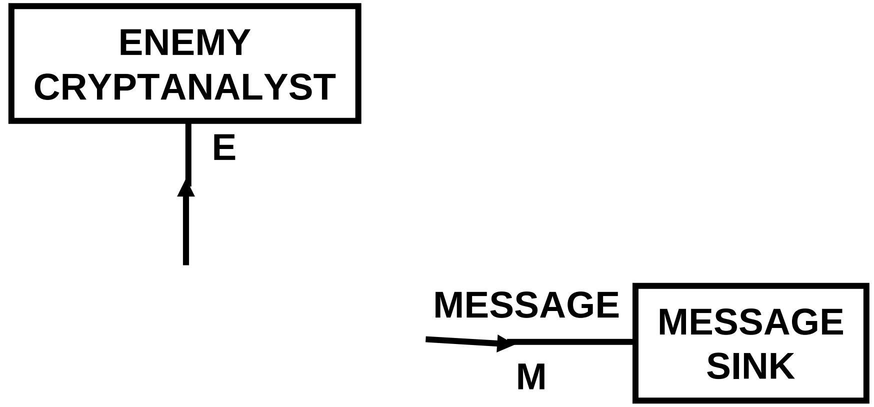
- The secret key is somehow delivered to the sender and receiver in such a way that the **enemy cryptanalyst has no access to the secret key** **K**.
- The **enemy cryptanalyst has access to the cryptogram** **E** **but nothing else**.
- The location of the key source is not specified, but the **key source and message source are independent**.
- The **cryptogram** **E** **and the key** **K** **uniquely determine the (plaintext) message** **M**.

In the text of his paper, Shannon also states that he will "***assume that the enemy knows the system being used***" (Kerkhoffs' principle).

Shannon takes it as an **axiom** that **sources may be described as** devices whose outputs are **random processes**.

Shannon's Fig. 1 should really have been entitled "**Schematic of a general one-key point-to-point secrecy system with no feedback and with no public randomizer under a ciphertext-only attack**".

It would also have been better had Shannon added the following details to his picture



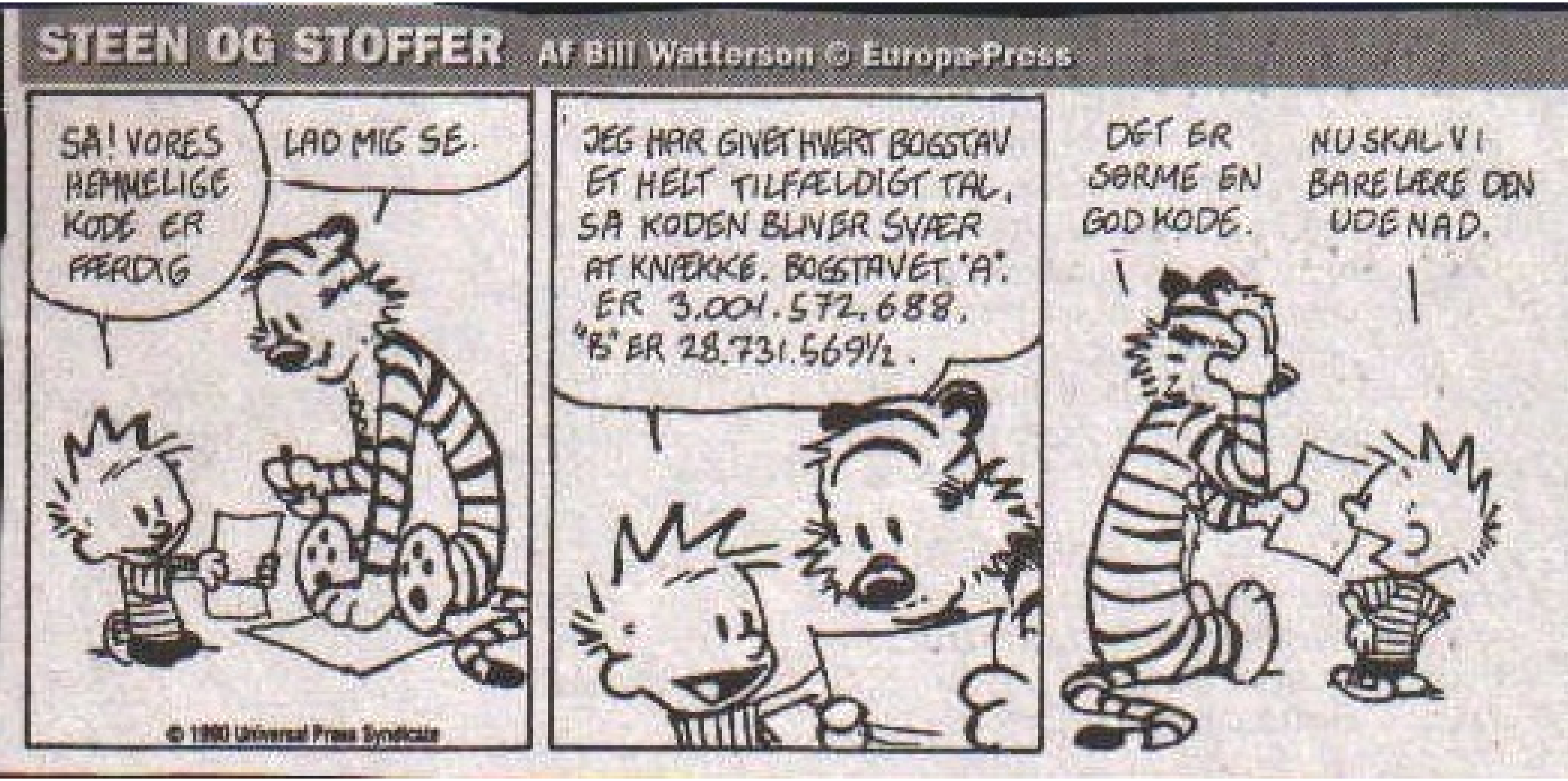
to make it clear that the enemy cryptanalyst sees only **E** and that the decrypted message **M** goes nowhere else.

