Public key crypto (quick intro)
Provable cryptography

Slides from Bart Preneel and Phil Rogaway

Comp527 status

- Hack-a-Vote phase 2 complete
  - Scott will make everything public
  - See what you missed / what others found
- Phase 3 now assigned
  - Use *cryptyc* to model a better crypto protocol
  - Scott's tutorial from Monday online later today

Public key primitives

- Diffie-Hellman
  - Hard problem: Discrete logarithms
- RSA
  - Hard problem: Factoring composite numbers

- Field: integers modulo a large prime number (numbers wrap around)

A public-key distribution protocol: Diffie-Hellman

- Before: Alice and Bob have never met and share no secrets; they know a public system parameter α

generate x $\qquad \xrightarrow{\alpha^x} \qquad$ *generate* y

compute α^x $\qquad \xleftarrow{\alpha^y} \qquad$ *compute* α^y

compute $k=(\alpha^y)^x$ $\qquad\qquad$ *compute* $k=(\alpha^x)^y$

- After: Alice and Bob share a short term key k
  - Eve cannot compute $\boldsymbol{k}$: in several mathematical structures it is hard to derive x from α^x (this is known as the discrete logarithm problem)

Diffie-Hellman (continued)

generate x	$\xrightarrow{\alpha^{x}}$	*generate* y
compute α^{x}		*compute* α^{y}
	$\xleftarrow{\alpha^{y}}$	
compute $k=(\alpha^{y})^{x}$		*compute* $k=(\alpha^{x})^{y}$

- BUT: How does Alice know that she shares this secret key ***k*** with Bob?
- Answer: Alice has no idea at all about who the other person is! The same holds for Bob.

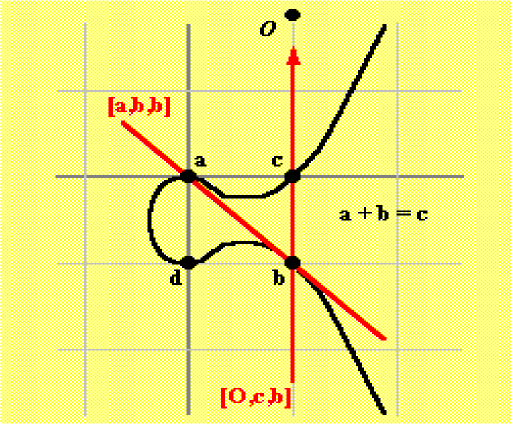
Station to Station protocol (STS)

- The problem can be fixed by adding digital signatures
- Many variations on this theme used in practice

choose x	$\xrightarrow{\alpha^{x}}$	
	$\xleftarrow{\alpha^{y}}$	*choose* y
$k=(\alpha^{y})^{x}$	$\xrightarrow{SigA(\alpha^{x},\alpha^{y})}$	$k=(\alpha^{x})^{y}$
$\surd SigB$	$\xleftarrow{SigB(\alpha^{y},\alpha^{x})}$	$\surd SigA$

Footnote: if you can define multiplication…

- "Elliptic curve" crypto looks the same as Diffie-Hellman
- Instead of integers mod N
  - $y^2 = x^3 + Ax^2 + B \pmod{p}$
  - A, B, p are "carefully chosen"
  - Integers (x,y) on the curve form a group
  - Addition, multiplication, exponentiation can be defined



- Claim: DLog is harder for elliptic curves than modular integer arithmetic
  - Therefore we can use smaller numbers ➔ faster computation

RSA ('78)

- Choose 2 "large" prime numbers p and q
- modulus $n = p.q$
- compute $\lambda(n) = \mathrm{lcm}(p-1,q-1)$
- choose e relatively prime w.r.t. $\lambda(n)$
- compute $d = e^{-1} \mod \lambda(n)$

