

The growth of cryptography

Ronald L. Rivest

Viterbi Professor of EECS
MIT, Cambridge, MA

James R. Killian Jr. Faculty Achievement Award Lecture
February 8, 2011

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

Euclid – 300 B.C.



There are infinitely many primes:
2, 3, 5, 7, 11, 13, ...

Euclid – 300 B.C.



There are infinitely many primes:
2, 3, 5, 7, 11, 13, ...

The greatest common divisor of two
numbers is easily computed
(using “Euclid’s Algorithm”):
 $\text{gcd}(12, 30) = 6$

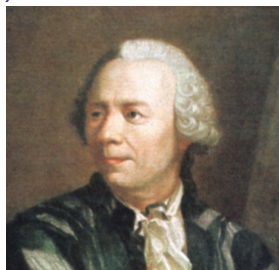
Greek Cryptography – The Scytale



An unknown *period* (the circumference of the scytale) is the secret key, shared by sender and receiver.

Pierre de Fermat (1601-1665)

Leonhard Euler (1707–1783)



Fermat's Little Theorem (1640):

For any prime p and any a , $1 \leq a < p$:

$$a^{p-1} = 1 \pmod{p}$$

Pierre de Fermat (1601-1665)

Leonhard Euler (1707–1783)



Fermat's Little Theorem (1640):

For any prime p and any a , $1 \leq a < p$:

$$a^{p-1} = 1 \pmod{p}$$

Euler's Theorem (1736):

If $\gcd(a, n) = 1$, then

$$a^{\phi(n)} = 1 \pmod{n},$$

where $\phi(n) = \#$ of $x < n$ such that $\gcd(x, n) = 1$.

Carl Friedrich Gauss (1777-1855)



Published *Disquisitiones Arithmeticae* at age 21

Carl Friedrich Gauss (1777-1855)



Published *Disquisitiones Arithmeticae* at age 21

“The problem of *distinguishing prime numbers from composite numbers and of resolving the latter into their prime factors* is known to be one of the most important and useful in arithmetic. . . . the dignity of the science itself seems to require solution of a problem so elegant and so celebrated.”

William Stanley Jevons (1835–1882)



Published *The Principles of Science* (1874)

William Stanley Jevons (1835–1882)



Published *The Principles of Science* (1874)

Gave world's first *factoring challenge*:

“What two numbers multiplied together will produce 8616460799 ? I think it unlikely that anyone but myself will ever know.”

William Stanley Jevons (1835–1882)



Published *The Principles of Science* (1874)

Gave world's first *factoring challenge*:

“What two numbers multiplied together will produce 8616460799 ? I think it unlikely that anyone but myself will ever know.”

Factored by Derrick Lehmer in 1903. (89681 * 96079)

World War I – Radio

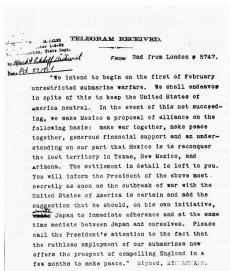
- ▶ A marvelous new communication technology—*radio* (Marconi, 1895)—enabled instantaneous communication with remote ships and forces, but also gave all transmitted messages to the enemy.

World War I – Radio

- ▶ A marvelous new communication technology—*radio* (Marconi, 1895)—enabled instantaneous communication with remote ships and forces, but also gave all transmitted messages to the enemy.
- ▶ Use of cryptography soars.

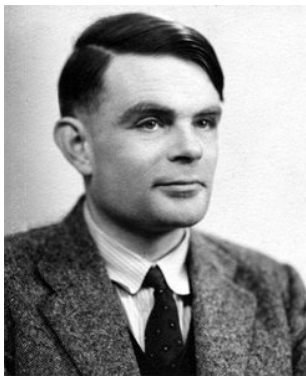
World War I – Radio

- ▶ A marvelous new communication technology—*radio* (Marconi, 1895)—enabled instantaneous communication with remote ships and forces, but also gave all transmitted messages to the enemy.
- ▶ Use of cryptography soars.



Decipherment of
Zimmermann Telegram by
British made American
involvement in World War I
inevitable.

Alan Turing (1912–1954)



Developed foundations of theory of computability (1936).

World War II – Enigma, Purple, JN25, Naval Enigma



- ▶ Cryptography performed by (typically, rotor) *machines*.

World War II – Enigma, Purple, JN25, Naval Enigma



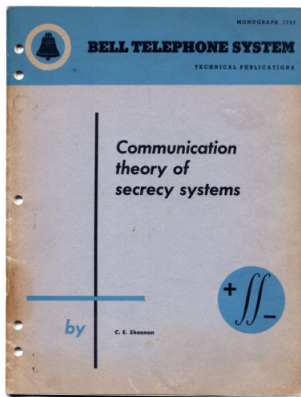
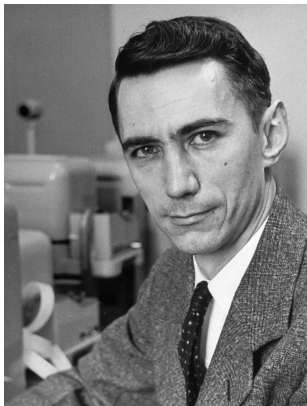
- ▶ Cryptography performed by (typically, rotor) *machines*.
- ▶ Work of Alan Turing and others at Bletchley Park, and William Friedman and others in the USA, on breaking of Axis ciphers had great success and immense impact.

World War II – Enigma, Purple, JN25, Naval Enigma



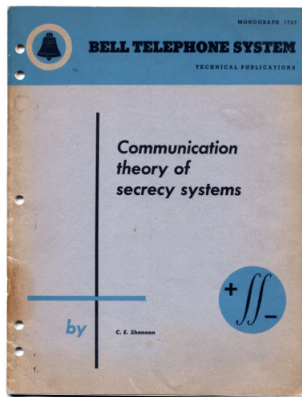
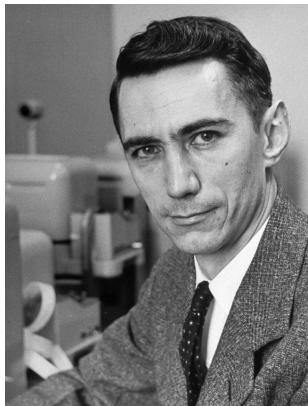
- ▶ Cryptography performed by (typically, rotor) *machines*.
- ▶ Work of Alan Turing and others at Bletchley Park, and William Friedman and others in the USA, on breaking of Axis ciphers had great success and immense impact.
- ▶ Cryptanalytic effort involved development and use of early computers (Colossus).

