KU LEUVEN COSIC

Introduction to Cryptography

Bart Preneel
COSIC-KU Leuven
firstname.lastname(AT)esat.kuleuven.be
@bpreneel1 preneel@infosec.exchange
September 2023

KU LEUVEN

1

Definitions

1970s

Confidentiality
Integrity
Availability

	data	entities
confidentiality	encryption	anonymity
authentication	data authentication	"identification"

Don't use the word authentication without defining it

Authorisation

Non-repudiation of origin, receipt

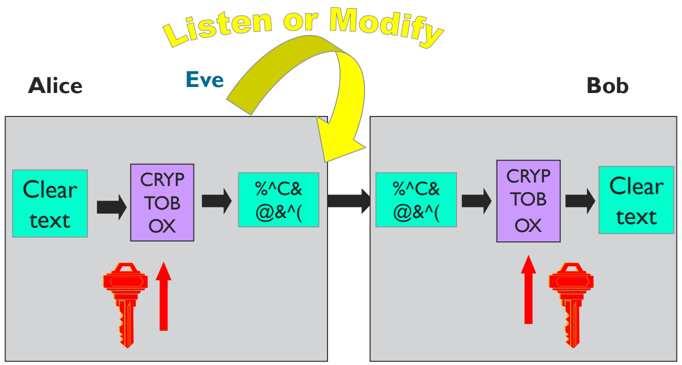
Contract signing

Notarisation and Timestamping

2 KU LEUVEN

2

Cryptology: principle



3 KU LEUVEN

3

Outline

- Symmetric cryptology
  - confidentiality
  - data authentication
  - authenticated encryption
- Public key cryptology (asymmetric cryptology)
- Hybrid cryptology
- Distributing public keys
- Applications

4 KU LEUVEN

4

Symmetric cryptology: confidentiality

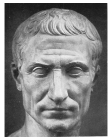
- Old cipher systems:
  - transposition, substitution
- Opponent and her power
- One time pad
- Stream ciphers
- Block ciphers
- Authenticated encryption

5

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 FDHVDU FLSKHU



Julius Caesar never changed his key (k=3)