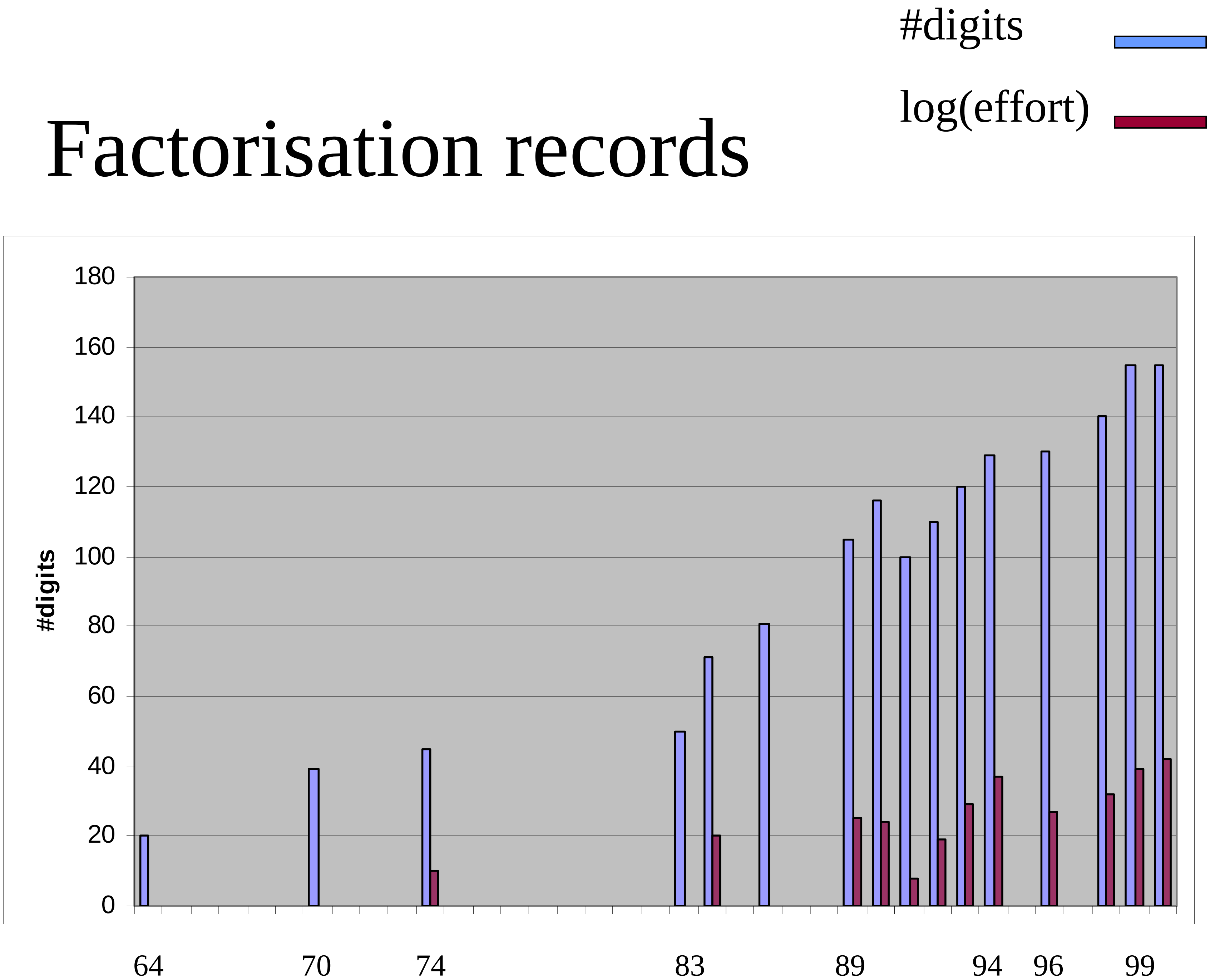
The security of RSA is based on the "fact" that it is easy to generate two large primes, but that it is hard to factor their product

- public key = (e,n)
- private key = (d,p,q)

- encryption: $c = m^e \mod n$
- decryption: $m = c^d \mod n$

try to factor 2419

Factorisation records



What about quantum computers?

