


## Cryptographic algorithms


Prof. Bart Preneel  
COSIC  
Bart.Preneel(at)esatDOTkuleuven.be  
<http://homes.esat.kuleuven.be/~preneel>

© Bart Preneel. All rights reserved



## Outline

- 1. Cryptology: concepts and algorithms
  - symmetric algorithms for confidentiality
  - symmetric algorithms for data authentication
  - public-key cryptology
- 2. Cryptology: protocols
  - identification/entity authentication
  - key establishment
- 3. Public-Key Infrastructure fundamentals



## Outline (2)

- 4. Network security protocols
  - web (SSL/TLS) and IPsec
- 5. Cryptography best practices
- 6. Recent developments in cryptology

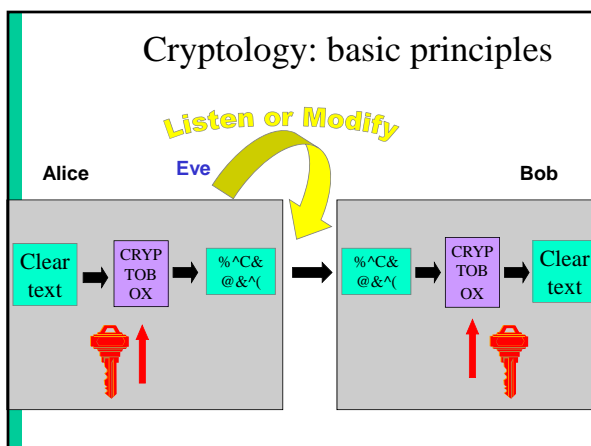
## Definitions

	<b>data</b>	<b>entities</b>	
Confidentiality	confidentiality	encryption	anonymity
Integrity	authentication	data authentication	identification
Availability			

- Authorisation
- Non-repudiation of origin, receipt
- Contract signing
- Notarisation and Timestamping

Don't use the word authentication without defining it

4




## Symmetric cryptology: confidentiality

- old cipher systems:
  - transposition, substitution, rotor machines
- the opponent and her power
- the Vernam scheme
- DES and triple-DES
- AES
- RC4

### Old cipher systems (pre 1900)

- Caesar cipher: shift letters over  $k$  positions in the alphabet ( $k$  is the secret key)

THIS IS THE CAESAR CIPHER  
WKLV LV WKH FDHYDU FLSKHU



- Julius Caesar never changed his key ( $k=3$ ).

7

### Cryptanalysis example:

TIPGK RERCP JZJZJ WLE	GVCTX EREPC WMMWV JYR
UJQHL SFSDQ KAKAK XMF	HWDUY FSPQD XNXXN KZS
VKRIM TGTER LBLBL YNG	IXEVZ GTGRE YOYOY LAT
WLSJN UHUFV MCMCM ZOH	JYFWA HUHSF ZPZPZ MBU
XDTKO VOVGT NDNDN API	KZGXB IVITG AQAQA NCV
YNULP WKWHU OEOEO BQJ	LAHYC JWJUH BRBRB ODW
ZOVMQ KXKIV PFPFP CRK	MBIZD KXKVI CSCSC PEX
APWNR YLYJW QQQQQ DSL	NCJAE LYLWJ DTDTD QFY
BQXOS ZMXXK RHRHR ETM	ODKBF MZMXX EUEUE RGZ
<u>CRYPT ANALY SISIS FUN</u>	PELCG NANYL FVVFV SHA
DSZQU BOBMZ TJTJT GVO	QFMDH OBOZM GWGWG TIB
ETARV CPCNA UKUKU HWP	RGNEI PCPAN HXHXH UJC
FUBSW DQDOB VLVLV IXQ	SHOFJ QDQBO IYIYI VKD

Plaintext?  $k = 17$

8

### Old cipher systems (pre 1900) (2)

- Substitutions
  - ABCDEF...IJKLMNOPQRSTUVWXYZ
  - MZNTSOAXFQGYKHLUCTDVBWIPER
- Transpositions
  - TRANS      ORI S
  - POSIT      NOTIT
  - IONS       OSANP

! Easy to break using statistical techniques

9

### Security

- there are  $n!$  different substitutions on an alphabet with  $n$  letters
- there are  $n!$  different transpositions of  $n$  letters
- $n=26$ :  $n! = 403291461126605635584000000 = 4 \cdot 10^{26}$  keys
- trying all possibilities at 1 nanosecond per key requires....