Claude Shannon (1916–2001)



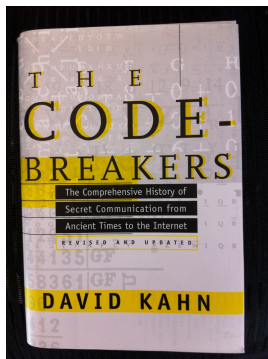
- ▶ “Communication Theory of Secrecy Systems” Sept 1945 (Bell Labs memo, classified).

Claude Shannon (1916–2001)



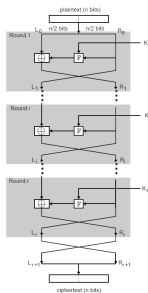
- ▶ “Communication Theory of Secrecy Systems” Sept 1945 (Bell Labs memo, classified).
- ▶ Information-theoretic in character—proves unbreakability of one-time pad. (Published 1949).

Kahn – The Codebreakers



In 1967 David Kahn published
The Codebreakers—The Story of Secret Writing.
A monumental history of cryptography.
NSA attempted to suppress its publication.

DES – U.S. Data Encryption Standard (1976)



DES Designed at IBM; Horst Feistel supplied key elements of design, such as ladder structure. NSA helped, in return for keeping key size at 56 bits.(?)

Computational Complexity



- ▶ Theory of Computational Complexity started in 1965 by Hartmanis and Stearns; expanded on by Blum, Cook, and Karp.
- ▶ Key notions: polynomial-time reductions; NP-completeness.

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

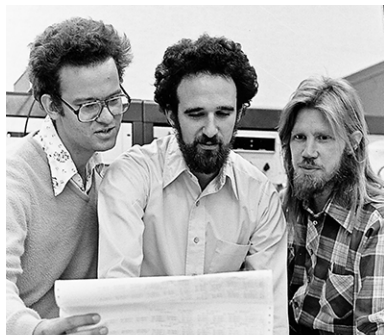
Attacks

More New Directions

What Next?

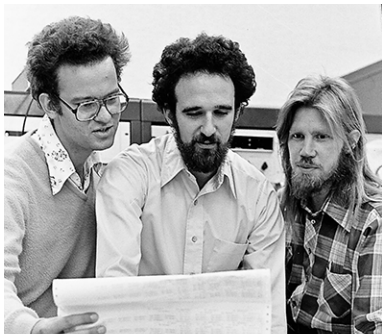
Conclusion and Acknowledgments

Invention of Public Key Cryptography



- ▶ Ralph Merkle, and independently Marty Hellman and Whit Diffie, invented the notion of *public-key cryptography*.

Invention of Public Key Cryptography



- ▶ Ralph Merkle, and independently Marty Hellman and Whit Diffie, invented the notion of *public-key cryptography*.
- ▶ In November 1976, Diffie and Hellman published *New Directions in Cryptography*, proclaiming
“We are at the brink of a revolution in cryptography.”

Public-key encryption (as proposed by Diffie/Hellman)

- ▶ Each party A has a *public key* PK_A others can use to encrypt messages to A :

$$C = PK_A(M)$$

Public-key encryption (as proposed by Diffie/Hellman)

- ▶ Each party A has a *public key* PK_A others can use to encrypt messages to A :

$$C = PK_A(M)$$

- ▶ Each party A also has a *secret key* SK_A for decrypting a received ciphertext C :

$$M = SK_A(C)$$

Public-key encryption (as proposed by Diffie/Hellman)

- ▶ Each party A has a *public key* PK_A others can use to encrypt messages to A :

$$C = PK_A(M)$$

- ▶ Each party A also has a *secret key* SK_A for decrypting a received ciphertext C :

$$M = SK_A(C)$$

- ▶ It is easy to compute matching public/secret key pairs.

Public-key encryption (as proposed by Diffie/Hellman)

- ▶ Each party A has a *public key* PK_A others can use to encrypt messages to A :

$$C = PK_A(M)$$

- ▶ Each party A also has a *secret key* SK_A for decrypting a received ciphertext C :

$$M = SK_A(C)$$

- ▶ It is easy to compute matching public/secret key pairs.
- ▶ **Publishing PK_A does not compromise SK_A !** It is *computationally infeasible* to obtain SK_A from PK_A . Each public key can thus be safely listed in a public directory with the owner's name.

Digital Signatures (as proposed by Diffie/Hellman)

- ▶ Idea: sign with SK_A ; verify signature with PK_A .

Digital Signatures (as proposed by Diffie/Hellman)

- ▶ Idea: sign with SK_A ; verify signature with PK_A .
- ▶ A produces signature σ for message M

$$\sigma = SK_A(M)$$

Digital Signatures (as proposed by Diffie/Hellman)

- ▶ Idea: sign with SK_A ; verify signature with PK_A .
- ▶ A produces signature σ for message M

$$\sigma = SK_A(M)$$

- ▶ Given PK_A , M , and σ , anyone can verify validity of signature σ by checking:

$$M \stackrel{?}{=} PK_A(\sigma)$$

Digital Signatures (as proposed by Diffie/Hellman)

- ▶ Idea: sign with SK_A ; verify signature with PK_A .
- ▶ A produces signature σ for message M

$$\sigma = SK_A(M)$$

- ▶ Given PK_A , M , and σ , anyone can verify validity of signature σ by checking:

$$M \stackrel{?}{=} PK_A(\sigma)$$

- ▶ Amazing ideas!

Digital Signatures (as proposed by Diffie/Hellman)

- ▶ Idea: sign with SK_A ; verify signature with PK_A .
- ▶ A produces signature σ for message M

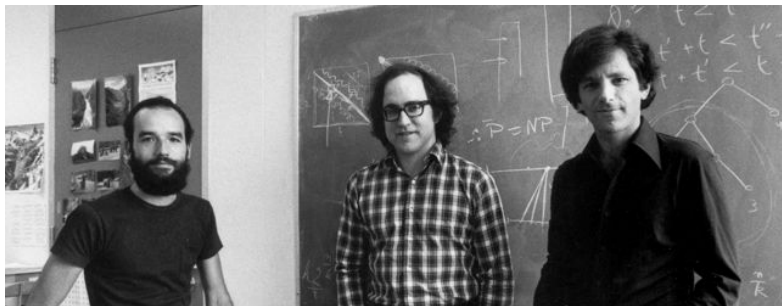
$$\sigma = SK_A(M)$$

- ▶ Given PK_A , M , and σ , anyone can verify validity of signature σ by checking:

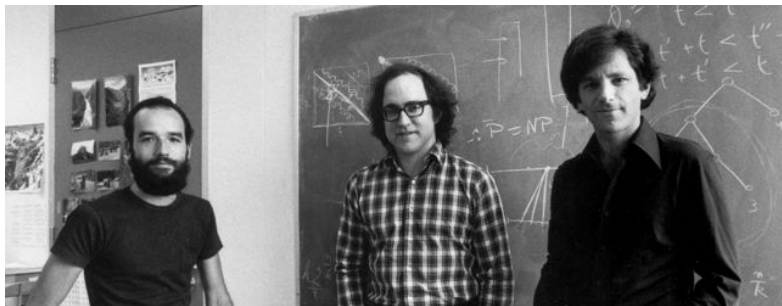
$$M \stackrel{?}{=} PK_A(\sigma)$$

- ▶ Amazing ideas!
- ▶ But they couldn't see how to implement them...