Security

applies to all types of protection that we may be interested in, e.g., **secrecy** or **authenticity**.

What does security really mean?

Security as it is usually considered today:


STEEN OG STOFFER Af Bill Watterson © Europa-Press
SÅ! VORES HEMMELIGE KODE ER FÆRDIG
LAD MIG SE.
© 1990 Universal Press Syndicate
JEG HAR GIVET HVERT BOGSTAV ET HELT TILFÆLDIGT TAL, SÅ KODEN BLIVER SVÆR AT KNÆKKE. BOGSTAVET "A" ER 3.004.572.688. "B" ER 28.731.569½.
DET ER SØRME EN GOD KODE.
NU SKAL VI BARE LÆRE DEN UDENAD.

Again Shannon put us on a “mathematically acceptable” path by distinguishing between **two types of security**:

- **Unconditional*** (or **theoretical** as Shannon called it) - which means the security against an enemy who has **unlimited time and computational resources**.
- **Computational** (or **practical** as Shannon called it) - which means the security against an enemy who has a **specified limited amount of time and computational resources**.

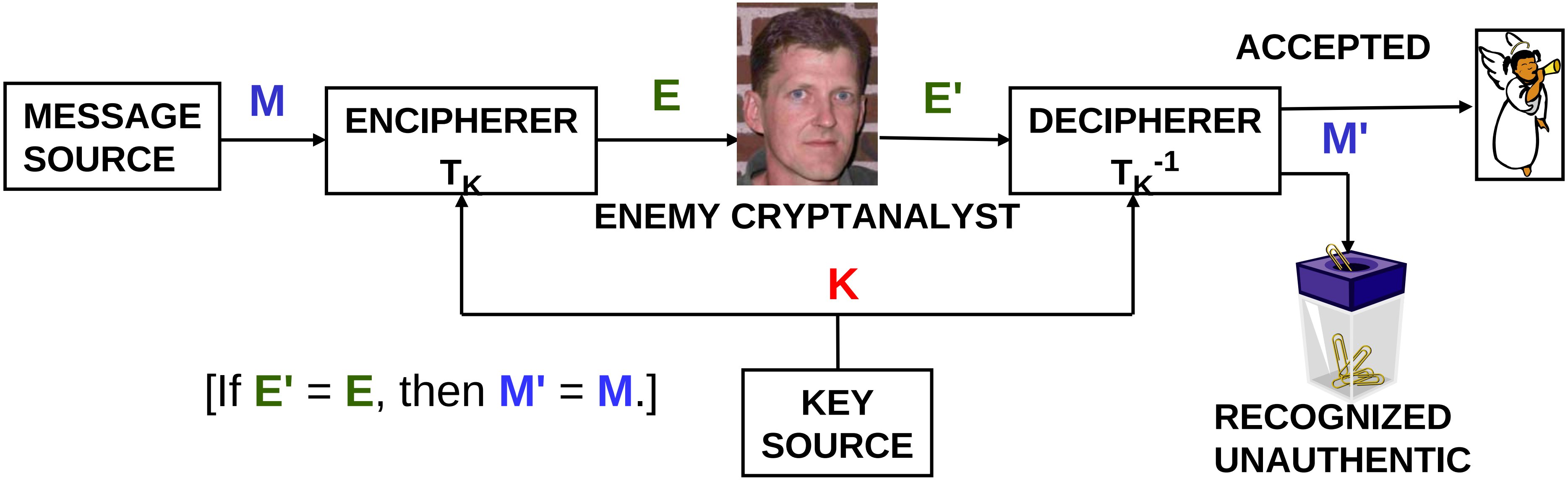
*Today, unconditional security is often called **information-theoretical security**.