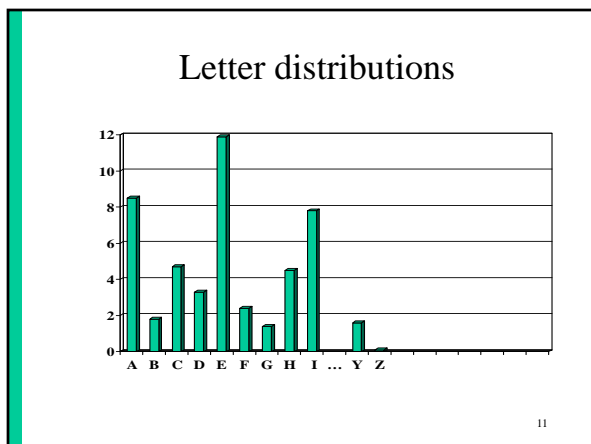
$$4 \cdot 10^{26} / (10^9 \cdot 10^5 \cdot 4 \cdot 10^2) = 10^{10} \text{ years}$$

keys per second

seconds per day

days per year

10



### Assumptions on Eve (the opponent)

- A scheme is broken if Eve can deduce the key or obtain additional plaintext
- Eve can always **try all keys** till "meaningful" plaintext appears: a brute force attack
  - solution: large key space
- Eve will try to find **shortcut attacks** (faster than brute force)
  - history shows that designers are too optimistic about the security of their cryptosystems

12

### Assumptions on Eve (the opponent)

- Cryptology = cryptography + cryptanalysis
- Eve knows the algorithm, except for the key (Kerckhoffs's principle)
- increasing capability of Eve:
  - knows some information about the plaintext (e.g., in English)
  - knows part of the plaintext
  - can choose (part of) the plaintext and look at the ciphertext
  - can choose (part of) the ciphertext and look at the plaintext



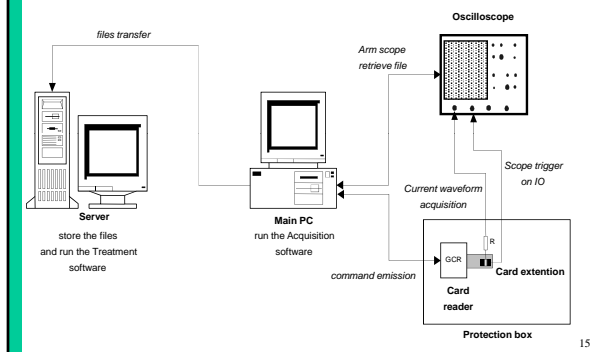
13

### New assumptions on Eve

- Eve may have access to **side channels**
  - timing attacks
  - simple power analysis
  - differential power analysis
  - acoustic attacks
  - electromagnetic interference
- Eve may launch **(semi-)invasive attacks**
  - differential fault analysis
  - probing of memory or bus