RSA (Ron Rivest, Adi Shamir, Len Adleman, 1977)

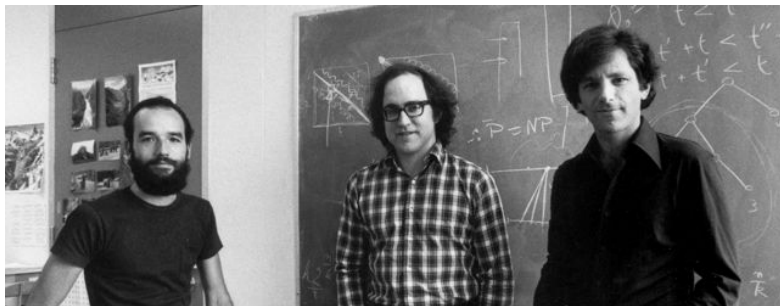


RSA (Ron Rivest, Adi Shamir, Len Adleman, 1977)



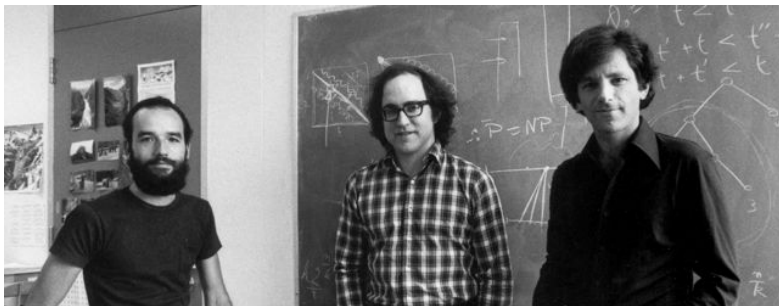
- ▶ Security relies (in part) on inability to factor product n of two large primes p, q .

RSA (Ron Rivest, Adi Shamir, Len Adleman, 1977)



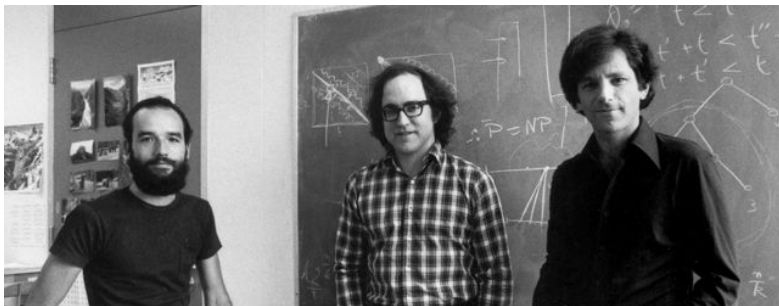
- ▶ Security relies (in part) on inability to factor product n of two large primes p, q .
- ▶ $PK = (n, e)$ where $n = pq$ and $\gcd(e, \phi(n)) = 1$

RSA (Ron Rivest, Adi Shamir, Len Adleman, 1977)



- ▶ Security relies (in part) on inability to factor product n of two large primes p, q .
- ▶ $PK = (n, e)$ where $n = pq$ and $\gcd(e, \phi(n)) = 1$
- ▶ $SK = d$ where $de = 1 \pmod{\phi(n)}$

RSA (Ron Rivest, Adi Shamir, Len Adleman, 1977)

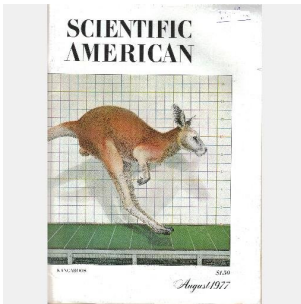
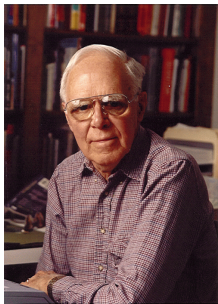


- ▶ Security relies (in part) on inability to factor product n of two large primes p, q .
- ▶ $PK = (n, e)$ where $n = pq$ and $\gcd(e, \phi(n)) = 1$
- ▶ $SK = d$ where $de = 1 \pmod{\phi(n)}$
- ▶ Encryption/decryption (or signing/verify) are simple:

$$C = PK(M) = M^e \pmod{n}$$

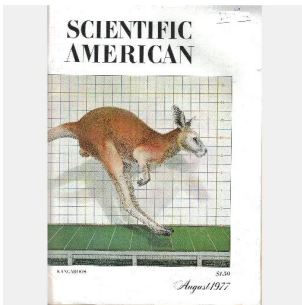
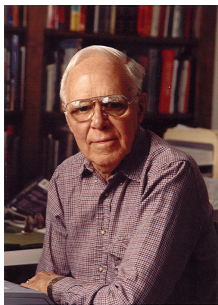
$$M = SK(C) = C^d \pmod{n}$$

Martin Gardner column and RSA-129 challenge



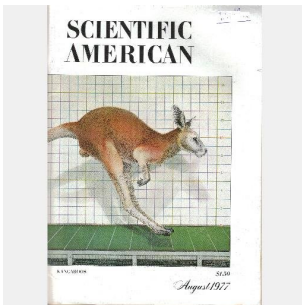
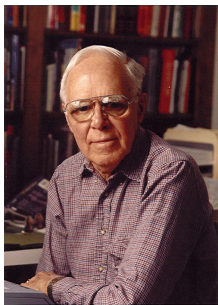
- ▶ Described public-key and RSA cryptosystem in his Scientific American column, *Mathematical Games*

Martin Gardner column and RSA-129 challenge



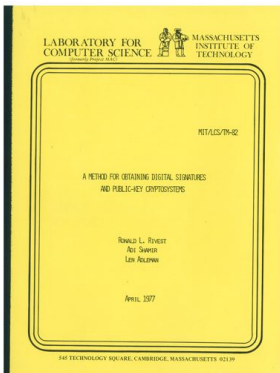
- ▶ Described public-key and RSA cryptosystem in his Scientific American column, *Mathematical Games*
- ▶ Offered copy of RSA technical memo.