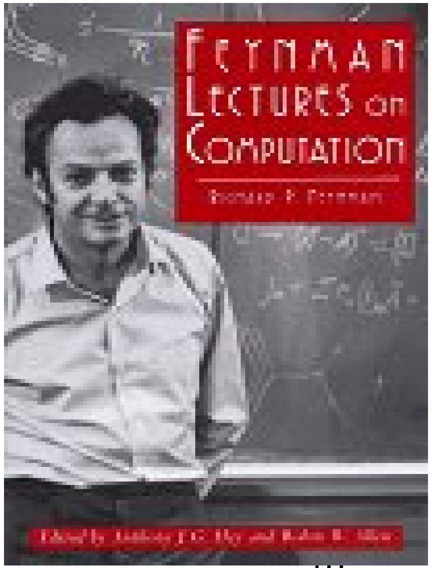
- exponential parallelism

n coupled quantum bits

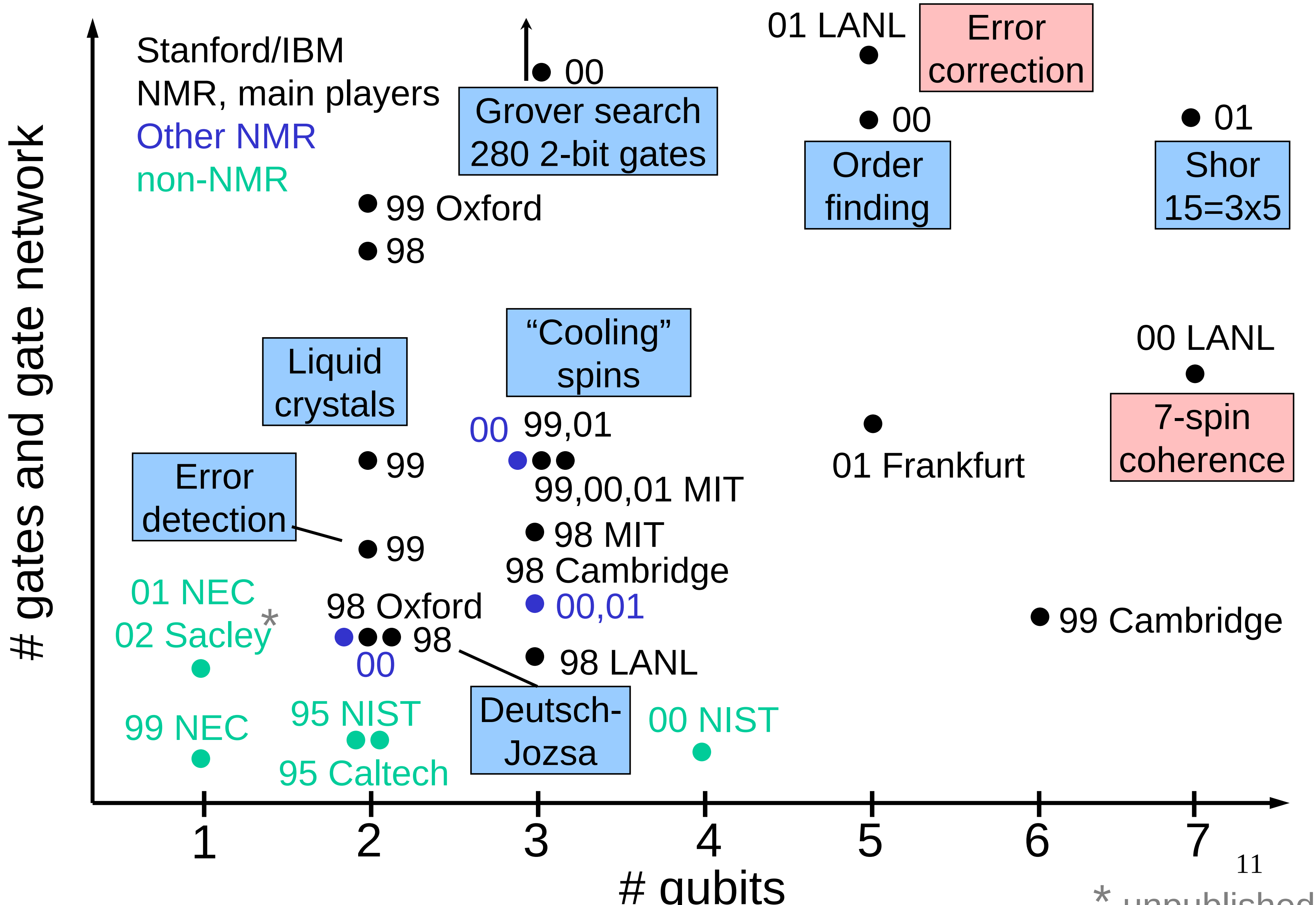
$\Downarrow$

2^{n} degrees of freedom !

- Shor 1994: perfect for factoring
- But: can a quantum computer be built?