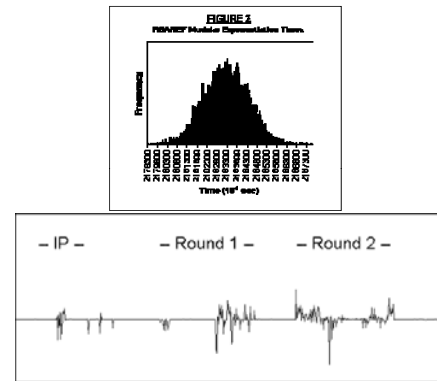
14

### Side channel analysis



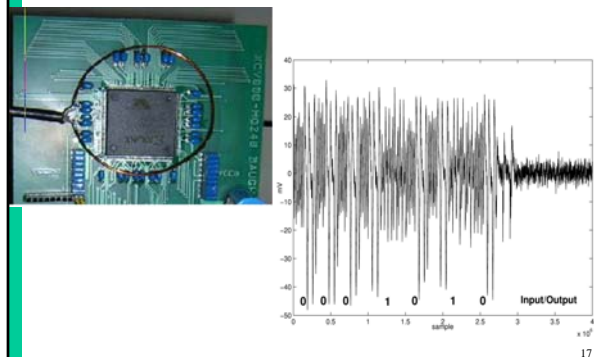
15

### Timing attacks and power analysis



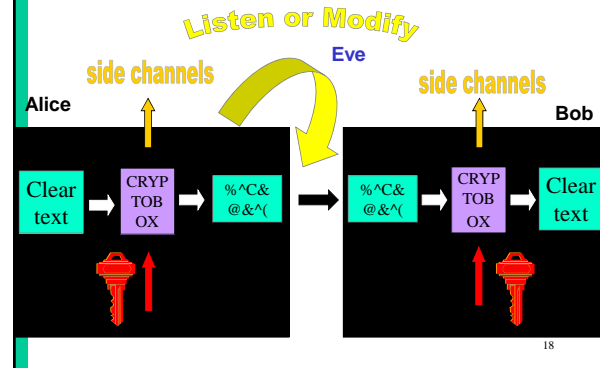
16

### Side channel analysis: EMA



17

### Cryptology + side channels



18



### Problem: what is this?

- Cryptogram [=14 January 1961 11.00 h]
- <AHQNE XVAZW IQFFR JENFV OUXBD  
LQWDB BXFRZ NJVYB QVGOZ KFYQV  
GEDBE HGMP5 GAZJK RDJQC VJTEB  
XNZZH MEVGS ANLLB DQCGF PWCVR  
UOMWW LOGSO ZWVVV LDQNI YTZAA  
OIJDR UEAAV RWYXH PAWSV CHTYN  
HSUIY PKFPZ OSEAW SUZMY QDYEL  
FUVOA WLSSD ZVKPU ZSHKK PALWB  
SHXRR MLQOK AHQNE 11205  
141100>

20

### The answer

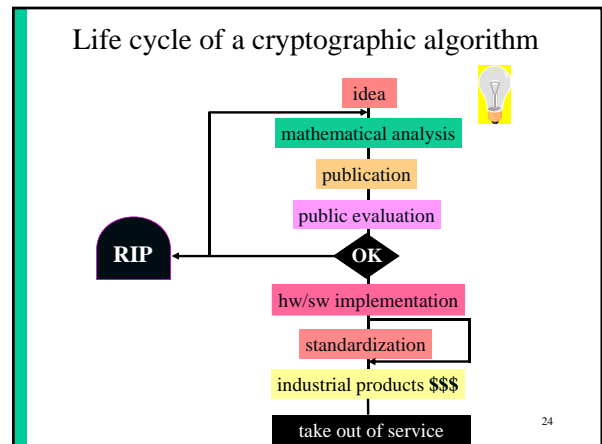
- Plaintext [=14 January 1961 11.00 h]
- DOFGD VISWA WVISW JOSEP HWXXW  
TERTI OWMIS SIONW BOMBO KOWVO  
IRWTE LEXWC EWSUJ ETWAM BABEL  
GEWXX WJULE SWXXW BISEC TWTRE  
SECVX XWRWV WMWPR INTEX WXXWP  
RIMOW RIENW ENVOY EWRUS URWWX  
XWPOU VEZWR EGLER WXXWS ECUND  
OWREP RENDR EWDUR GENGE WPLAN  
WBRAZ ZAWWC

21

### The answer (in readable form)

- Plaintext [=14 January 1961 11.00 h]
- TRESECV. R V M PRINTEX. PRIMO  
RIEN ENVOYE RUSUR. POUVEZ  
REGLER. SECUNDO REPREDRE  
DURGENCE PLAN BRAZZA VIS A VIS  
JOSEP H. TERTIO MISSION  
BOMBOKO VOIR TELEX CE SUJET  
AMBABELGE. JULES.

22



Vernam scheme (1917)  
Mauorgne: one time pad (1917+x)

Shannon (1948)

F. Miller (1882)

key is random string, as long as the plaintext  
information theoretic proof of security

### Vernam scheme

- $0 + 1 = 1$
- $1 + 0 = 1$
- $0 + 0 = 0$
- $1 + 1 = 0$

- identical mathematical symbols can result in different electrical signals

26

### Three approaches in cryptography

- **information theoretic** security
  - ciphertext only
  - part of ciphertext only
  - noisy version of ciphertext
- **system-based** or practical security
  - also known as “prayer theoretic” security
- **complexity theoretic** security:
  - model of computation, definition, proof
  - variant: quantum cryptography

27

### Synchronous Stream Cipher (SSC)

### A5/1 stream cipher (GSM)

Clock control: registers agreeing with majority are clocked (2 or 3)

29

### A5/1 stream cipher (GSM)

- exhaustive key search:  $2^{64}$  (or rather  $2^{54}$ )
  - hardware 10K\$ < 1 minute ciphertext only
- search 2 smallest registers:  $2^{45}$  steps
- [BWS00] 1 minute on a PC
  - 2 seconds of known plaintext
  - $2^{48}$  precomputation, 146 GB storage
- [BB05]: 10 minutes on a PC,
  - 3-4 minutes of **ciphertext only**
- [Nohl-Paget'09]: rainbow tables
  - a few frames of **ciphertext only**

30

### Bluetooth stream cipher

brute force:  $2^{128}$  steps  
[Lu+05] 24 known bits of  $2^{24}$  frames,  $2^{38}$  computations,  $2^{33}$  memory