6

Cryptanalysis example:

```
TIPGK RERCP JZJZJ WLE     GVCTX EREPC WMWMW JYR
UJQHL SFSDQ KAKAK XMF     HWDUY FSFQD XNXNX KZS
VKRIM TGTER LBLBL YNG     IXEVZ GTGRE YOYOY LAT
WLSJN UHUFS MCMCM ZOH     JYFWA HUHSF ZPZPZ MBU
XDTKO VOVGT NDNDN API     KZGXB IVITG AQAQA NCV
YNULP WKWHU OEOEO BQJ     LAHYC JWJUH BRBRB ODW
ZOVMQ XKXIV PFPFP CRK     MBIZD KXKVI CSCSC PEX
APWNR YLYJW QGQGQ DSL     NCJAE LYLWJ DTDTD QFY
BQXOS ZMXKX RHRHR ETM     ODKBF MZMXK EUEUE RGZ
CRYPT ANALY SISIS FUN     PELCG NANYL FVFVF 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

7

Old cipher systems (pre 1900) (2)

- Substitutions

ABCDEFGHIJKLMNOPQRSTUVWXYZ

MZNJSOAXFQGYKHLUCTDVWBIRPE

! Easy to break using statistical techniques

- Transpositions

TRANS OIPSR
POSIT NOTNT
IONS OSAI

8

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 . 10^{26} keys**
- trying all possibilities at 1 nanosecond per key requires....

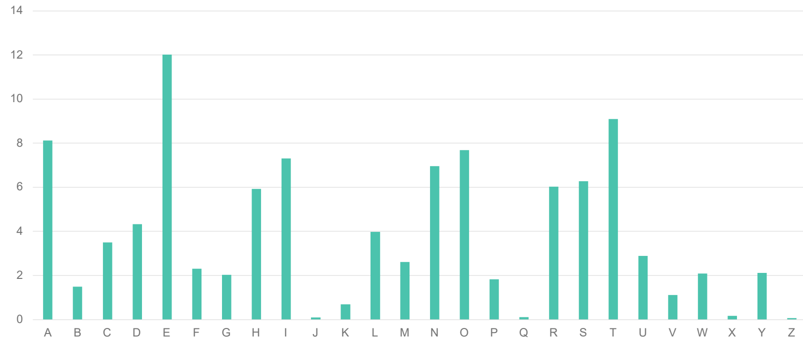
$$4.10^{26} \;/(10^9 \;.\; 10^5 \;.\; 4\; 10^2) = 10^{10} \text{ years}$$

keys per second

seconds per day

days per year

Letter distributions



Substitutions ABCDEFGHIJKLMNOPQRSTUVWXYZ

MZNJSOAXFQGYKHLUCTDVWBIRPE

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

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

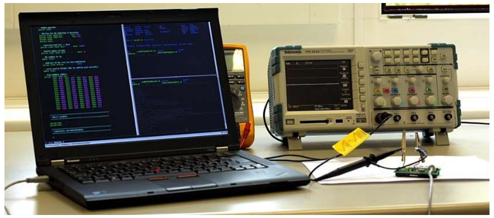
New assumptions on Eve

- 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

13 KU LEUVEN

13

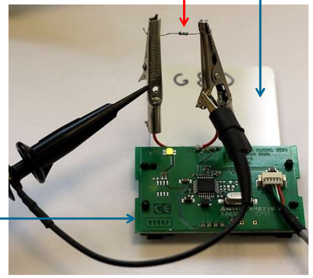
Side channel analysis: power setup



Measure voltage over a resistor to measure the current (and thus the power consumption) of a smart card

resistor

smart card