Shannon's **mathematical model of a secrecy system** was given in **1949**.

It was not until **1984** that Simmons gave a corresponding **mathematical model for an authenticity system**.

One reason for this lag was that it took a long time before we understood that **secrecy systems and authenticity systems are two very different things**.

[E' can be the legitimate cryptogram E or a phony cryptogram E' (E' ≠ E) created by the enemy.]



MESSAGE SOURCE

M

ENCIPHERER T_K

E

ENEMY CRYPTANALYST

E'

DECIPHERER T_K^{-1}

ACCEPTED

M'

RECOGNIZED UNAUTHENTIC

K

KEY SOURCE

[If E' = E, then M' = M.]

[E' is ACCEPTED if and only if it is a valid cryptogram for the key K.]

Simmons' 1984 Model of a **Substitution Attack** on an **Authenticity System**

(but Simmons did not himself draw such a picture!)

The **necessary ingredients for a mathematical theory** of cryptography are:

- a carefully defined **mathematical model** of the cryptographic system to be considered with specification of the enemy's knowledge.
- a **precise definition** of what **security** means.
- statements of the **axioms** (the "self-evident truths" not provable within the model).

Shannon provided the above for a secrecy system under a ciphertext-only attack. For instance, he defined unconditional security (which he called *perfect secrecy*) to mean that the *message and the cryptogram are independent*. He was then in position to **prove theorems** about security.

Is a Mathematical Theory of Cryptography Possible?

Yes, at least if one restricts the theory to **unconditional security**. Shannon and Simmons initiated such a theory, which is still under development.

Shannon's "mathematical theory of communication" (now usually called "information theory") is **essentially parallel** to his theory of secrecy for **unconditional security**. In fact, he entitled his 1949 paper

"**The communication theory of secrecy systems**".

What did Shannon have to say about **computational security** of a secrecy system?

"**The problem of good cipher design is essentially one of finding difficult problems**, subject to certain other conditions. **This is a rather unusual situation, since one is ordinarily seeking the simple and easily soluble problems in a field**."

. . .

"How can we ever be sure that a system which is not ideal and therefore has a unique solution for sufficiently large *N* will require **a large amount of work to break with every method of analysis**? ... **We may construct our cipher in such a way that breaking it is equivalent to (or requires at some point in the process) the solution of some problem known to be laborious**."

(Shannon, 1949)

A sobering thought:
Shannon was unable to prove anything interesting about unconditional security.

Something I heard recently:
"If it says it's provably secure, it's probably not."

Lars R. Knudsen

How did we get into such a situation and is it possible to get out of it?

We have not demanded that cryptographic research be conducted with mathematical rigor and objectivity.

There are far too many things taken for granted without proof in cryptographic research today.

What is a “hard problem”?

The most commonly used definition today in cryptography (but not many cryptographers would admit this) of a “hard problem” is:

“This is a hard problem—no one (that we know of) has been able to solve this problem.”

Another often used definition is:

“A hard problem is one that is at least as difficult to solve as some problem many people believe to be a hard problem.”

My favorite definition:

“A hard problem is one that nobody works on.”