31

### A simple cipher: RC4 (1987)

- designed by Ron Rivest (MIT)
- leaked in 1994
- $S[0..255]$ : secret table derived from user key K

```

for i=0 to 255 S[i]:=i
j:=0
for i=0 to 255
    j:=(j + S[i] + K[i]) mod 256
    swap S[i] and S[j]
i:=0, j:=0
    
```

32

### A simple cipher: RC4 (1987)

Generate key stream which is added to plaintext

```

i:=i+1
j:=(j + S[i]) mod 256
swap S[i] and S[j]
t:=(S[i] + S[j]) mod 256
output S[t]
    
```

000	001	002		093	094	095		254	255
205	162	013	...	033	92	079	...	099	143

i ↑    j ↑    t ↓

33

### RC4: weaknesses

- often used with 40-bit key
  - US export restrictions until Q4/2000
- best known general shortcut attack:  $2^{241}$
- weak keys and key setup (shuffle theory)
- some statistical deviations
  - e.g., 2nd output byte is biased
  - solution: drop first 256 bytes of output
- problem with resynchronization modes (WEP)

34

### Block cipher

- large table: list n-bit ciphertext for each n-bit plaintext
  - if n is large: very secure (codebook)
  - but for an n-bit block:  $2^n$  values
  - impractical if  $n \geq 32$
- alternative  $n = 64$  or  $128$ 
  - simplify the implementation
  - repeat many simple operations

35

### Block cipher (2)

- larger data units: 64...128 bits
- memoryless
- repeat simple operation (round) many times

36

### Exhaustive key search

- 2013:  $2^{40}$  instructions is easy,  $2^{60}$  is somewhat hard,  $2^{80}$  is hard,  $2^{128}$  is completely infeasible
  - 1 million machines with 16 cores and a clock speed of 4 GHz can do  $2^{56}$  instructions per second or  $2^{80}$  per year
  - trying 1 key requires typically a few 100 instructions
- Moore's "law": speed of computers doubles every 18 months: key lengths need to grow in time
  - but adding 1 key bit doubles the work for the attacker
- Key length recommendations in 2013
  - < 70 bits: insecure
  - 80 bits: a few years
  - 100 bits: 20-25 years

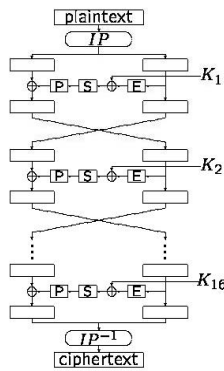
37

### Data Encryption Standard (1977)

- encrypts 64 plaintext bits under control of a 56-bit key
- 16 iterations of a relatively simple mapping
- FIPS: US government standard for sensitive but unclassified data
- worldwide de facto standard since early 80ies
- surrounded by controversy

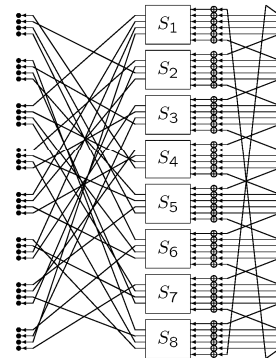
38

### Data Encryption Standard (DES)

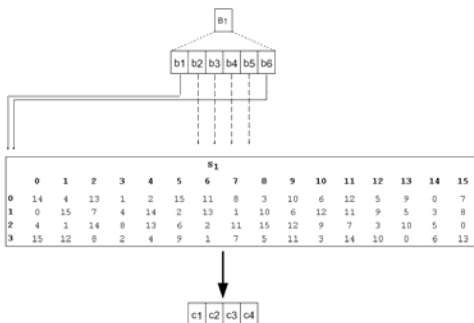


39

### The DES round function



### DES S-box 1

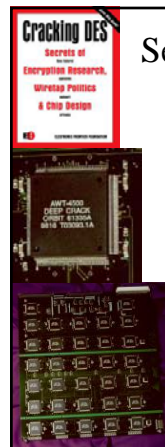


www.gungfu.de

### Security of DES (56 bit key)

- PC: trying 1 DES key: 7.5 ns
- Trying all keys on 128 PCs: 1 month:  $2^{27} \times 2^{16} \times 2^5 \times 2^7 = 2^{55}$
- M. Wiener's design (1993): 1,000,000 \$ machine: 3 hours (in 2012: 3 seconds)

EFF Deep Crack (July 1998)  
250,000 \$ machine: 50 hours...