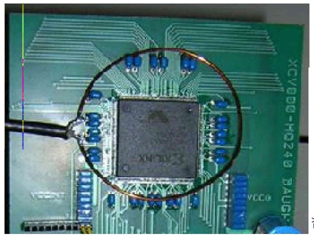


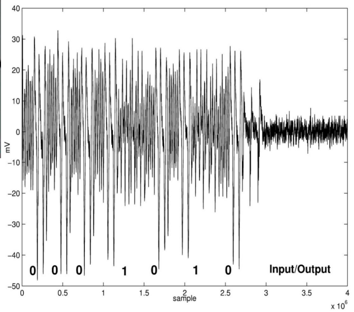
card reader

KU LEUVEN

14

Side channel analysis: electromagnetic setup



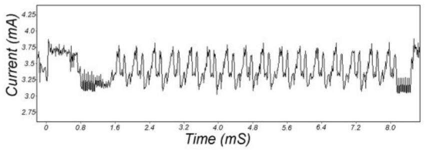


Use simple antenna to measure radiation of an FPGA computing a public key operation

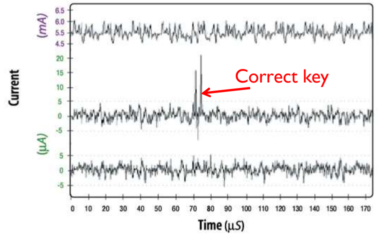
15 KU LEUVEN

15

Simple and differential power analysis: DES block cipher



DES on a smart card: power consumption



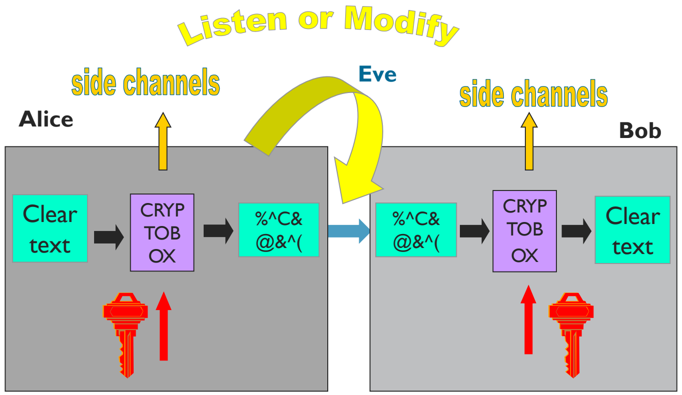
average power

2 correlation methods (example of success and failure)

16 KU LEUVEN

16

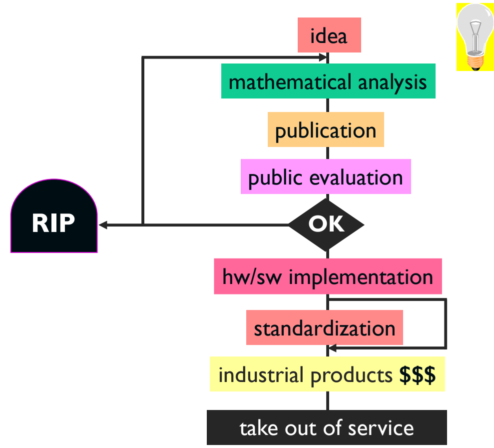
Cryptology + side channels



17

17

Life cycle of a cryptographic algorithm

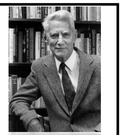


18

One time pad

Vernam scheme (1917)

Shannon (1948)

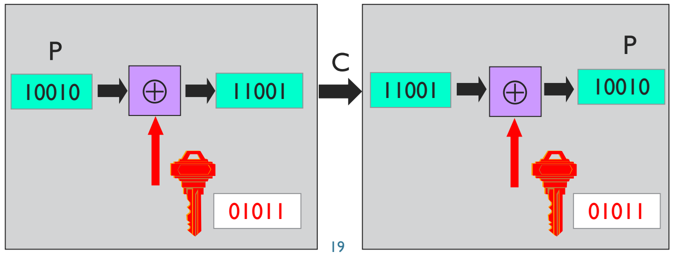


Mauborgne: one time pad (1917+x)

F. Miller (1882)

key is random string, as long as the plaintext

information theoretic proof of security



19

19

One time pad: properties

- perfect secrecy: ciphertext gives opponent no additional information on the plaintext or H(P|C)=H(P)
- impractical: key is as long as the plaintext
- but this is optimal: for perfect secrecy one has always H(K) $\geq$ H(P)

20

20

One time pad: Venona Project (1942-1948)

$c_1 = p_1 + k$
$c_2 = p_2 + k$
then $c_1 - c_2 = p_1 - p_2$

Example:
c1 V c2
(not +)

a skilled cryptanalyst can recover p_1 and p_2 from $p_1 - p_2$ using the redundancy in the language

reuse of key material is also known as "transmission in depth"
https://en.wikipedia.org/wiki/Venona_project

21

21