If we want to work mathematically, then to determine the difficulty of a problem we need

- a **well-defined computational model** (e.g., a 10 GHz 64-bit processor with a vast and expandable memory, or a quantum computer with specified power). This specifies the "limited computational resources" available to the enemy cryptanalyst.
- a **careful formulation of the problem** that pins down all details in a precise manner.

If we are to use the results **to claim computational security** then we also need

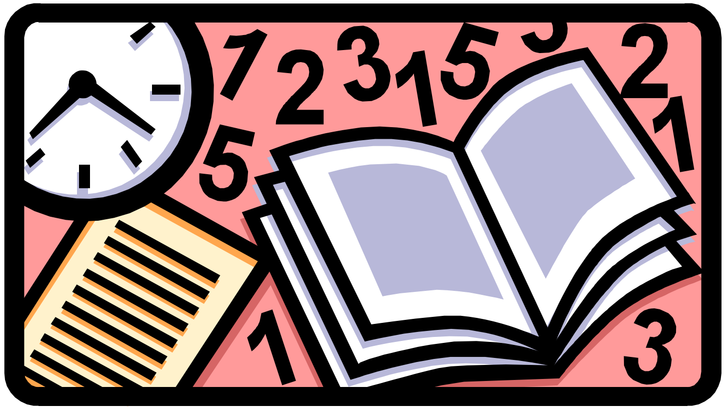
- a **proof** that breaking the cryptographic system **"is equivalent to (or requires at some point in the process) the solution of"** this problem.

Where can we look for help in determining the difficulty of a problem as part of a mathematical theory of cryptography for computational security?

(The "usual suspects":)

- **Number theory**?
- Theoretical computer science, i.e., **computational-complexity theory**?
- **Gate complexity theory**?

Number theory?



Forget it!
Nobody has ever proved an interesting **lower bound** on the complexity of anything in number theory. Upper bounds are easy to come by, but are essentially uninteresting.

Theoretical computer science?

"A **rose** is a **rose** is a **rose**."
Gertrude Stein

But in **computational complexity theory**,

"a ***problem*** is not a **problem** is not a ***problem***"
and
"a ***function*** is not a **function** is not a ***function***".

Problems (or ***functions***) must have countably infinitely many instances of increasing size, each of which is a **problem** (or a **function**).
[In simpler words, a ***problem*** (or ***function***) is a countably infinite family of **problems** (or **functions**)] whose sizes grow without bound.]

Example (Jevon's **problem**, 1873):
m = 8 616 460 799 is the product of two distinct primes, what are they?

Jevon stated that "I think it is unlikely that anyone will ever know; for they are two large prime numbers."

N. B.: 8 616 460 799 = 96 079 * 89 681

Example: The ***problem:*** Given the product m of two distinct primes p_1 and p_2, find these primes.

(Jevon's **problem** is an **instance** of the above ***problem***.)

Breaking a cryptographic system (say, the AES) is a **problem**, not a ***problem***.

My opinion: **Computational complexity theory is of little or no use** in determining the computational security of cryptographic systems.

Would you be happy if you designed a cipher for which breaking the cipher required the attacker to solve one instance of an **NP-hard** ***problem*** ?????

All bets based on computational complexity theory are off if **quantum computers** become a reality.

Gate complexity of functions?

(N.B. Shannon liked to work with gate complexity!)

$\mathbf{P_n}$ = set of permutations on $\mathbf{\{0, 1\}^n}$ (i.e., the set of **invertible** **functions** from n bits to **n** bits).

The **gate complexity** of a **function** in $\mathbf{P_n}$ is the smallest number of gates (a gate is defined as a boolean function of two variables) in an acyclic gate network that computes this **function**.

Can we find good **"one-way" functions** in $\mathbf{P_n}$, i.e., invertible functions with **great computational asymmetry**? This would seem to be the first step on the way to a theory of cryptography for computational security.

Alain Hiltgen holds the current world record for **computational asymmetry** for constructive **functions** in $\mathbf{P_n}$. He can, for every **n**, construct a function whose inverse requires **twice as many gates** as the function itself!

Is a Mathematical Theory of Cryptography for computational security Possible?

No, not with the methods of number theory and theoretical computer science.

Maybe with the methods of gate complexity.

Another sobering thought:

We do not even know whether genuine one-way functions, as measured by gate complexity, exist.

