Foundations of Cryptography

MIT-6.875/18.425 , UCB CS-276 Lecture 1

Shafi Goldwasser –MIT, UCB
Raluca Ada Popa-UCB
Vinod Vaikuntanathan-MIT

Adminstrivia

TAs

- Nick Ward: UCB
- Ofer Grossman: MIT
- Lisa Yang: MIT
- Rachel Zhang: MIT

Course Secretary:

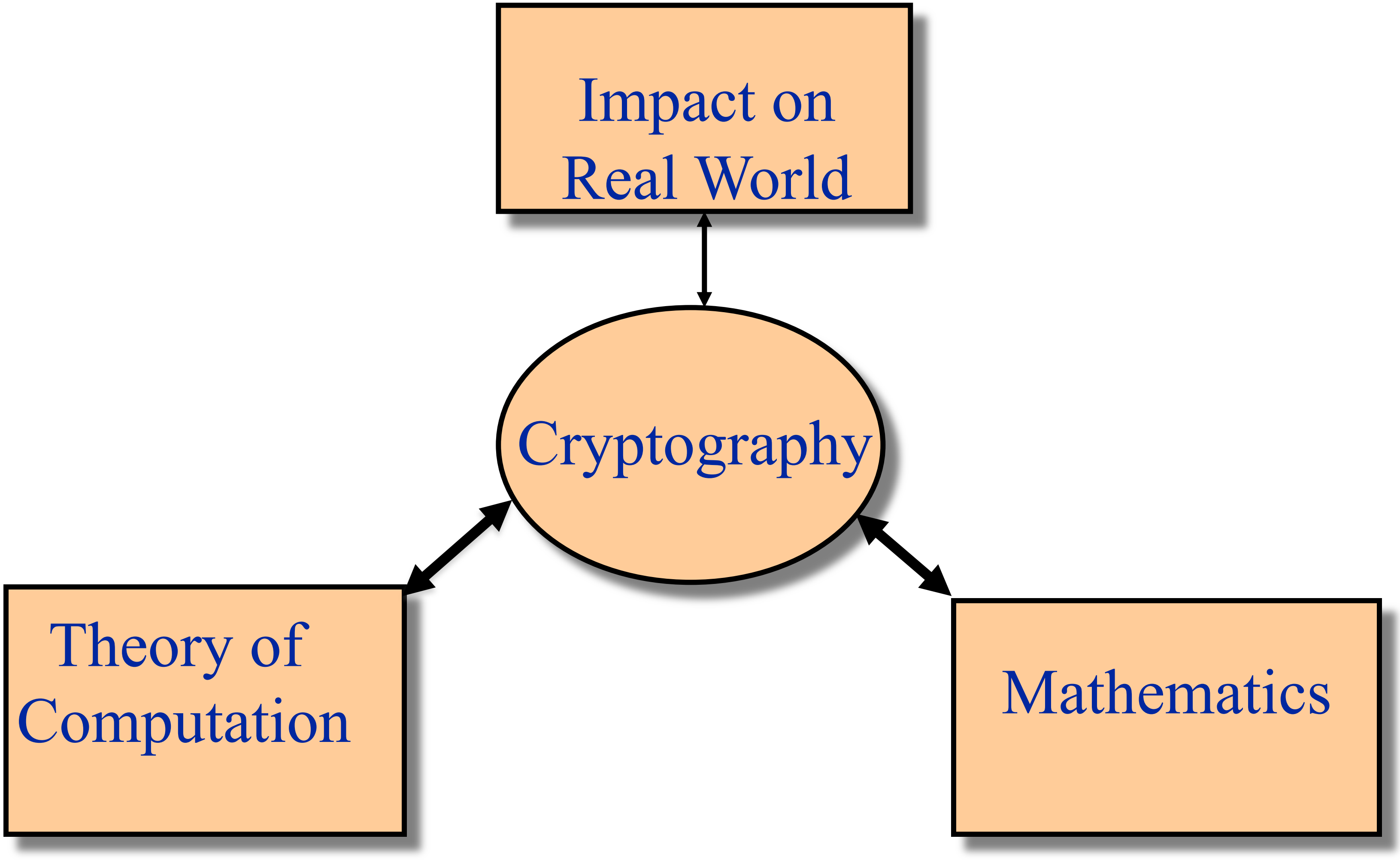
- Debbie Lehto

Website

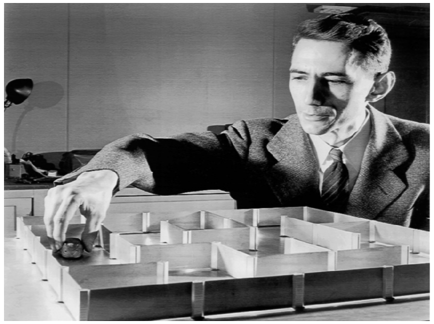
Expectations

- Homework: 6 problem sets every 2 weeks, typed using latex for equations
- Attendance (with exception to those in different time zones) and Participation
- Knowledge: intro to algorithms, probability, mathematical maturiyu

Theory and Practice



Historically



Shannon

"A Mathematical Theory of Communication"(1948)
"A Communication Theory of Secrecy Systems" (1945)

Turing

Inventor of the Universal computing machine
Theory and Practice: Breaking the enigma

War Time Research

Modern Cryptography:

- **Classical** war time effort

- **Modern** with the rise of the internet to enable secure electronic commerce transactions (DiffieHellman 1976, RivestShamirAdleman 1977)

- **Current & Future** enable utilization of remote **computing** and availability of large amounts of data while maintaining our basic right to “be left alone”: privacy

Communication & Computation

Communication: Privacy, Integrity, Authenticity

Computation: Privacy & Correctness of
- Input Data
- Programs and Executions

Catalyst notions and techniques that led to a series of leaps in Complexity Theory
- Pseudo Randomness
- Interactive and Probabilistic Proof Verification
- Average Case vs. Worst Case Hardness

Theory Focus

1. Careful **Definitions** of Cryptographic Tasks and Adversary Models