State of the art in coherent qubit control



* unpublished

Advantages of public-key cryptology

- Reduce protection of information to protection of authenticity of public keys
- Confidentiality without establishing secret keys
  - extremely useful in an open environment
- Data authentication without shared secret keys: digital signature
  - sender and receiver have different capability
  - third party can resolve dispute between sender and receiver

Disadvantages of public-key cryptology

- Calculations in software or hardware two to three orders of magnitude slower than symmetric algorithms
- Longer keys: 1024 bits rather than 56…128 bits
- What if factoring is easy?

Practical Cryptography:
Provable Security as a Tool for Protocol Design

Phillip Rogaway
UC Davis & Chiang Mai Univ
rogaway@cs.ucdavis.edu
http://www.cs.ucdavis.edu/~rogaway

Summer School on Foundations of Internet Security
17-19 June 2002
Duszniki Zdroj, Poland
(three two-hour lectures)

Slides modified and tweaked by Dan Wallach, with permission

Outline from the paper board

0. Opening comments
1. What is "provably security"?
2. Blocks ciphers
   - 2.1 Syntax
   - 2.2 Notions of security (prp, prf, kr)
3. Symmetric Encryption
   - 3.1 Syntax
   - 3.2 Notions of security (sem, ind, ind$, all under CPA)
4. Relating the notions (ind$, ind, 01)
5. Sample block-cipher-using encryption schemes
6. Security of modes
   - 6.1 CTR-rand
   - 6.2 CBC-rand
7. MACs and authenticated encryption
   - 7.1 Notion of authenticated encryption
   - 7.2 Notion of MACs
   - 7.3 Ways to MAC (CBC, XCBC, CW (w/ poly-based universal hash, UMAC)
   - 7.4 Ways to achieve auth enc (generic composition, IAPM/OCB)

Concluding comments

Recognize Problem

Protocol

Bug

New Protocol

Bug

New Protocol

Publish

Implement

Ship

Bug

“Classical Approach”

"Provable-Security Approach"

begins with [GM82]

Recognize Problem

Definition

Π Protocol

Definition

Protocol π

Proof: reduction

Publish

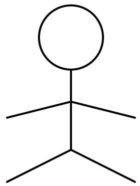
Instantiate

Implement

Ship

Done

Primitive π		**Protocol** Π
Block Cipher	$\longrightarrow$	Sym enc scheme
Block Cipher	$\longrightarrow$	MAC
Block Cipher	$\longrightarrow$	OWF
OWF	$\longrightarrow$	Block Cipher
Sym enc scheme + MAC	$\longrightarrow$	AKE
RSA primitive	$\longrightarrow$	Asym enc scheme



If primitive π is secure **then** protocol Π is secure

If $\nexists$ a good adv for attacking π **then** $\nexists$ no good adv for attacking Π

If $\nexists$ a good adv for attacking Π **then** $\nexists$ a good adv for attacking π

Block-Cipher Syntax

$$E: \boldsymbol{K} \times \{0,1\}^n \rightarrow \{0,1\}^n$$

where each $E_K(\cdot) = E(K, \cdot)$ is a permutation

Eg: $E_K(X)=X$

$E_K(X)=AES128_K(X)$

Notions of Block-Cipher Security

Key-recover (kr) under chosen-plaintext attack (CPA)

$$\mathrm{Adv}^{kr}_{E}(A) = \Pr[\, K \xleftarrow{\$} \boldsymbol{\mathcal{K}} : A^{E(K,\cdot)} = K \,]$$