Martin Gardner column and RSA-129 challenge



- ▶ Described public-key and RSA cryptosystem in his Scientific American column, *Mathematical Games*
- ▶ Offered copy of RSA technical memo.
- ▶ Offered \$100 to first person to break challenge ciphertext based on 129-digit product of primes.
(Our) estimated time to solution: 40 quadrillion years

Publication of RSA memo and paper



LCS-82 Technical Memo (April 1977) CACM article (Feb 1978)

Programming S.L. Graham, R.L. Rivest*
Editor EBH

A Method for Obtaining Digital Signatures and Public- Key Cryptosystems

R. L. Rivest, A. Shamir, and L. Adleman
MIT Laboratory for Computer Science
and Department of Mathematics

Abstract
An encryption method is presented with the novel property that publicly revealing an encryption key does not thereby reveal the corresponding decryption key. This has two important consequences:

- (1) Confidentiality of communication is achieved by using an encryption key publicly revealed by the intended recipient. Only he can decipher the message, since only he knows the corresponding decryption key.
- (2) A message can be "signed" using a privately held decryption key. Anyone can verify this signature using the corresponding publicly revealed encryption key. Signatures cannot be forged, and a signer cannot later deny the validity of his signature. This has obvious applications to "electronic mail" and "electronic funds transfer" systems. A message is encrypted by representing it as a number M , raising M to a publicly specified power e , and then taking the remainder when the result is divided by a different, secret, power d to yield, where $d \cdot e \equiv 1 \pmod{\phi}$. The security of the system rests in part on the difficulty of factoring the published modulus n .

Key Words and Phrases: digital signatures, public-key cryptosystems, private authentication, security, factorization, prime numbers, electronic mail, message-passing, electronic funds transfer, cryptography.
CR Categories: 2.3.2, 3.3.3, 3.3.6, 3.3.1, 3.2.3

*Detailed permission to make, but not to distribute, copies of all or parts of this memo is granted to individuals registered with the ACM for personal use only. For all other use, permission should be obtained from the ACM. This memo is published by permission of the Association for Computing Machinery. To receive more copies, please write our editorial system, to which you may request specific permission in this specification, or otherwise, contact the Association. Request forms should be sent to ACM, 375 Littleton, New York, New York 10017, and for those of those between postmaster: ACM, 440 Technology Square, Cambridge, Massachusetts 02139. Because of the Association's editorial policies, this memo is published under the name of S. L. Graham, R. L. Rivest, and L. Adleman. Author's Address: MIT Laboratory for Computer Science, 440 Technology Square, Cambridge, MA 02139.
© 1978 ACM 0014-5585/78/0000-0000\$01.50

1. Introduction

The use of "electronic mail" [1] has not been great so far; we must create that two important properties of the current paper are preserved: (a) messages are private, and (b) messages are authentic. We demonstrate in this paper how to build these capabilities into an electronic mail system.

As the kind of our proposal is a new encryption method, this method provides an implementation of a "public-key cryptosystem." An elegant message is used by Diffie and Hellman [1]. Their article motivated our research, since they presented the concept but not any practical implementation of such a system. Readers familiar with [1] may wish to skip directly to Section 5 for a description of our method.

II. Public-Key Cryptosystems

In a "public-key cryptosystem" each user places in a public file an encryption procedure E . That is, the public file is a directory giving the encryption procedure of each user. The user keeps secret the details of his corresponding decryption procedure D . These procedures have the following four properties:

- (1) Disrupting the unencrypted form of a message M yields M . Formally,
 $DE(M) = M$ (1)
- (2) Both E and D are easy to compute.
- (3) By publicly revealing E , the user does not reveal an easy way to compute D . This means that to generate code he can decrypt messages encrypted with E , or compute D efficiently.
- (4) If a message M is first disrupted and then encrypted, M is the result. Formally,
 $ED(M) = M$ (2)

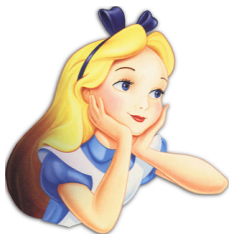
An encryption (or decryption) procedure typically consists of a general method and an encryption key. The general method, under control of the key, enciphers a message M to obtain the unencrypted form of the message, called the ciphertext C . Everyone can use the same general method, the security of a given procedure will rest on the secrecy of the key. Every encryption algorithm thus requires creating the key.

When the user wants to receive a message, he uses the method of computing $D(C)$, using all possible messages M and one such that $DE(M) = C$. If this property (1) is satisfied the number of such messages to test will be so large that this approach is impractical.

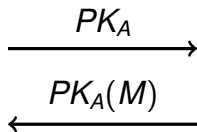
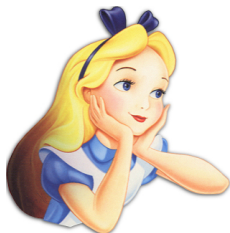
A function f satisfying (a)-(c) is a "trap-door one-way permutation." Diffie and Hellman [1] introduced the concept of trap-door one-way functions that

Communications February 1978
Volume 31
Number 2

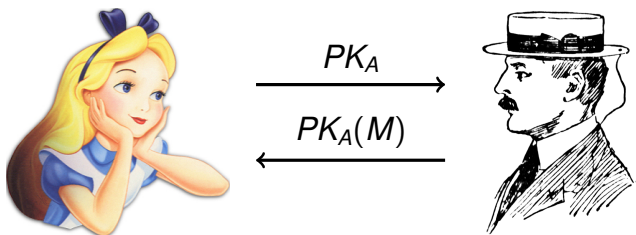
Alice and Bob (1977, in RSA paper)



Alice and Bob (1977, in RSA paper)



Alice and Bob (1977, in RSA paper)



Alice and Bob now have a life of their own—they appear in hundreds of crypto papers, in `xkcd`, and even have their own Wikipedia page:

The screenshot shows the Wikipedia article for "Alice and Bob". The page title is "Alice and Bob" and the subtitle is "From Wikipedia, the free encyclopedia". The article text begins with "The names **Alice and Bob** are commonly used [placeholder names](#) are used for convenience; for example, "Alice sends a message to Party B encrypted by Party B's public key within these fields—helping technical topics to be explained". The article also mentions "In [cryptography](#) and [computer security](#), there are a number of [various protocols](#). The names are conventional, somewhat".

Independent Invention of Public-Key Revealed



In 1999 GCHQ announced that James Ellis, Clifford Cocks, and Malcolm Williamson had invented public-key cryptography, the “RSA” algorithm, and “Diffie-Hellman key exchange” in the 1970’s, before their invention outside.

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

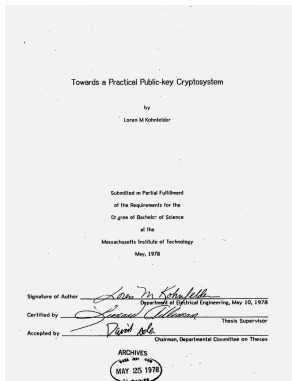
Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

Loren Kohnfelder – Invention of Digital Certificates



- ▶ Loren Kohnfelder's B.S. thesis (MIT 1978, supervised by Len Adleman), proposed notion of *digital certificate*—a digitally signed message attesting to another party's public key.