2. **Critic** of Existing Systems in light of above

3. **Design** systems which can be **proved** secure with respect to definitions made

4. Often Security Proofs are: efficient reductions to explicit assumptions on the complexity of some computational hard problems (or simpler cryptohgraphic primitives)

Design cryptographic systems so science wins either way

Methodology: Efficient Reductions

Given any adversary Strategy to **break** the system in time T(k) with prob. a	→	Construct an algorithm solving the hard problem in time T' = poly (T(k)) with prob a/poly (k)

Which Hard Problems

NP-Hard? No. Worst Case hardness is not enough

Require: Problems which are Average Case Hard

Hard Problems

- Number Theory

Hardy, ‘A Mathematician’s Apology” writes:
“Both Gauss and lesser mathematicians may be justified in rejoicing that there is one such science [number theory] at any rate, whose very remoteness from ordinary human activities should keep it gentle and clean”

No longer: Number theory is the basis of modern security systems

Most recent: Geometry and Coding are the basis of post-quantum systems

Topics: 1976-onward

- Public Key Encryption: Sending Secret Messages without ever Meeting
- Digital Signatures: Signing Contracts Remotely
- Pseudo Random Number Generation Indistinguishable from random

Derandomization

- Zero Knowledge Proofs: Proofs that Reveal Nothing But the Truth (modern use: Block Chains)
- Two Party Secure Computation: coin flipping, oblivious transfer, secure function evaluation
- Multi Party Secure Protocols: Computing on Distributed Secret Data Revealing Nothing but the result without referees, Private Information Retrieval