42

### DES: security (ct'd)

- Moore's "law": speed of computers doubles every 18 months
  - key lengths need to grow in time
- Use new algorithms with longer keys
  - adding 1 key bits doubles the work for the attacker
- Key length recommendations in 2012
  - < 64 bits: insecure
  - 80 bits: 1-2 years
  - 100 bits: 18-22 years

43

### Federal Register, July 24, 2004

**DEPARTMENT OF COMMERCE** • **SUMMARY:** The Data Encryption Standard (DES), currently specified in Federal Information Processing Standard (FIPS) 46-3, was evaluated pursuant to its scheduled review. At the conclusion of this review, **NIST determined that the strength of the DES algorithm is no longer sufficient to adequately protect Federal government information.** As a result, NIST proposes to withdraw FIPS 46-3, and the associated FIPS 74 and FIPS 81. Future use of DES by Federal agencies is to be permitted only as a component function of the Triple Data Encryption Algorithm (TDEA).

**National Institute of Standards and Technology**  
[Docket No. 040602169- 4169- 01]

**Announcing Proposed Withdrawal of Federal Information Processing Standard (FIPS) for the Data Encryption Standard (DES) and Request for Comments**

**AGENCY:** National Institute of Standards and Technology (NIST), Commerce.

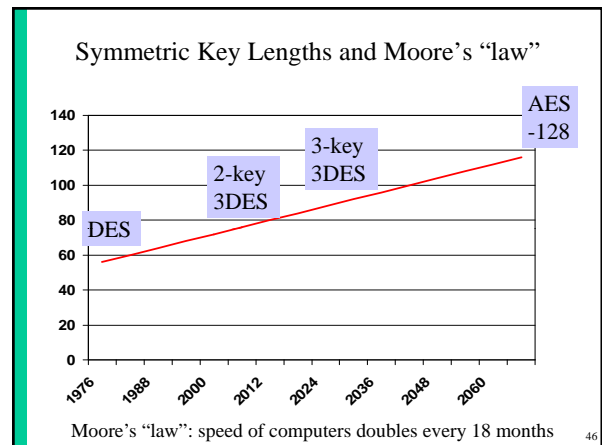
**ACTION:** Notice; request for comments.

44

### 3-DES: NIST Spec. Pub. 800-67 (May 2004)

- two-key triple DES: until 2009
- three-key triple DES: until 2030

45



### AES (Advanced Encryption Standard)

- open competition launched by US government (Sept. '97) to replace DES
- 22 contenders including IBM, RSA, Deutsche Telekom
- 128-bit block cipher with key of 128/192/256 bits
- as strong as triple-DES, but more efficient
- royalty-free

A machine that cracks a DES key in 1 second would take 149 trillion years to crack a 128-bit key

47

### AES: Rijndael

- Key length: 16/24/32 bytes
- Block length:
  - Rijndael: 16/24/32 bytes
  - AES: 16 bytes only

48



## AES (2001)

- FIPS 197 published on December 2001 after 4-year open competition
  - other standards: ISO, IETF, IEEE 802.11,...
- fast adoption in the market
  - except for financial sector
  - NIST validation list: > 2300 implementations
    - <http://csrc.nist.gov/groups/STM/cavp/documents/aes/aesval.html>
- 2003: AES-128 also for **classified** information and AES-192/-256 for **secret** and **top secret** information!

49

## AES (2001)

- **security:**
  - algebraic attacks of [Courtois+02] not effective
  - side channel attacks: cache attacks on **unprotected** implementations
- **speed:**
  - software: 7.6 cycles/byte [Käsper-Schwabe'09]
  - hardware: Intel provides AES instruction (Westmere/Sandy Bridge, 2010/2011) at 0.75 cycles/byte for decryption – AMD one year behind

[Shamir '07] AES may well be the last block cipher

50

## Encryption limitations

- Ciphertext becomes random string: “normal” crypto does not encrypt a credit card number into a (valid) credit card number
- Typically does not hide the length of the plaintext (unless randomized padding)
- Does **not** hide existence of plaintext (requires steganography)
- Does **not** hide that Alice is talking to Bob (requires traffic confidentiality)

## Symmetric cryptology: data authentication

- the problem
- hash functions without a key
  - MDC: Manipulation Detection Codes
- hash functions with a secret key
  - MAC: Message Authentication Codes

52

## Data authentication: the problem

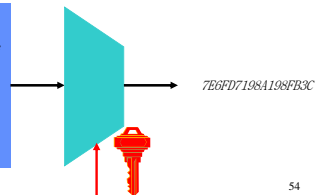
- encryption provides confidentiality:
  - prevents Eve from learning information on the cleartext/plaintext
  - but does not protect against modifications (active eavesdropping)
- Bob wants to know:
  - the **source** of the information (data origin)
  - that the information has not been **modified**
  - (optionally) **timeliness** and **sequence**
- data authentication is typically more complex than data confidentiality