IACR—International Assn. for Cryptologic Research

- ▶ Established 1982 by David Chaum, myself, and others, to promote academic research in cryptology.
- ▶ Sponsors three major conferences/year (Crypto, Eurocrypt, Asiacrypt) and four workshops; about 200 papers/year, plus another 600/year posted on web. Publishes J. Cryptography
- ▶ Around 1600 members, (25% students), from 74 countries, 27 Fellows.



Theoretical Foundations of Security



- ▶ “Probabilistic Encryption” Shafi Goldwasser, Silvio Micali (1982) (Encryption should be *randomized!*)

Theoretical Foundations of Security



- ▶ “Probabilistic Encryption” Shafi Goldwasser, Silvio Micali (1982) (Encryption should be *randomized!*)
- ▶ “A Digital Signature Scheme Secure Against Adaptive Chosen Message Attacks” Goldwasser, Micali, Rivest (1988) (Uses well-defined *game* to define security objective.)

RC4 stream cipher (Rivest, 1987)

- ▶ RC4 is the most widely used software stream cipher

RC4 stream cipher (Rivest, 1987)

- ▶ RC4 is the most widely used software stream cipher
- ▶ Not public-key; exclusive-ors stream of pseudo-random bytes with plaintext to derive ciphertext.

RC4 stream cipher (Rivest, 1987)

- ▶ RC4 is the most widely used software stream cipher
- ▶ Not public-key; exclusive-ors stream of pseudo-random bytes with plaintext to derive ciphertext.
- ▶ Extremely simple and fast: uses array $S[0..255]$ to keep a permutation of $0..255$, initialized using secret key, and uses two pointers i, j into S .

To output a pseudo-random byte:

```
i = (i + 1) mod 256
```

```
j = (j + S[i]) mod 256
```

```
swap S[i] and S[j]
```

```
Output S[(S[i] + S[j]) mod 256]
```

RC4 stream cipher (Rivest, 1987)

- ▶ RC4 is the most widely used software stream cipher
- ▶ Not public-key; exclusive-ors stream of pseudo-random bytes with plaintext to derive ciphertext.
- ▶ Extremely simple and fast: uses array $S[0..255]$ to keep a permutation of $0..255$, initialized using secret key, and uses two pointers i, j into S .

To output a pseudo-random byte:

$i = (i + 1) \bmod 256$

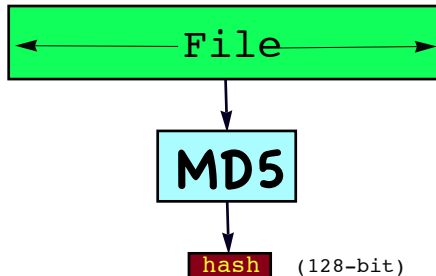
$j = (j + S[i]) \bmod 256$

swap $S[i]$ and $S[j]$

Output $S[(S[i] + S[j]) \bmod 256]$

- ▶ Used in: WEP, BitTorrent, SSL, Kerberos, PDF, Skype, ...

MD5 Cryptographic Hash Function (Rivest, 1991)



- ▶ MD5 proposed as pseudo-random function mapping files to 128-bit fingerprints. (variant of earlier MD4)
- ▶ Collision-resistance was a design goal – it should be infeasible to find two files with the same fingerprint.
- ▶ Many, many uses (e.g. in digital signatures) – very widely used, and a model for many other later hash function designs.

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

U.S. Patent 4,405,829

United States Patent [19]		[11]	4,405,829
Rivest et al.		[45]	Sep. 20, 1983
[54] CRYPTOGRAPHIC COMMUNICATIONS SYSTEM AND METHOD			
[75] Inventors: Ronald L. Rivest, Belmont; Adi Shamir, Cambridge; Leonard M. Adleman, Arlington, all of Mass.			
[73] Assignee: Massachusetts Institute of Technology, Cambridge, Mass.			
[21] Appl. No.: 860,586			
[22] Filed: Dec. 14, 1977			
[51] Int. Cl.: H04K 1/00; H04L 9/04			
[52] U.S. Cl.: 178/22.1; 178/22.11			
[54] Field of Search: 178/22, 22.1, 22.11, 178/22.14, 22.15			
[56] References Cited			
U.S. PATENT DOCUMENTS			
1,657,476 4/1972 Aiken 178/22			
OTHER PUBLICATIONS			
"New Directions in Cryptography", Diffie et al., <i>IEEE Transactions on Information Theory</i> , vol. IT-22, No. 6, Nov. 1976, pp. 644-654.			
"Theory of Numbers" Stewart, MacMillan Co., 1952, pp. 133-135.			
"Diffie et al., Multi-User Cryptographic Techniques", AFIPS, Conference Proceedings, vol. 45, pp. 109-112, Jun. 8, 1976.			
<i>Primary Examiner</i> —Sal Cangialosi <i>Attorney, Agent, or Firm</i> —Arthur A. Smith, Jr.; Robert J. Horn, Jr.			
ABSTRACT			
[57] A cryptographic communications system and method. The system includes a communications channel coupled to at least one terminal having an encoding device and to at least one terminal having a decoding device. A message-to-be-transferred is enciphered to ciphertext at the encoding terminal by first encoding the message as a number M in a predetermined set, and then raising that number to a first predetermined power (associated with the intended receiver) and finally computing the remainder, or residue, C, when the exponentiated number is divided by the product of two predetermined prime numbers (associated with the intended receiver). The residue C is the ciphertext. The ciphertext is deciphered to the original message at the decoding terminal in a similar manner by raising the ciphertext to a second predetermined power (associated with the intended receiver), and then computing the residue, M', when the exponentiated ciphertext is divided by the product of the two predetermined prime numbers associated with the intended receiver. The residue M' corresponds to the original encoded message M.			
40 Claims, 7 Drawing Figures			

```
graph TD
    M((M)) --> Enc[ENCODING]
    Enc --> CA[CA_s]
    CA --> CC[COMMUNICATIONS CHANNEL]
    CC --> CD[CAs]
    CD --> Dec[DECODING]
    Dec --> M_prime((M))
```

Filed December 1977 (MIT TLO)
Issued September 1983

RSA the company (1983)

RSA the company (1983)



- ▶ Jim Bidzos joined in 1986

RSA the company (1983)



- ▶ Jim Bidzos joined in 1986
- ▶ Lotus (1987), Motorola, Apple, Novell, Netscape, Microsoft, ...