Byznatine Agreement

- Fully Homomorphic Encryption
- Private Machine Learning using all of the above

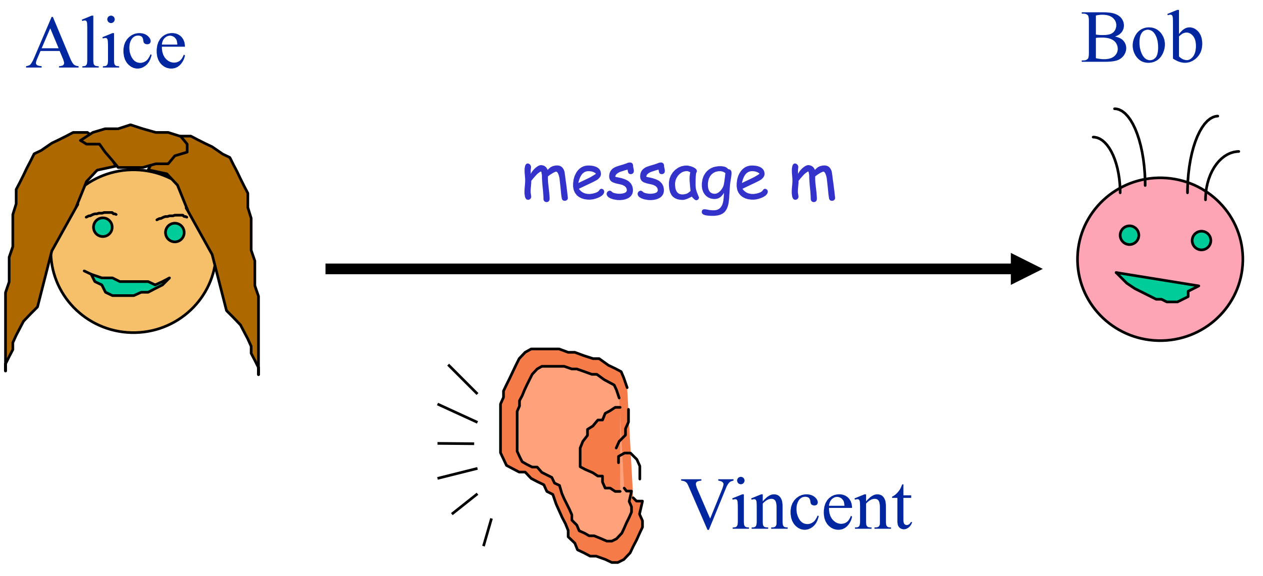
Unifying Theme: The Presence of a Worst Case Adversary

- Integral Part of the Definition of the Problem
- Determines the Quality of Acceptable Solutions