53

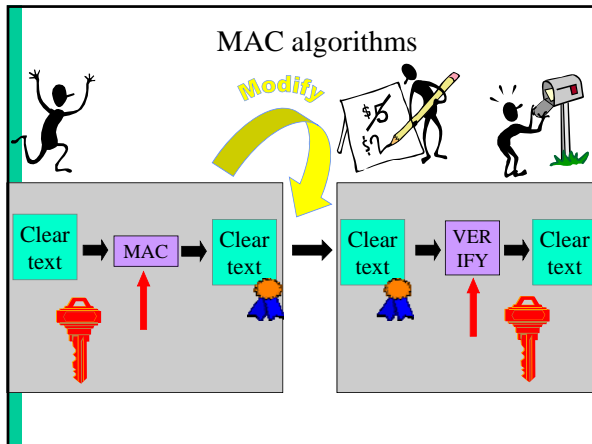
## Data authentication: MAC algorithms

- Replace protection of authenticity of (long) message by protection of secrecy of (short) key
  - Add MAC to the plaintext
- CBC-MAC (CMAC)
  - HMAC
  - GMAC

This is an input to a MAC algorithm. The input is a very long string, that is reduced by the hash function to a string of fixed length. There are additional security conditions: it should be very hard for someone who does not know the secret key to compute the hash function on a new input.



54



**Data authentication: MAC algorithms**

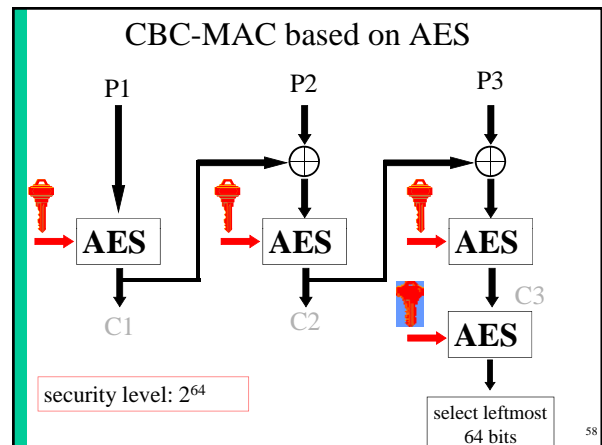
- typical MAC lengths: 32..96 bits
  - Forgery attacks:  $2^m$  steps with  $m$  the MAC length in bits
- typical key lengths: (56)..112..160 bits
  - Exhaustive key search:  $2^k$  steps with  $k$  the key length in bits
- birthday attacks: security level smaller than expected

56

**MAC algorithms**

- Banking: CBC-MAC based on triple-DES
- Internet: HMAC and CBC-MAC based on AES
- information theoretic secure MAC algorithms (authentication codes): GMAC/UMAC
  - highly efficient
  - rather long keys (some)
  - part of the key refreshed per message

57



**Data authentication: MDC**

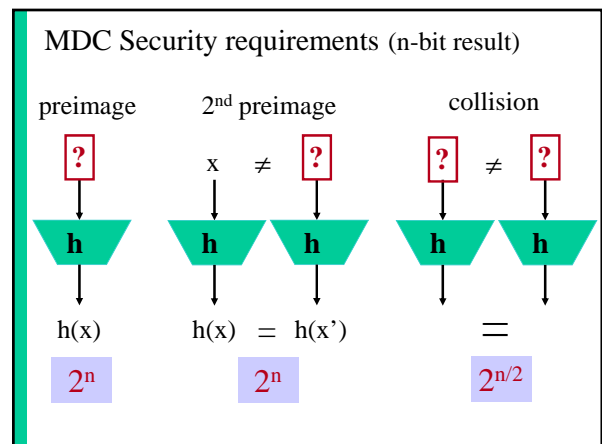
- MDC (manipulation detection code)
- Protect short hash value rather than long text

*This is an input to a cryptographic hash function. The input is a very long string, that is reduced by the hash function to a string of fixed length. There are additional security conditions: it should be very hard to find an input hashing to a given value (a preimage) or to find two colliding inputs (a collision).*

1A3FD4128A108F83CA345032

- (MD5)
- (SHA-1), SHA-256, SHA-512
- RIPEMD-160
- SHA-3

59



### Data authentication: MDC

- n-bit result
- preimage resistance: for given  $y$ , hard to find input  $x$  such that  $h(x) = y$  ( $2^n$  operations)
- 2<sup>nd</sup> preimage resistance: hard to find  $x' \neq x$  such that  $h(x') = h(x)$  ( $2^n$  operations)
- Collision resistance: hard to find  $(x, x')$  with  $x' \neq x$  such that  $h(x') = h(x)$  ( $2^{n/2}$  operations)