RSA the company (1983)



- ▶ Jim Bidzos joined in 1986
- ▶ Lotus (1987), Motorola, Apple, Novell, Netscape, Microsoft, ...
- ▶ RSA Conference series (1991)

RSA the company (1983)



- ▶ Jim Bidzos joined in 1986
- ▶ Lotus (1987), Motorola, Apple, Novell, Netscape, Microsoft, ...
- ▶ RSA Conference series (1991)
- ▶ Verisign spun out in 1995
 - 1.3 billion certificate status checks/day
 - 65 billion DNS requests/day (DNSSEC coming)

RSA the company (1983)



- ▶ Jim Bidzos joined in 1986
- ▶ Lotus (1987), Motorola, Apple, Novell, Netscape, Microsoft, ...
- ▶ RSA Conference series (1991)
- ▶ Verisign spun out in 1995
 - 1.3 billion certificate status checks/day
 - 65 billion DNS requests/day (DNSSEC coming)
- ▶ RSA acquired by Security Dynamics in 1996, now part of EMC.

World Wide Web (Sir Tim Berners-Lee, 1990)



- ▶ Just as radio did, this new communication medium, the World-Wide Web, drove demand for cryptography to new heights.
- ▶ Cemented transition of cryptography from primarily military to primarily commercial.

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

U.S. cryptography policy evolves

- ▶ U.S. government initially tried to control and limit public-sector research and use of cryptography
- ▶ Attempt to chill research via ITAR (1977)
- ▶ MIT “Changing Nature of Information” Committee (1981; Dertouzos, Low, Rosenblith, Deutch, Rivest,...)

MIT Committee Seeks Cryptography Policy

Questions of who should do research on cryptography and how results should be disseminated are the first order of business

Within the next 10 years, networks consisting of tens of thousands of computers will connect businesses, corporations and banks in ways that make

quencies for individuals and for society if computers continue to be connected, as they are now, according to local deci-

easy to send computer programs between connected machines and to instruct a program to search for, select,

Science, 13 Mar 1981

U.S. cryptography policy evolves

- ▶ U.S. government tried to mandate availability of all encryption keys via “key escrow” and/or “Clipper Chip” (1993)

U.S. cryptography policy evolves

- ▶ U.S. government tried to mandate availability of all encryption keys via “key escrow” and/or “Clipper Chip” (1993)



U.S. cryptography policy evolves

- ▶ U.S. government tried to mandate availability of all encryption keys via “key escrow” and/or “Clipper Chip” (1993)



- ▶ Today, US policy leans toward strong cybersecurity, including strong cryptography, for all information systems as a matter of national security.

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

Factorization of RSA-129 (April 1994)

▶ RSA-129 =

```
11438162575788886766923577997614661201021829  
67212423625625618429357069352457338978305971  
23563958705058989075147599290026879543541
```


Factorization of RSA-129 (April 1994)

▶ RSA-129 =

11438162575788886766923577997614661201021829
67212423625625618429357069352457338978305971
23563958705058989075147599290026879543541

▶ Derek Atkins, Michael Graff, Arjen Lenstra,
Paul Leyland: RSA-129 =

34905295108476509491478496199038981334177646
38493387843990820577 x
32769132993266709549961988190834461413177642
967992942539798288533

Factorization of RSA-129 (April 1994)

- ▶ RSA-129 =

```
11438162575788886766923577997614661201021829  
67212423625625618429357069352457338978305971  
23563958705058989075147599290026879543541
```

- ▶ Derek Atkins, Michael Graff, Arjen Lenstra,
Paul Leyland: RSA-129 =

```
34905295108476509491478496199038981334177646  
38493387843990820577 x  
32769132993266709549961988190834461413177642  
967992942539798288533
```

- ▶ 8 months work by about 600 volunteers from more than 20 countries; 5000 MIPS-years.

Factorization of RSA-129 (April 1994)

- ▶ RSA-129 =

```
11438162575788886766923577997614661201021829  
67212423625625618429357069352457338978305971  
23563958705058989075147599290026879543541
```

- ▶ Derek Atkins, Michael Graff, Arjen Lenstra,
Paul Leyland: RSA-129 =


```
34905295108476509491478496199038981334177646  
38493387843990820577 x  
32769132993266709549961988190834461413177642  
967992942539798288533
```

- ▶ 8 months work by about 600 volunteers from more than 20 countries; 5000 MIPS-years.

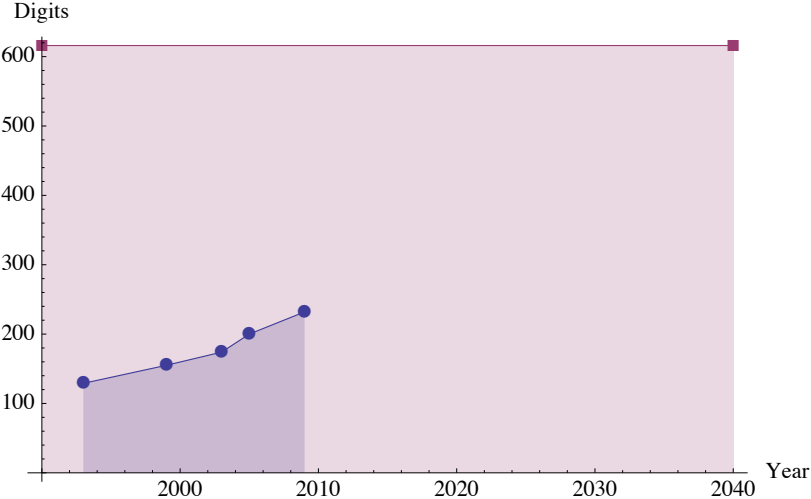
- ▶ secret message:

The Magic Words Are Squeamish Ossifrage



BayBank For Solving the Scientific American RSA Challenge		0254643
Massachusetts	53-235 113	Official Bank Check
Date		April 22, 1994
PAY	The sum of 100 dollars 00 cts	\$ *****100.00*****
AMOUNTS IN EXCESS OF \$100,000.00 REQUIRE TWO SIGNATURES		
To the order of	**Derek Atkins or Michael Graff or Arjen Lenstra or Paul Leyland**	 Authorized Signature
		Authorized Signature
⑈0254643⑈ ⑆011302357⑆ 317 83321⑈		

Factoring Records



Factoring on a Quantum Computer?



$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$
$$|\alpha|^2 + |\beta|^2 = 1$$

The diagram shows a circle with a diagonal arrow pointing from the bottom-left to the top-right. This is equal to the sum of two circles: one with a vertical arrow pointing up and one with a vertical arrow pointing down.

$$\alpha|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \beta|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

In 1994, Peter Shor invented a fast factorization algorithm that runs on a (hypothetical) *quantum computer* and works by determining multiplicative period of elements mod n .

Factoring on a Quantum Computer?



$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$
$$|\alpha|^2 + |\beta|^2 = 1$$

The diagram shows a quantum state $|\psi\rangle$ represented by a circle with a diagonal arrow. This is equal to the sum of two states: $\alpha|0\rangle$ (circle with an upward vertical arrow) and $\beta|1\rangle$ (circle with a downward vertical arrow). The corresponding column vectors are $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

In 1994, Peter Shor invented a fast factorization algorithm that runs on a (hypothetical) *quantum computer* and works by determining multiplicative period of elements mod n .