$$\mathrm{Adv}^{kr}_{E}(t,q) = \max_{A} \{\mathrm{Adv}^{kr}_{E}(A)\}$$

Runs in time $\leq t$
Asks $\leq q$ queries

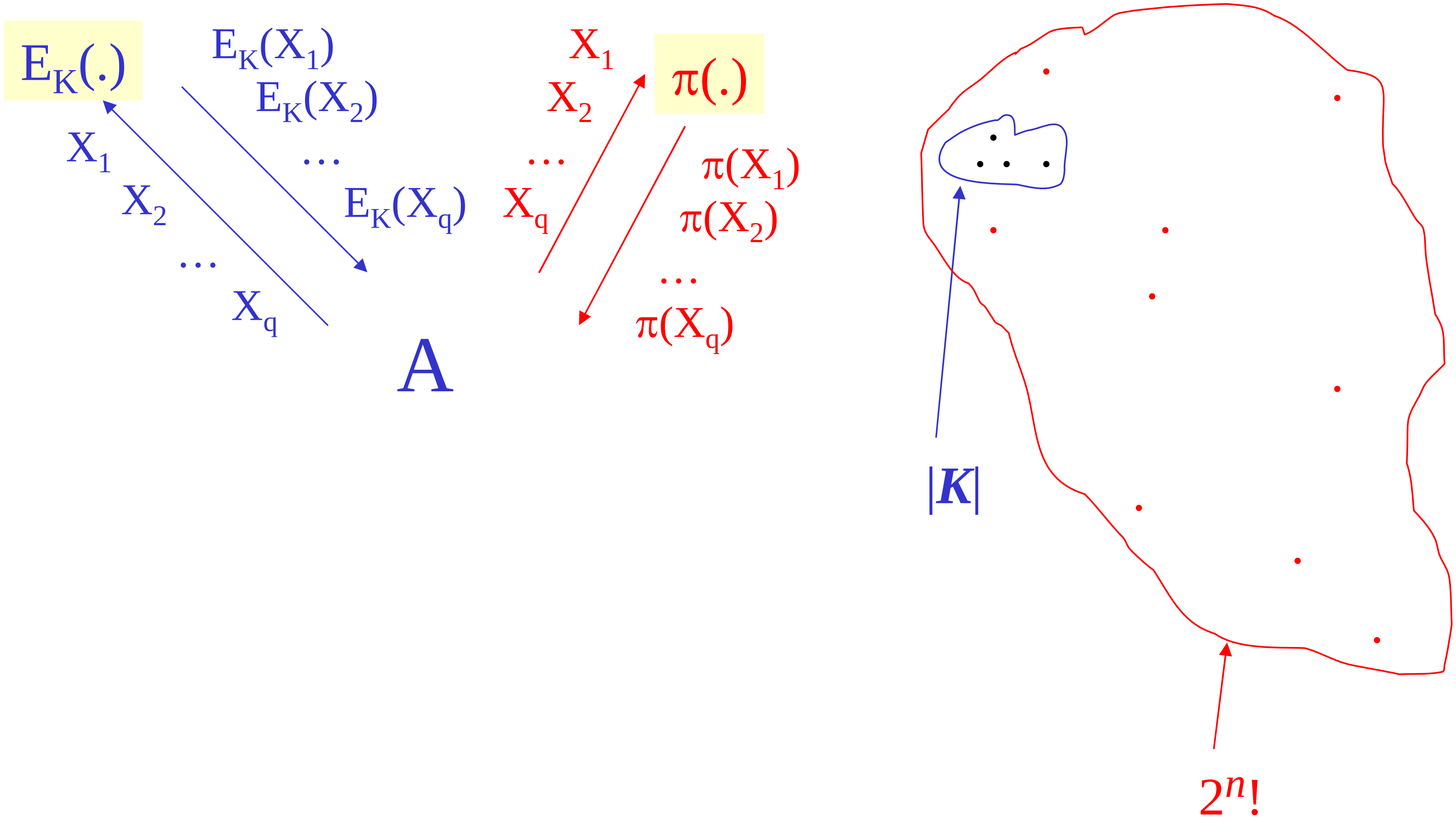
$E_K(.)$

X_1
X_2
…
X_q

$E_K(X_1)$
$E_K(X_2)$
…
$E_K(X_q)$

A

PRP-sense of a block cipher being good



$$\mathrm{Adv}_E^{\mathrm{prp}}(A) = \Pr[\, K \xleftarrow{\$} \boldsymbol{K} : A^{E(K,\cdot)} = 1] -$$

$$\Pr[\, \pi \xleftarrow{\$} \mathrm{Perm}(n) : A^{\pi(\cdot)} = 1]$$

Attacker A responds:

0: it's a permutation

1: it's the cipher

$$\mathrm{Adv}_E^{\mathrm{prp}}(t,q) = \max_A \{\mathrm{Adv}_E^{\mathrm{prp}}(A)\}$$

Runs in time $\leq$ t

Asks $\leq$ q queries

Breaking $E_K(X)=X$

A: Ask 0^n, receiving Y
if $Y=0^n$ return 1 (cipher returns the identity)
else return 0