In 1996 I was able to prove (with an assist from Eli Biham) the following:

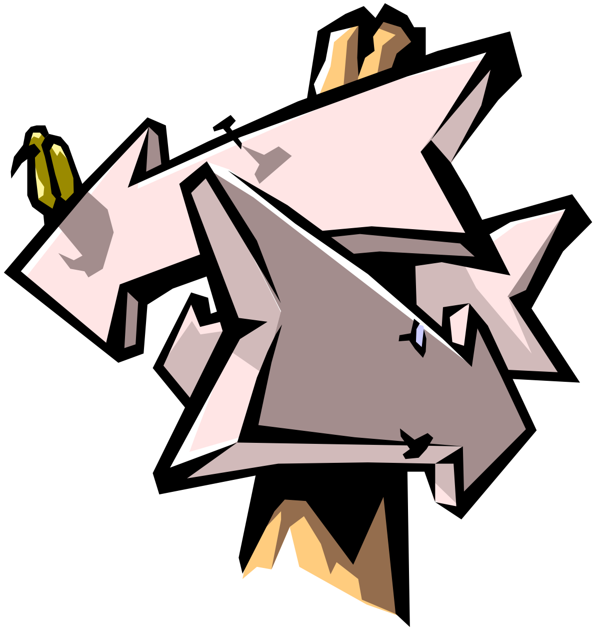
Proposition: For all $\mathbf{n \geq 6}$, **virtually all functions** in $\mathbf{P_n}$ have gate complexity **that differs by a factor of less than 2.5** from the gate complexity of their **inverse function**.

This is in stark contrast to Shannon's channel coding theorem in which he showed that **virtually all codes are good**, even if they might be hard to find.

If **one-way functions** turn out to **exist**, the important **practical task** will be to find one.

You can find a proof of the previous proposition as well as more information on gate complexity by going to **http://www.iacr.org** and following the links to the IACR Distinguished Lectures and then to my 1996 lecture.

Are there any other approaches that might lead us to a mathematical theory of cryptography for computational security?

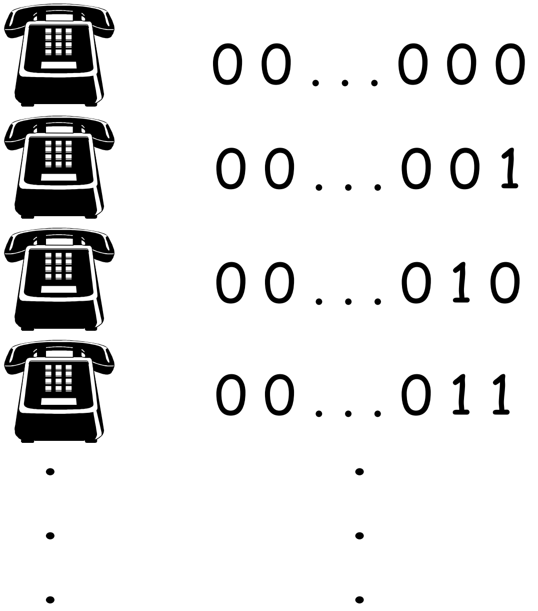


Probabilistic methods?

Maybe! These techniques allow one to **combine information-theoretic reasoning with measures of difficulty**.

We give an example to demonstrate this.

Diffie's (unpublished) scheme with a public randomizer:



0 0 . . . 0 0 0

0 0 . . . 0 0 1

0 0 . . . 0 1 0

0 0 . . . 0 1 1

.
.
.

When one dials the **L-bit telephone number** of one of these 2^L telephones, one receives a recorded message consisting of an **N**-bit string (**N** » **L**) that was a produced by a BSS for this telephone only.

- Alice and Bob agree privately on an L-bit secret key **K**.
- Later, when Alice wants to send Bob a confidential message of length **N** bits, she dials telephone **K**, gets its sequence, and does a **"one-time pad encryption"** of the message.
- When Bob gets the cryptogram from Alice, he dials telephone **K**, gets its sequence, and does a "one-time pad decryption" of the message.