61

### MD5 and SHA-1

- SHA-1:
  - (2<sup>nd</sup>) preimage  $2^{160}$  steps
  - collisions  $2^{80}$  steps
- MD5
  - (2<sup>nd</sup>) preimage  $2^{128}$  steps (improved to  $2^{123}$  steps)
  - collisions  $2^{64}$  steps

100 M\$ for 1 year in '05  
Shortcut: Aug. '05:  $2^{69}$  steps  
20 K\$ for 1 month in '05  
Shortcut: Aug. '04:  $2^{39}$  steps; '09:  $2^{20}$  steps


### Public-key cryptology

- the problem
- public-key encryption
- digital signatures
- an example: RSA
- advantages of public-key cryptology

63

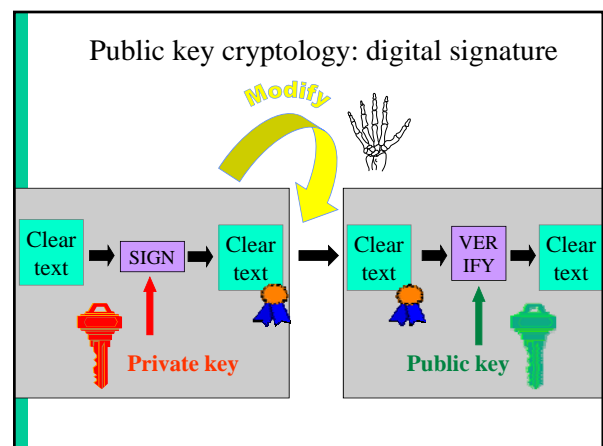
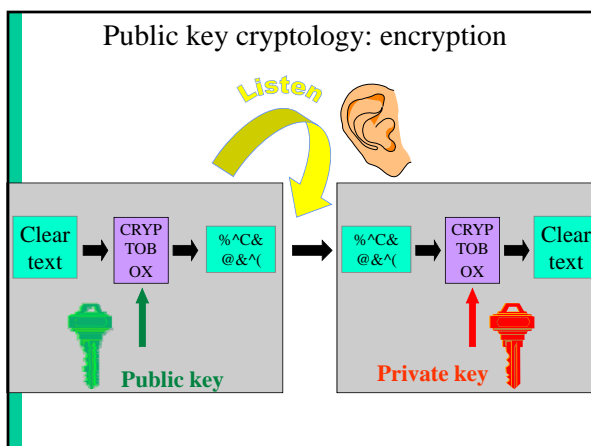
### Limitation of symmetric cryptology

- Reduce security of information to security of keys



- But: how to establish these secret keys?
  - Cumbersome and expensive
  - Or risky: all keys in 1 place
- Do we really need to establish secret keys?

64



A public-key distribution protocol: Diffie-Hellman

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

$$\begin{array}{ccc} \text{generate } x & \xrightarrow{\alpha^x} & \text{generate } y \\ \text{compute } \alpha^x & & \text{compute } \alpha^y \\ & \xleftarrow{\alpha^y} & \\ \text{compute } k=(\alpha^y)^x & & \text{compute } k=(\alpha^x)^y \end{array}$$

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

67

### RSA ('78)

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

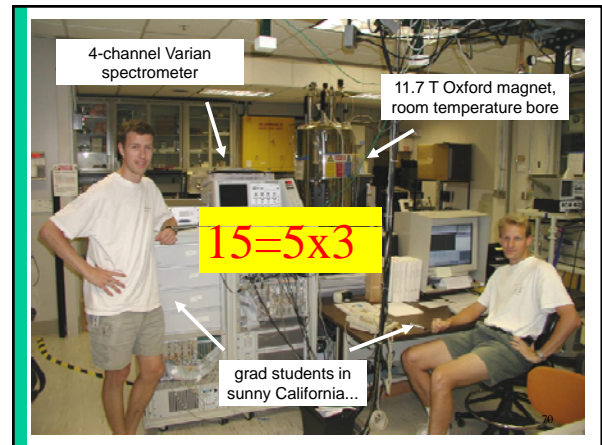
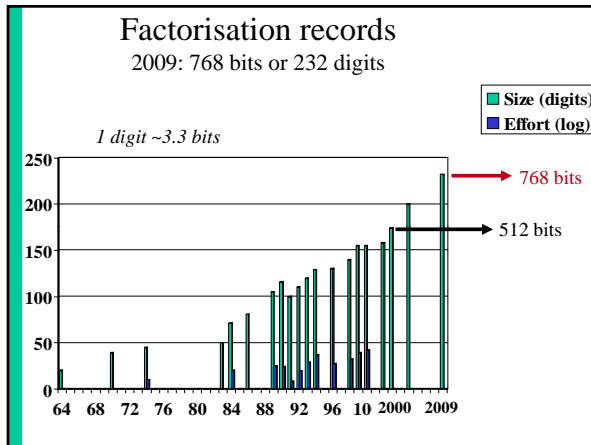
The security of RSA is based on the "fact" that it is easy to generate two large primes, but that it is hard to factor their product