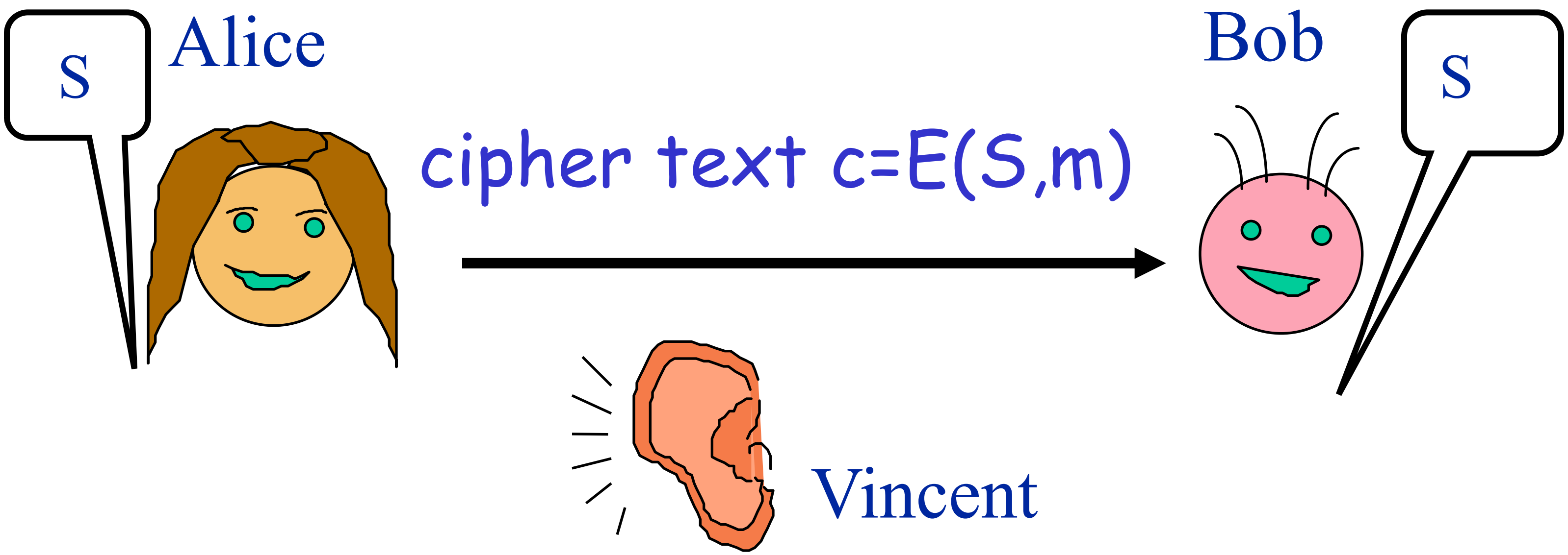
What Can you Get from This Course

- We are not going to be able to cover everything
- Main goals
  - Exposure to the "mindset" of security
    - Identify the Adversary
    - Identify the goal
    - Evaluate Security
  - In Depth: "Basic" cryptography & protocols
  - Exposure: current trends
- If nothing else, a healthy dosage of paranoia…

Secret Communication



Secret Communication



Alice and Bob met to agree on a secret key S

Define Encryption scheme

- An encryption scheme (G,E,D) is a triplet of (possibly probabilistic) algorithms where
  - key generation $G(1^n)$ outputs secret key sk of length n

    [n is also called the security parameter]
  - Encryption algorithm E(sk,m) outputs ciphertext c
  - Decryption algorithm D(sk,c) outputs plaintext m

- Requirements:
  - Correctness: D(sk,E(sk,m)) =m for all m in M.
  - Security Definition…with respect to adversaries

- K = key probability space, Prob[K=sk]
- *M* = message probability space, Prob[M=m]
- C = ciphertext probability space. Prob[C=c] = Prob[E(K,M)=c]

Ancient Codes

Secret Key:

A → T
B → U
…
S → L
…

"Pen and Paper Cryptography"

``MAX YTNEM, WXTK UKNMNL, EBXL GHM BG HNK LMTKL UNM BG HNKLXSCXL''

ciphertext

``THE FAULT, DEAR BRUTUS, LIES NOT IN OUR STARS BUT IN OURSELVES''

plaintext

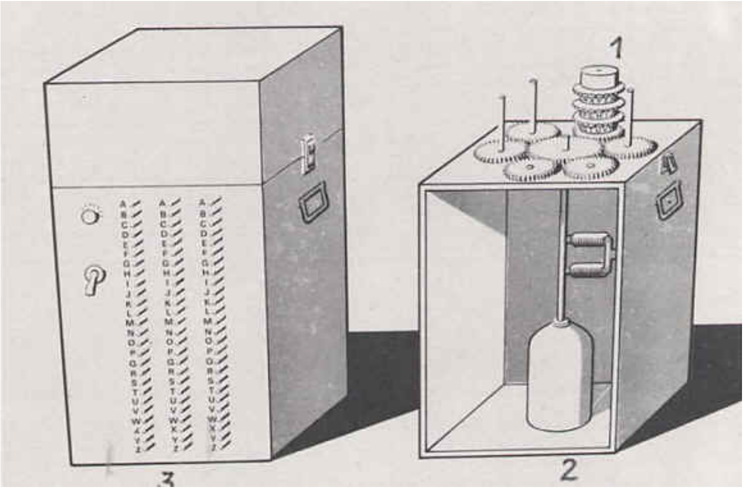
Security? Easy to break, by frequency analysis,