To attack Diffie's scheme in a ciphertext-only attack, the enemy cryptanalyst must dial one telephone after another and check if its sequence decrypts Alice's cryptogram to a meaningful message (we assume that Alice did not remove the redundancy from her plaintext before encryption). Suppose the attacker checks a total of **T** telephones. The probability ofsuccess is

$$P(\text{attack success}) = \mathbf{T}/2^{\mathbf{L}}.$$

Example: **L** = 40 bits and **N** = 10^6. **T** = 10^9.

$$P(\text{attack success}) = \mathbf{T}/2^{\mathbf{L}} \approx 10^{-3}.$$

Diffie's scheme is an example of what Maurer at Eurocrypt '90 called "**conditionally perfect secrecy**", i.e., it gives **perfect secrecy conditioned on the occurrence of some event**, here the event that the attacker does not dial the phone dialed to encrypt.

Diffie's unpublished scheme predates Maurer's scheme, but Maurer greatly reduced the amount of public randomizer required. We chose Diffie's scheme for our example because of its conceptual simplicity and the ease with which one can calculate the attacker's **probability of success for a given amount of work**.

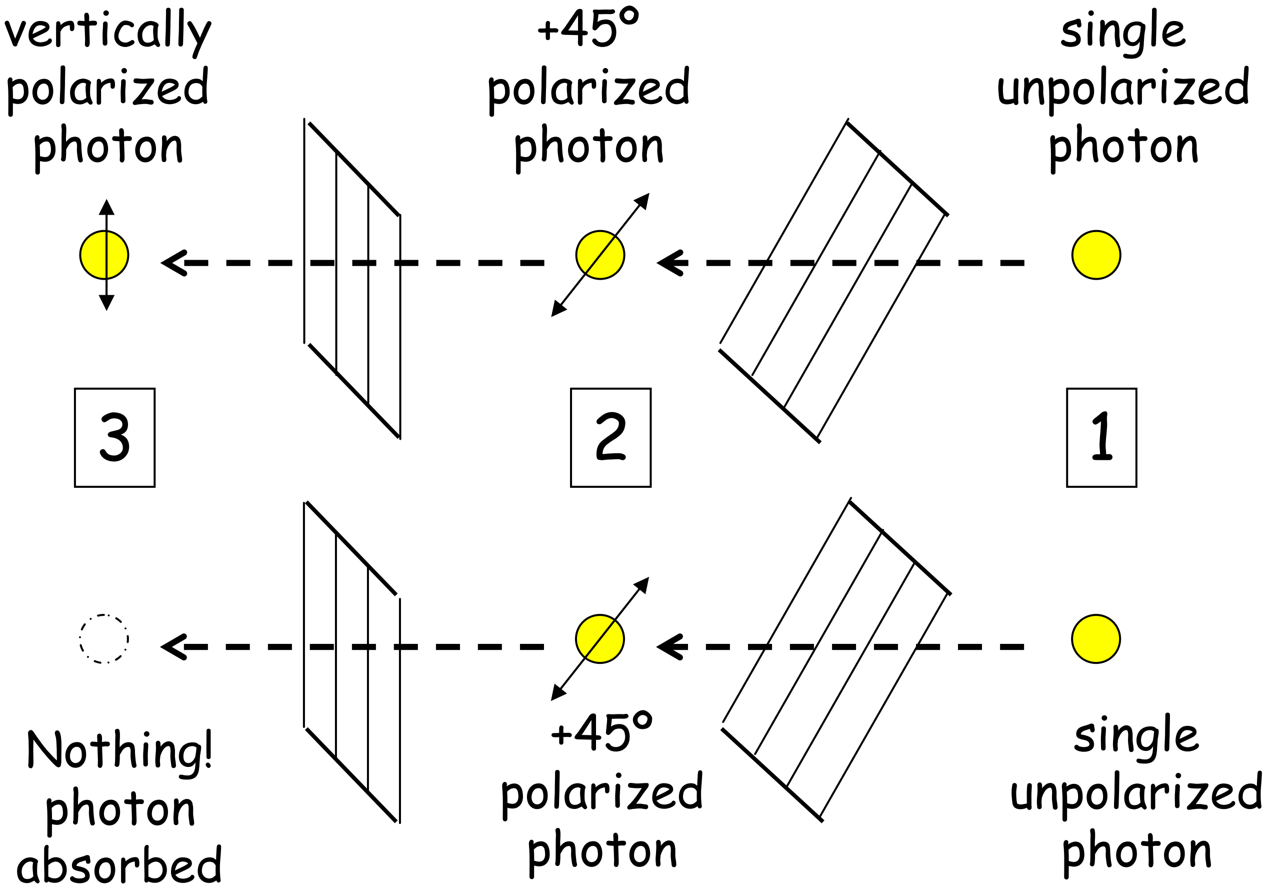
Isomorph s.r.l., a spin-off company from Udine University, now markets a cipher much like that proposed by Maurer at Eurocrypt '90. See http://www.isomorph.it/crypto_application.htm for their entertaining animations of such probabilistic ciphers.