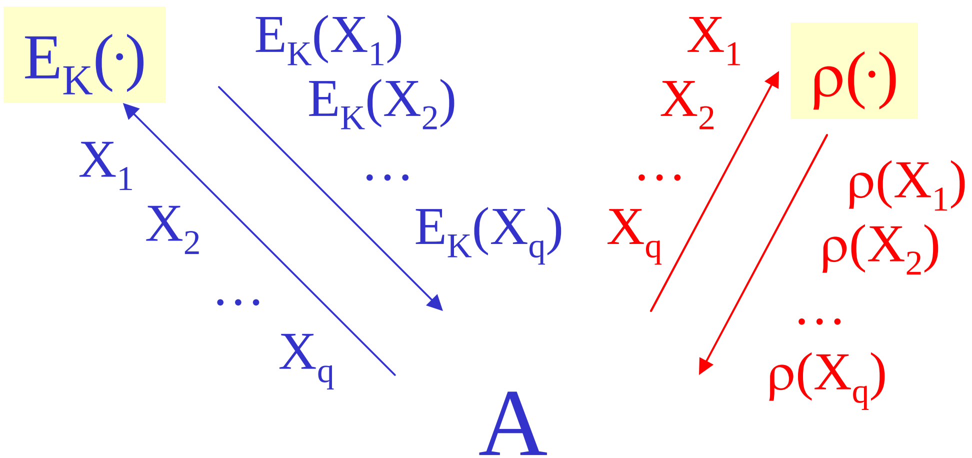
$\mathrm{Adv}_E^{\mathrm{prp}}(A) = 1 - 2^{-n}$ (permutation might also)

$\mathrm{Adv}_{\mathrm{AES}}^{\mathrm{prp}}(t,q) \leq t / 2^{128}$ Strong assumption

$\mathrm{Adv}_{\mathrm{AES}}^{\mathrm{prp}}(t,q) \leq 2^{-40}$ if $t<2^{80}$, $q<2^{40}$ Weaker assumption

$$\mathrm{Adv}^{\mathrm{prf}}_{E}(A) = \Pr[\,K \stackrel{\$}{\leftarrow} \boldsymbol{\mathcal{K}}:\ A^{E(K,\cdot)} = 1] -$$

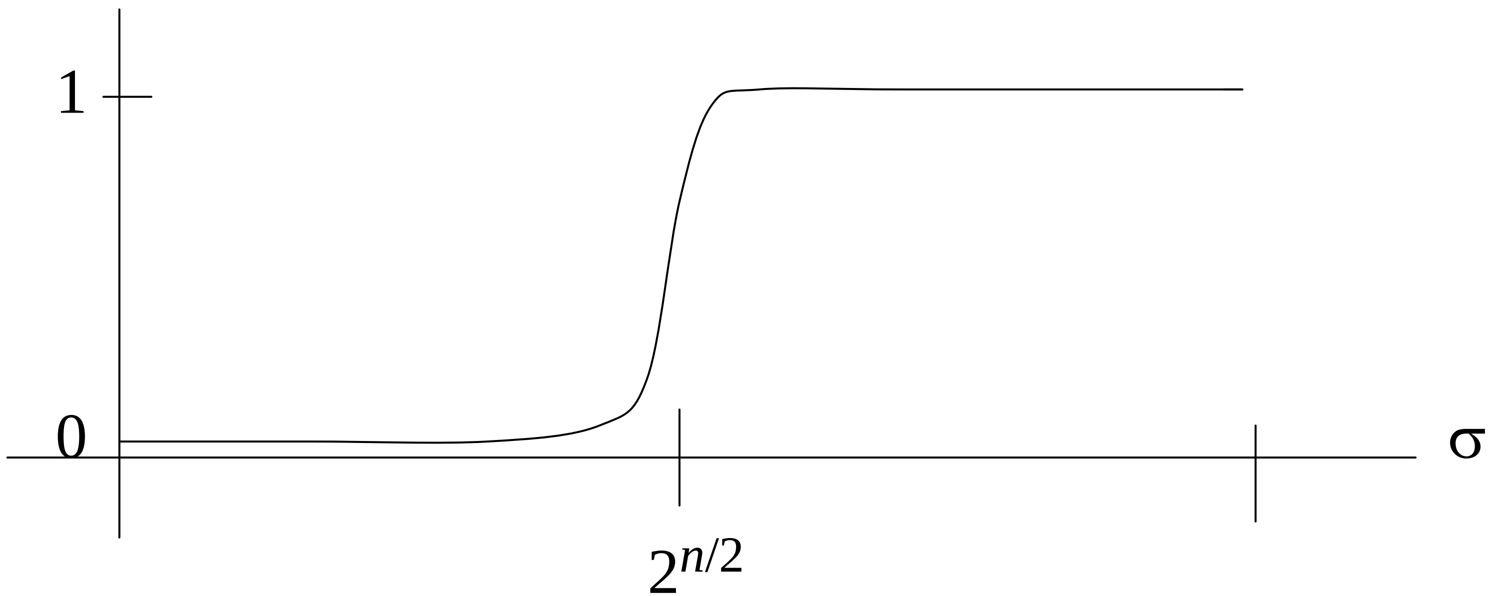
$$\Pr[\,\rho \stackrel{\$}{\leftarrow} \mathrm{Rand}(n):\ A^{\rho(\cdot)} = 1]$$