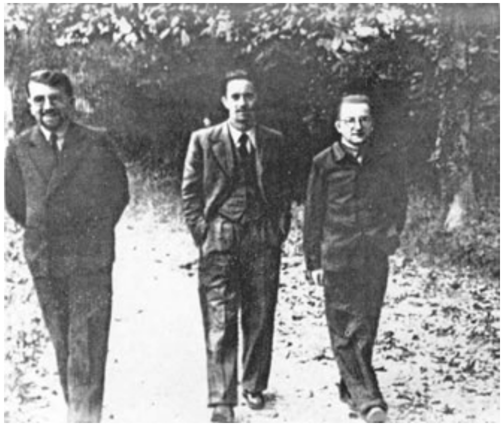
Enigma Machine

Electro-mechanical Devices

Automated Cryptography &

Cryptanalysis





Rejewski, Zygalski, Rozycki

Mid Century: From Art to Science

Shannon '49: Perfect Secrecy Theory

Adversary: unbounded computationally, security analysis is information theoretic

What Does the Adversary Know?

- Kerckohoff Law: A cryptographic system should be secure even if everything about the system (e.g. the algorithms G,E and D in the context of a secrecy system)is known to the adversary <u>except for the key and the randomness of the legal users</u>

- Ciphertext Only: Can see c transmitted over an insecure channel (but not request c for m of its choice)

What Security Guarantee Do We Want?

It should be impossible to

- compute plaintext from cipher text
- Compute the i-th bit of the plaintext
- compute any partial information about the plaintext from the cipher text.
- compute relations between plaintexts

} For any message space, with high probability

How do we define that?

Shannon Secrecy Definition (aka perfect secrecy)

Let EVE be an unbounded adversary.

Note 1: $C=E(K,M)$

We say that (G,E,D) satisfies Shannon-secrecy if and only if:

$\forall$ probability distribution over M,

$\forall$ c in C, $\forall$ m in M

$$\Pr[M=m] = \Pr[M=m \mid E(K,M)=c]$$

A-priori = A-posteriori

Note 2: When a r.v. (random variable) Appears in a context of prob statement., the prob is taken over the choices of the r.v.

Slight Notational Abuse: All capital letters denote r.v's and prob distribution at the same time

Perfect Indistinguishability Alternative Security Definition

Let EVE be an unbounded adversary.

We say that (G,E,D) satisfies

Perfect indistinguishability if :

$\forall$ Probability distribution over $\mathcal{M}$

$\forall$ m, m' in $\mathcal{M}$,

$\forall$ c in $\mathcal{C}$

$\Pr[E(K,m)=c] = \Pr[E(K,m')=c]$

Note : EVE is not used
In the definition but
Is implicitly there computing
probabilities…

The Definitions are Equivalent

Theorem:

(G,E,D) satisfies perfect indistinguishability iff (G,E,D) satisfies Shannon secrecy.

Proof: Simple use of Bayes Theorem

Indistinguishability implies Shannon

For all m, m',c perfect indistinguishability guarantees that Pr[E(K, m)=c]=Pr[E(K, m')=c] =[call it α]

fact1 $\Pr[E(K,M)=c]=\Sigma_m \Pr[M=m]\Pr(E(K,m)=c]=$

$\Sigma_m \Pr(M=m)\alpha = \alpha\Sigma_m \Pr(M=m) = \alpha$

Bayes: P[A|B]=Pr[B|A] Pr[A]/Pr[B]

For all m: A-posteriori

Pr[M=m|E(K,M) = c]= (Bayes)