Is a Mathematical Theory of Cryptography for computational security possible using probabilistic techniques?

Maybe, but there has been little fundamental progress since the work of Maurer.

Quantum Physics?

The two equally likely outcomes of an experiment:


vertically polarized photon
+45° polarized photon
single unpolarized photon
3
2
1
Nothing! photon absorbed
+45° polarized photon
single unpolarized photon

Quantum Cryptography

is more accurately called

Quantum Key Distribution

and even more accurately called

Quantum Key Agreement

Quantum Key Agreement refers to schemes in which **two parties reach agreement on a random key (chosen by nature)** in such a way that an eavesdropper will obtain no information about this key and, moreover, the presence of the **eavesdropper** will be detected if the eavesdropping is done for an extended period.

The basic idea: Alice transmits a random sequence by sending a +45° polarized photon to represent a 1 and a vertically polarized photon to represent a 0. Bob randomly chooses between a horizontal polarizer and a -45° polarizer to detect each photon he receives.

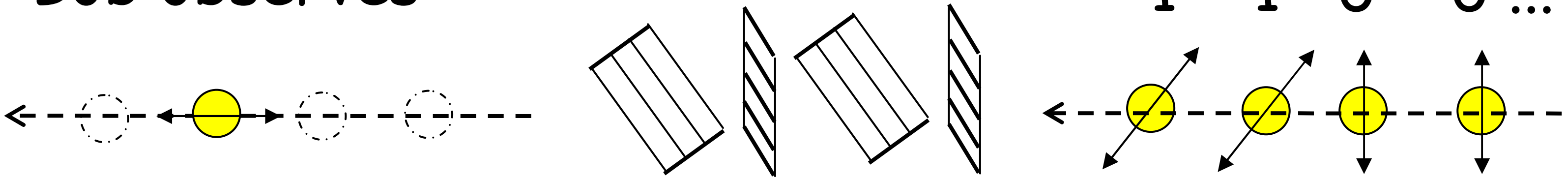
Example:

Bob observes

Bob polarizes

Alice sends

1 1 0 0 ...



Bob decides

Δ 1 Δ Δ ...

(where Δ indicates an “erasure”)

- Bob’s decisions (non-erasures) will never be wrong
- Eavesdropper will cause errors with probability 1/4.

Alice and Bob need a protocol, which includes the use of error-correcting codes, to reach agreement on a key of a specified size in such a way that the eavesdropper is kept in the dark with high probability, unless the eavesdropper “listens” to a substantial fraction of the photons in which case the eavesdropper will be detected through the error probability that Alice and Bob observe during the performance of their protocol.

The eavesdropper can successfully **deny service** to Alice and Bob by listening to all transmissions.