- ▶ In 2001, researchers at IBM used this algorithm on a (real) quantum computer to factor $15 = 3 \times 5$.

Factoring on a Quantum Computer?



$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$
$$|\alpha|^2 + |\beta|^2 = 1$$

The diagram shows a circle with a diagonal arrow pointing from the bottom-left to the top-right. This is equal to the sum of two circles: one with a vertical arrow pointing up and one with a vertical arrow pointing down.

In 1994, Peter Shor invented a fast factorization algorithm that runs on a (hypothetical) *quantum computer* and works by determining multiplicative period of elements mod n .

- ▶ In 2001, researchers at IBM used this algorithm on a (real) quantum computer to factor $15 = 3 \times 5$.
- ▶ Dark clouds on horizon for RSA?

Hash Function Attacks



- ▶ In 2004 Xiaoyun Wang and colleagues found a way to produce *collisions* for MD5:

$$\text{MD5}(\textit{file1}) = \text{MD5}(\textit{file2}) \quad !!!$$

Also for SHA-1 and many other hash functions.
Major break!!

Hash Function Attacks



- ▶ In 2004 Xiaoyun Wang and colleagues found a way to produce *collisions* for MD5:

$$\text{MD5}(\textit{file1}) = \text{MD5}(\textit{file2}) \quad !!!$$

Also for SHA-1 and many other hash functions.
Major break!!

- ▶ NIST now running competition for new hash function standard (SHA-3).

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

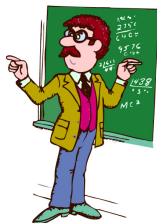
Many new research problems and directions

- ▶ secret-sharing
- ▶ anonymity
- ▶ commitments
- ▶ multi-party protocols
- ▶ elliptic curves
- ▶ crypto hardware
- ▶ key leakage
- ▶ proxy encryption
- ▶ crypto for smart cards
- ▶ password-based keys
- ▶ random oracles
- ▶ oblivious transfer
- ▶ ...
- ▶ zero-knowledge proofs
- ▶ payment systems
- ▶ voting systems
- ▶ homomorphic encryption
- ▶ lattice-based crypto
- ▶ private information retrieval
- ▶ public-key infrastructure
- ▶ concurrent protocols
- ▶ randomness extractors
- ▶ tweakable encryption
- ▶ differential cryptanalysis
- ▶ identity-based encryption
- ▶ ...

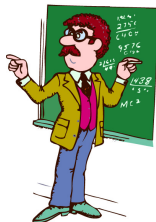
Many new research problems and directions

- ▶ secret-sharing
- ▶ anonymity
- ▶ commitments
- ▶ multi-party protocols
- ▶ elliptic curves
- ▶ crypto hardware
- ▶ key leakage
- ▶ proxy encryption
- ▶ crypto for smart cards
- ▶ password-based keys
- ▶ random oracles
- ▶ oblivious transfer
- ▶ ...
- ▶ zero-knowledge proofs
- ▶ payment systems
- ▶ voting systems
- ▶ homomorphic encryption
- ▶ lattice-based crypto
- ▶ private information retrieval
- ▶ public-key infrastructure
- ▶ concurrent protocols
- ▶ randomness extractors
- ▶ tweakable encryption
- ▶ differential cryptanalysis
- ▶ identity-based encryption
- ▶ ...

Zero-Knowledge Proofs

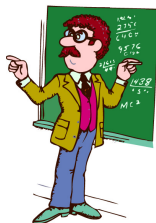


Zero-Knowledge Proofs



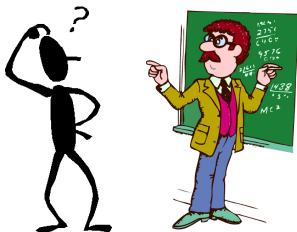
I can convince you

Zero-Knowledge Proofs



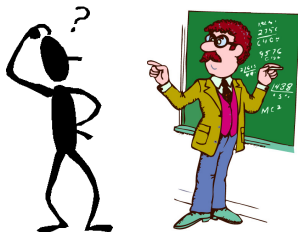
*I can convince you
I know a solution
to a hard problem*

Zero-Knowledge Proofs



*I can convince you
I know a solution
to a hard problem
while telling you nothing
about my solution*

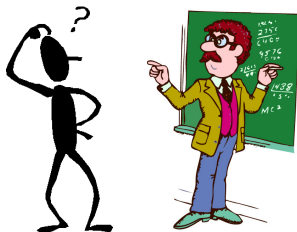
Zero-Knowledge Proofs



*I can convince you
I know a solution
to a hard problem
while telling you nothing
about my solution
even if you are very skeptical!*

*Goldwasser, Micali, Rackoff (1985)
Goldreich, Micali, Wigderson (1986)*

Zero-Knowledge Proofs

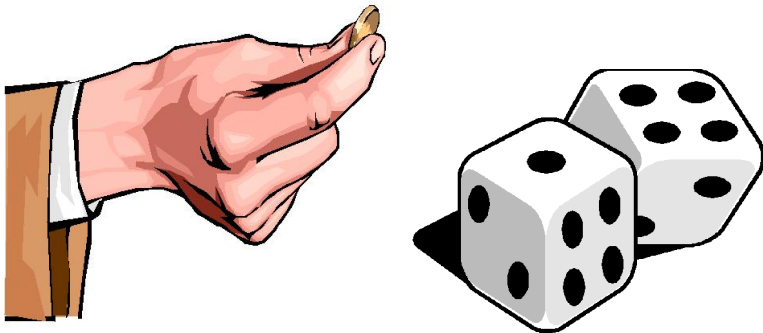


*I can convince you
I know a solution
to a hard problem
while telling you nothing
about my solution
even if you are very skeptical!*

*Goldwasser, Micali, Rackoff (1985)
Goldreich, Micali, Wigderson (1986)*

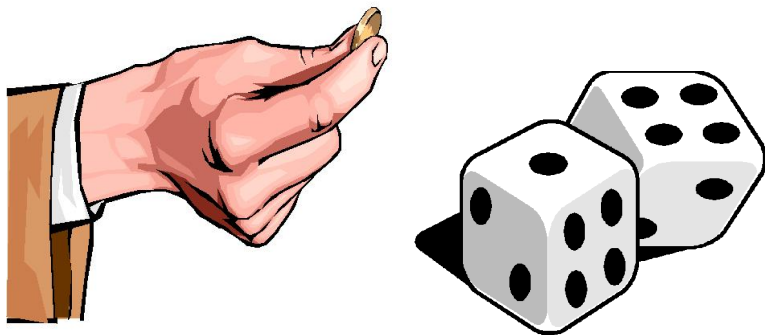
An enormously useful capability!