Pr(E(K,M)=c|M=m)Pr(M=m)/Pr[E[K,M]=c]= (fact1)

Pr[E(K,m)=c] Pr(M=m) /α= (def of indistinguishability)

αPr(M=m)/α= Pr[M=m] = A=priori QED

Shannon implies indistinguishability

Bayes: $P[A|B]=Pr[B|A]\ Pr[A]/Pr[B]$

For all m,c Shannon secrecy guarantees that

$Pr[M=m] = Pr[M=m|\ E(K,M)=c]$ for all m

For all m,

$Pr[E(K,m)=c]=$ (rewrite)

$Pr[E(K,M)=c\ |\ M=m] =$ (Bayes)

$\underline{Pr[M=m|E(K,M)=c]}Pr[E(K,M)=c]/Pr[M=m]=$ (def of Shannon)

$\underline{Pr[M=m]}\ Pr[E(K,M)=c]/Pr[M=m] =$

$Pr(E(K,M)=c]$

This is also true for m'. Namely, $Pr[E(K,m')=Pr[E(K,M)=c]$

Thus, for all m, m',c; $Pr[C=c|M=m]=Pr[C=c\ |\ M=m']$ QED

Shannon Secrecy is Achievable

One Time Pad: G chooses sk at random in $\{0,1\}^n$

$$E(sk,m)=sk\oplus m, \; D(sk,c)=sk\oplus c$$

Claim: One Time Pad Achieves Shannon Security

Proof: Fix m, c $\in\{0,1\}^n$.

$$\text{Prob}_{[}E(K,m)=c] = \text{Prob}\,[K\oplus m=c] = \text{Prob}[K=m\oplus c]=1/2^n$$

Thus, $\forall$c, m, m'

$$\text{Prob}(E(K,m)=c) = \text{Prob}(E(K,m')=c)$$

And one-time pad (G,E,D) achieves perfect indistinguishability $\Rightarrow$ Shannon secrecy.

How about using one-time pad to send more than one message?

Q: Would it preserve Shannon Secrecy?
A: No

Proof: Show Perfect Indistinguishability no longer holds.

Consider the case of two messages each of length n, each encrypted by "xoring" the message with the same sk.

Claim: there exists m=(m1, m2) & m'=(m1',m2') & ciphertext c=(c1,c2) such that $Pr[E(K,m)=c] \neq Pr_{SK}[E(K,m')=c]$

Pf: Set m1=m2 and m1'≠m2' and c=(c1,c1) .Then,
m1'≠m2' ⇒there is no sk for which $sk \oplus m1' = c1 = sk \oplus m2'$

$\Rightarrow Pr[E(K,m')=c]=0$

But there exist sk s.t. $sk \oplus m1 = c1$ and $sk \oplus m2 = c1$

$\Rightarrow Pr[E(K,m)=c]>0$ QED.

#Keys $\geq$ #Messages

Shannon Theorem: For perfect secrecy schemes, $|K| \geq |M|$

Note:
$|K|$= number of distinct keys
$|M|$=number of distinct messages

Proof: Suppose not and $|K| < |M|$.
Fix c s.t. Pr[E(K,M)=c]>0.

Let M_c={m s.t. ∃ some k for which m= D(k, c)}.
Then $|M_c| \leq |K|$ (since there is at least 1 key per message)
$< |M|$ (assumed for contradiction)

So, ∃some m' ∈ M for which there is no k
that yields m'=D(k,c). Namely, Pr (E(K,m')=c)=0

Whereas Pr(E(K,M)=c) >0, so there exists another m, s.t.
Pr[E(K,m)=c]>0 . Perfect Indistinguishability is violated.
Contradiction QED

$|K| \geq |M| \Rightarrow$

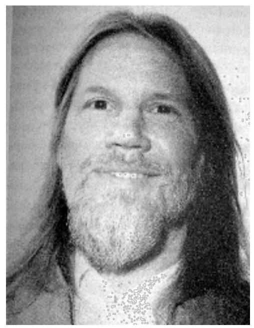
bits to specify Key $\geq$
bits to specify Message