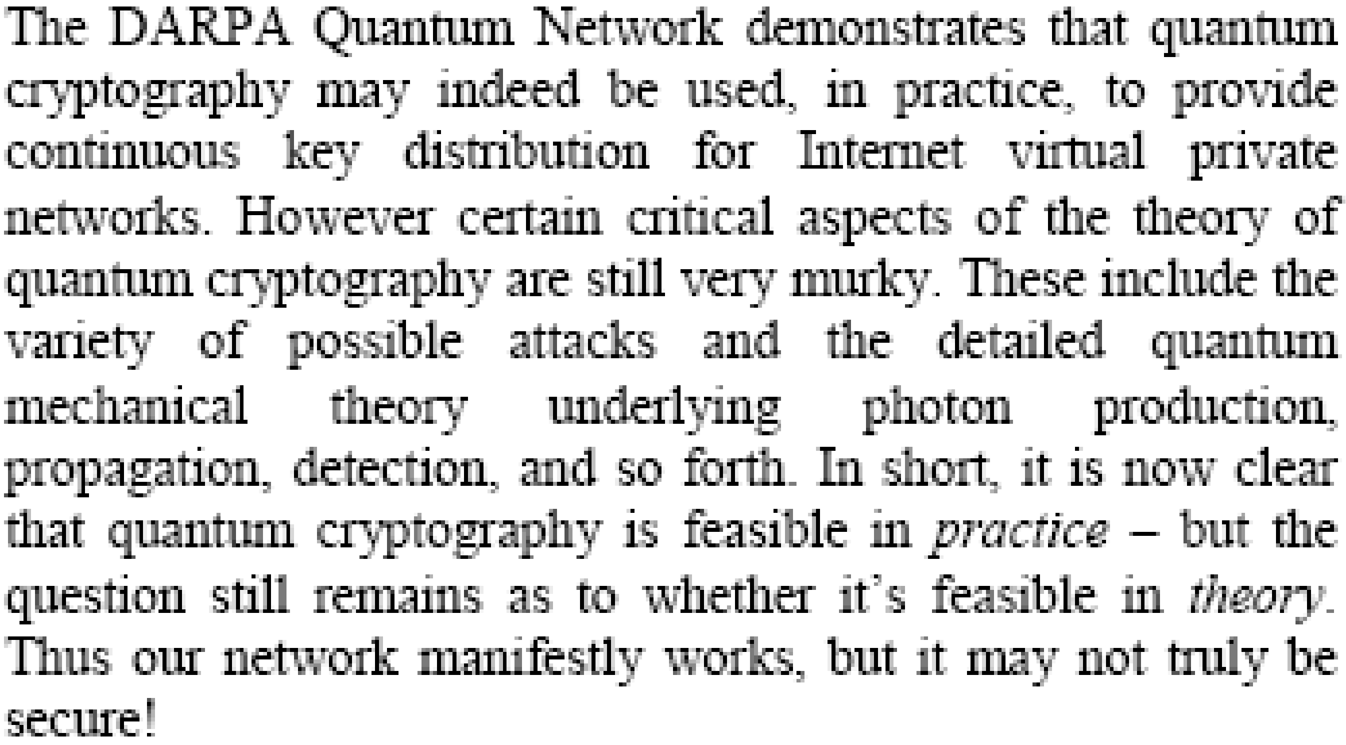
Quantum Key Agreement may be able to give a **provably secure** method to create secret keys

- **provided that the strange rules of quantum physics truly hold in the real world**,
- **provided that one can implement a single-photon transmission channel** between the encrypter and the decrypter,
- **provided that the adversary is only an eavesdropper who doesn't listen all the time**,
- and **provided that** the protocol for reaching key agreement by public discussion has no flaws.

(After the key has been established, Alice and Bob will still need some other kind of secret-key cryptographic system for the exchange of their messages.)

The whole point of science--and the feature to which it owes all its success--is that **no idea is to be believed until it has been rigorously tested by experiment**. Thus it is worrying that these days many theorists spend their whole careers on ideas that are untested experimentally.

L. Smolin, *American Scientist*, vol. 93, p. 453, 2005.



The DARPA Quantum Network demonstrates that quantum cryptography may indeed be used, in practice, to provide continuous key distribution for Internet virtual private networks. However certain critical aspects of the theory of quantum cryptography are still very murky. These include the variety of possible attacks and the detailed quantum mechanical theory underlying photon production, propagation, detection, and so forth. In short, it is now clear that quantum cryptography is feasible in *practice* – but the question still remains as to whether it's feasible in *theory*. Thus our network manifestly works, but it may not truly be secure!

C. Elliott, D. Pearson, G. Troxel, SIGCOMM 2003

Essentially there has been **zero progress** toward a **mathematical theory of cryptography for computational security**, whether in secret-key or in public-key cryptography!

Today almost no one works on the **mathematical** problem of developing provably computationally-secure systems!

Today almost everyone plies the **art** of cryptography (like Steen & Stoffer), generating more and more schemes that nobody can prove are computationally secure.

If cryptography **tomorrow** is going be fundamentally different from cryptography today, we must get to work on the mathematical theory of cryptography.

Developing a mathematical theory of cryptography for computational security is a wide-open area of research, success in which could have enormous practical consequences.

It might not be easy!

"Problems worthy of attack,
prove their worth by hitting back!"
Piet Hein

"A hard problem is one that nobody works on."