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

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

try to factor 2419

68



- 2001: 7-bit quantum computer factors 15
- 2007: two new 7-bit quantum computers
- 2012: 143 has been factored in Apr. '12
- 2012: 10 to 15 years for a large quantum computer

### Quantum Computing: An IBM Perspective

Steffen, M.; DiVincenzo, D. P.; Chow, J. M.; Theis, T. N.; Ketchen, M. B.

Quantum physics provides an intriguing basis for achieving computational power to address certain categories of mathematical problems that are completely intractable with machine computation as we know it today. We present a brief overview of the current theoretical and experimental works in the emerging field of quantum computing. The implementation of a functioning quantum computer poses tremendous scientific and technological challenges, but current rates of progress suggest that these challenges will be substantially addressed over the next ten years. We provide a sketch of a quantum computing system based on superconducting circuits, which are the current focus of our research. A realistic vision emerges concerning the form of a future scalable fault-tolerant quantum computer.

71

### Advantages of public key cryptography

- 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

72

### 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?

73

### Crypto software libraries

[http://ece.gmu.edu/crypto\\_resources/web\\_resources/libraries.htm](http://ece.gmu.edu/crypto_resources/web_resources/libraries.htm)

#### C/C++/C#

- Botan (C++)
- Cryptlib
- Crypto++ (C++)
- Libgcrypt (C++)
- MatrixSSL (C++) embedded
- Miracl (binaries)
- OpenSSL (C++)
  
- BouncyCastle (BC#)

#### Java

- SunJCA/JCE
- BouncyCastle (BC)
- CryptixCrypto (until '05)
- EspreSSL
- FlexiProvider
- GNU Crypto
- IAIK
- Java SSL
- RSA JSafe

### Reading material

- B. Preneel, Modern cryptology: an introduction.
  - This text corresponds more or less to the second half of these slides
  - It covers in more detail how block ciphers are used in practice, and explains how DES works.
  - It does not cover identification, key management and application to network security.

75

### Selected books on cryptology

- D. Stinson, *Cryptography: Theory and Practice*, CRC Press, 3<sup>rd</sup> Ed., 2005. Solid introduction, but only for the mathematically inclined.
- A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997. The bible of modern cryptography. Thorough and complete reference work – not suited as a first text book. Freely available at <http://www.cacr.math.uwaterloo.ca/hac>
- N. Smart, *Cryptography, An Introduction: 3<sup>rd</sup> Ed.*, 2008. Solid and up to date but on the mathematical side. Freely available at [http://www.cs.bris.ac.uk/~nigel/Crypto\\_Book/](http://www.cs.bris.ac.uk/~nigel/Crypto_Book/)
- B. Schneier, *Applied Cryptography*, Wiley, 1996. Widely popular and very accessible – make sure you get the errata, online
- Other authors: Johannes Buchmann, Serge Vaudenay

76

### Books on network security and more

- W. Stallings, *Network and Internetwork Security: Principles and Practice*, Prentice Hall, 5<sup>th</sup> Ed., 2010. Solid background on network security. Explains basic concepts of cryptography.
- W. Diffie, S. Landau, *Privacy on the line. The politics of wiretapping and encryption*, MIT Press, 2<sup>nd</sup> Ed., 2007. The best book so far on the intricate politics of the field.
- Ross Anderson, *Security Engineering*, Wiley, 2<sup>nd</sup> Ed., 2008. Insightful. A must read for every information security practitioner. Available for free at <http://www.cl.cam.ac.uk/~rja14/book.html>
- Jay Ramachandran, *Designing Security Architecture Solutions*, Wiley 2002.
- Gary McGraw, *Software Security: Building Security In*, Addison Wesley 2006.

### More information: some links

- IACR (International Association for Cryptologic Research): [www.iacr.org](http://www.iacr.org)
- IETF web site: [www.ietf.org](http://www.ietf.org)
- Cryptography faq: [www.faqs.org/faqs/cryptography-faq](http://www.faqs.org/faqs/cryptography-faq)
- Counterpane links: [www.counterpane.com/hotlist.html](http://www.counterpane.com/hotlist.html)
- Digicrime ([www.digicrime.org](http://www.digicrime.org)) - not serious but informative and entertaining

78