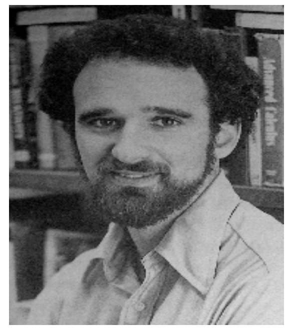
Disadvantages of One Time Pads

- The size of the key is huge: as many key bits as message bits and need to know in advance how many message bits
- Receiver needs to know which key goes with which ciphertext (some synchronization or state)
- Advantage
  - By Shannon's Theorem, this is BEST POSSIBLE.

Modern Cryptography

1976, New Directions in Cryptography



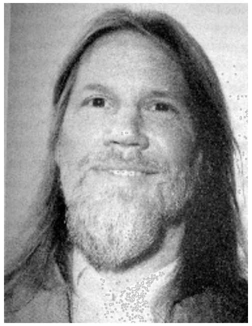
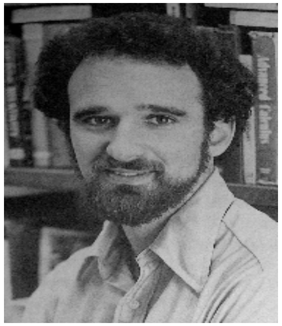


“ We stand today on the brink of a revolution in cryptography. The development of cheap digital hardware has freed it from the design limitations of mechanical computing and brought the cost of high grade cryptographic devices down to where they can be used in such commercial applications as remote cash dispensers and computer terminals. In turn, such applications create a need for new types of cryptographic systems ”

W. Diffie, M. Hellman, “New Directions in Cryptography”, 1976.

Modern Cryptography

1976, New Directions in Cryptography

The Adversary

Any probabilistic polynomial time algorithm: $O(n^c)$ for some $c>0$ for n=security parameter. Think of n=size of the secret key

Probabilistic Polynomial Time algorithms (PPT)

- A runs in polynomial time in its input length
- A is randomized: can flip fair coins
  - Las Vegas: $\forall$input, A is correct or with negligible probability A outputs $\perp$
  - Monte Carlo: $\forall$input, A is correct With all but negligible probability

Can Now Ask New Questions

1. Can A and B agree on key sk in person and subsequently exchange P(|sk|) messages where P is any polynomial?
2. Can A and B exchange messages without even meeting
3. Can B be assured that A's message was not modified: can A sign messages digitally so that B can verify that A signed the message, without A and B meeting

Possible for the new Adversary model and modified security definition

Conventions

- We say that a function $\varepsilon(n)$ is negligible if for every polynomial P, there exists n_0 s.t. for all $n>n_0$, $\varepsilon(n)<1/P(n)$
- We say that a function $\varepsilon(n)$ is non-negligible if there exists a polynomial P, such that for infinitely many k, $\varepsilon(n)>1/P(n)$
- Instead of "there exists a n_0 s.t. for all $n>n_0$ ", we often say "for sufficiently large n"
- b $\in_R$ {0,1} means "sampled at random" (often omitted)

Notations

PPT: Probabilistic Polynomial Time Algorithms. They can toss coins; different outputs are possible for the same input; and on length n input, the running time is bounded by $O(n^c)$ for some constant $c>0$.

Negligible neg(n): $< 1/p(n)$ for all polynomials p

non-neg: There exists a polynomial p s.t. non-neg(n)>1/p(n)

Security Parameter: is always presented in Unary

There Exists: ∃

For All: ∀

Such that: s.t.

|n|: number of bits in binary representation of n ,e.g. |8|=3

Big O-notation:

|S| : Cardinality of Set S

Prob (E), Pr[E]: probability that event E is true

iff : if and only if

o.w: other wise