Probabilistic MicroPayment System



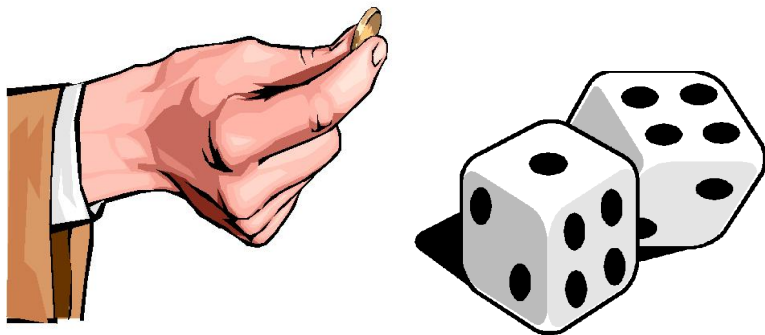
- ▶ Micali and Rivest (2001)

Probabilistic MicroPayment System



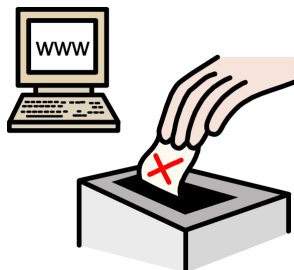
- ▶ Micali and Rivest (2001)
- ▶ Paying ten cents \equiv paying \$1 with probability $1/10$.
Uses pseudorandom digital signatures for “verifiable fair dice.”

Probabilistic MicroPayment System



- ▶ Micali and Rivest (2001)
- ▶ Paying ten cents \equiv paying \$1 with probability $1/10$.
Uses pseudorandom digital signatures for “verifiable fair dice.”
- ▶ Peppercoin Company founded 2001, sold in 2007.

Voting Systems



New “end-to-end” cryptographic voting systems (Chaum, Neff, Benaloh, Ryan, Rivest, Adida, ...):

- ▶ all ballots posted on web (encrypted)
- ▶ voters verify their votes are correct (while preventing vote-selling and coercion)
- ▶ anyone can verify final tally
- ▶ may be done with paper ballots

Cryptography *increases* transparency and verifiability!

Fully Homomorphic Encryption



- ▶ In 1978, Rivest, Adleman, and Dertouzos asked, *“Can one compute on encrypted data, while keeping it encrypted?”*

Fully Homomorphic Encryption



?



!

- ▶ In 1978, Rivest, Adleman, and Dertouzos asked, *“Can one compute on encrypted data, while keeping it encrypted?”*
- ▶ In 2009, Craig Gentry (Stanford, IBM) gave solution based on use of lattices. If efficiency can be greatly improved, could be huge implications (e.g. for cloud computing).

Outline

Some pre-1976 context

Invention of Public-Key Crypto and RSA

Early steps

The cryptography business

Crypto policy

Attacks

More New Directions

What Next?

Conclusion and Acknowledgments

Challenges

- ▶ Make more crypto theory results practical
- ▶ Is factoring really hard?
- ▶ Minimize assumptions
- ▶ Show $P \neq NP$!
- ▶ Is quantum computing practical?
- ▶ Give Alice and Bob smartphones!
- ▶ Ground crypto practice better in vulnerable computer systems

Stata Center Time Capsule Crypto Puzzle



“Time Capsule”
(really, a large lead bag)
sealed at LCS/AI 35th Celebration
April 1999

- ▶ Contains many historic LCS/AI artifacts
- ▶ There is an associated cryptographic puzzle I designed to be solvable around 2034, given predicted advances in computational power.
- ▶ When puzzle is solved, Time Capsule will be opened.

Conclusions

- ▶ Cryptography is not the solution to all of our cybersecurity problems, but it is an essential component of any solution.

Conclusions

- ▶ Cryptography is not the solution to all of our cybersecurity problems, but it is an essential component of any solution.
- ▶ Research in cryptography is a fascinating blend of mathematics, statistics, theoretical computer science, electrical engineering, and psychology.

Conclusions

- ▶ Cryptography is not the solution to all of our cybersecurity problems, but it is an essential component of any solution.
- ▶ Research in cryptography is a fascinating blend of mathematics, statistics, theoretical computer science, electrical engineering, and psychology.
- ▶ While we have accomplished a lot in a few decades, much remains to be done.

Conclusions

- ▶ Cryptography is not the solution to all of our cybersecurity problems, but it is an essential component of any solution.
- ▶ Research in cryptography is a fascinating blend of mathematics, statistics, theoretical computer science, electrical engineering, and psychology.
- ▶ While we have accomplished a lot in a few decades, much remains to be done.
- ▶ Like Alice and Bob, cryptography is here to stay.

Conclusions

- ▶ Cryptography is not the solution to all of our cybersecurity problems, but it is an essential component of any solution.
- ▶ Research in cryptography is a fascinating blend of mathematics, statistics, theoretical computer science, electrical engineering, and psychology.
- ▶ While we have accomplished a lot in a few decades, much remains to be done.
- ▶ Like Alice and Bob, cryptography is here to stay.
- ▶ Cryptography is fun!

Acknowledgments

L. Adleman, R. Anderson, D. Bailey, M. Bellare, J. Benaloh, J. Bidzos, M. Bond, K. Bowers, M. Burmeister, J. Brainard, R. Canetti, D. Chaum, J. Clulow, M. Dertouzos, S. Devadas, Y. Dodis, W. Diffie, C. Ellison, M. Gardner, O. Goldreich, S. Goldwasser, M. Hellman, P. Herson, M. Jakobsson, A. Juels, L. Knudsen, B. Lampson, R. Lee, R. Merkle, S. Micali, A. Oprea, M. Rabin, T. Rabin, L. Reyzin, V. Rijmen, M. Robshaw, P. Ryan, A. Shamir, A. Sherman, R. Silverman, M. Sudan, D. Sutherland, M. Szydlo, Y. Tauman, L. Trevisan, E. Tromer, S. Vadhan, M. Van Dijk, D. Wagner, P. Vora, H. Wee, T. Ylonen, M. Yung

Students: B. Adida, S. Boyack, V. Boyko, M. Burnside, R. Carback, B. Chor, D. Chau, J. Clark, D. Clarke, C. Crutchfield, J. Elien, A. Essex, K. Fleming, M. Fredette, K. Fu, B. Gassend, J. Herzog, S. Hohenberger, A. Khan, B. Kaliski, T. Kotwal, J. Krishnamurthy, A. Lin, A. Lysyanskaya, M. Liskov, A. Maywah, S. Popoveniuc, Z. Ramzan, E. Shen, R. Sidney, J. Sukha, S. Weis, Y. Yin

Be Blackburn

Mom and Dad, Gail, Alex, Chris.

Thank You!