$$\mathrm{Adv}^{\mathrm{prf}}_{E}(A) = 2\Pr[\,b \stackrel{\$}{\leftarrow} \{0,1\};$$
$$\text{if } b=1 \text{ then } K \stackrel{\$}{\leftarrow} \boldsymbol{\mathcal{K}},\ f=E_K \text{ else } f \stackrel{\$}{\leftarrow} \mathrm{Rand}(n):\ A^{f(\cdot)} = b] - 1$$

"Switching Lemma" If A asks σ queries

$$\left|\mathrm{Adv}_E^{\mathrm{prp}}(A) - \mathrm{Adv}_E^{\mathrm{prf}}(A)\right| \le \sigma^2 / 2^{n+1}$$



$$\Pr[A^{\pi(\cdot)} = 1] - \Pr[A^{\rho(\cdot)} = 1] \le \sigma^2 / \, ^{n+1}$$

Def. A (sym, prob) enc scheme is a 3-tuple

$\Pi = (\boldsymbol{K}, \boldsymbol{E}, \boldsymbol{D})$

Finite set

$\boldsymbol{M} \subseteq \{0,1\}^*$

$\boldsymbol{E}: \boldsymbol{K} \times \boldsymbol{M} \rightarrow \{0,1\}^*$ is a prob. function

If $M \in \boldsymbol{M}$ and $|M'| = |M|$
then $M' \in \boldsymbol{M}$

$\boldsymbol{D}: \boldsymbol{K} \times \{0,1\}^* \rightarrow \boldsymbol{M} \cup \{*\}$ (det funct)

$$M \in \boldsymbol{M},\ K \in \boldsymbol{K},\ C \xleftarrow{\$} \boldsymbol{E}_K(M) \Rightarrow \boldsymbol{D}_K(C) = M$$

$|C| = \text{clen}(\,|M|\,)$

CPA

$E_K(.)$

M_1
M_2
…
M_q

$E_K(X_1)$
$E_K(X_2)$
…
$E_K(X_q)$

A

support($\boldsymbol{M}$) only has strings of one length

$$\Pi = (\boldsymbol{K}, \boldsymbol{E}, \boldsymbol{D})$$

sem

$$\mathrm{Adv}^{\mathrm{sem}}_{\Pi}(A) = \Pr\left[\, K \xleftarrow{\$} \boldsymbol{K};\ (f, \boldsymbol{M}) \xleftarrow{\$} A^{\boldsymbol{E}(K,\cdot)}(\,);\ M \xleftarrow{\$} \boldsymbol{M};\ C \xleftarrow{\$} \boldsymbol{E}_K(M) : A^{\boldsymbol{E}(K,\cdot)}(C, f) = f(M) \right] -$$

$$\Pr\left[\, K \xleftarrow{\$} \boldsymbol{K};\ (f, \boldsymbol{M}) \xleftarrow{\$} A^{\boldsymbol{E}(K,\cdot)}(\,);\ M, M' \xleftarrow{\$} \boldsymbol{M};\ C \xleftarrow{\$} \boldsymbol{E}_K(M') : A^{\boldsymbol{E}(K,\cdot)}(C, f) = f(M) \right]$$