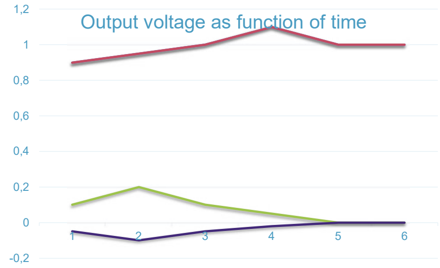
One time pad: insecure implementation

0 + 1 = 1

1 + 0 = 1

0 + 0 = 0

1 + 1 = 0

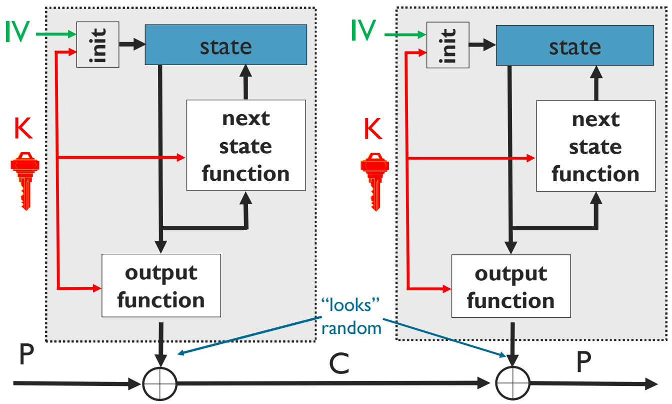


Implementation weakness: identical mathematical symbols can result in different electrical signals

22

22

Synchronous Stream Cipher (SSC)



23

23

Exhaustive key search

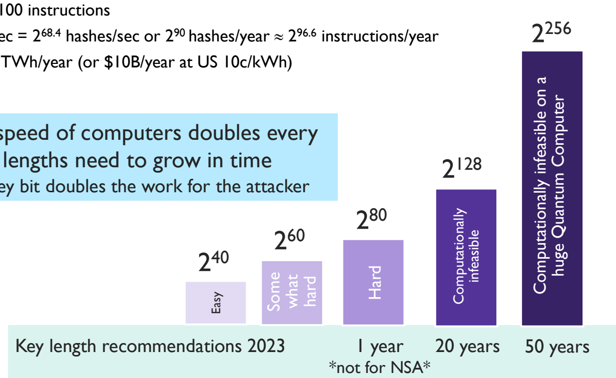
2023: 1 million machines with 24 cores @ 6 GHz can execute 2^{57} instructions/sec or 2^{81} instructions/year

» trying 1 key ≈ 100 instructions

Bitcoin: 400 Exahashes/sec = $2^{68.4}$ hashes/sec or 2^{90} hashes/year ≈ $2^{96.6}$ instructions/year

» Electricity: 100 TWh/year (or $10B/year at US 10c/kWh)

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



24

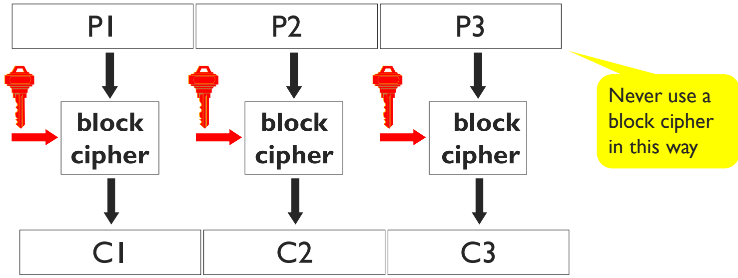
24

High profile stream ciphers

- A5/1 (GSM) (64 or 54)
- E0 (Bluetooth) (128)
- RC4 (browser) (40-128)
- SNOW-3G (3GSM) (128)
- HC-128 (128)
- Trivium (80)
- ChaCha20 (128)

$\left.\right\}$ Legacy - insecure! (A5/1, E0, RC4)

Block cipher



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

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

Widely used block ciphers

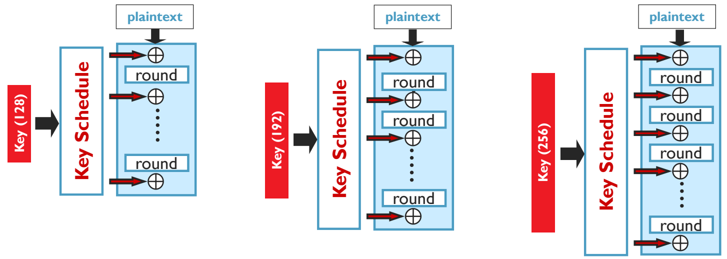
- DES: outdated (56-bit key)
- 3-DES: financial sector
- AES
- KASUMI (3G/4G)
- Keeloq (remote control for cars, garage doors) - insecure

AES variants (2001)

AES-128
10 rounds
Sensitive/classified (SECRET)

AES-192
12 rounds
Classified (TOP SECRET)

AES-256
14 rounds
Classified (TOP SECRET)



Encryption limitations

- Typically does not hide the **leng**th of the plaintext (unless randomized padding but even then…)
- Ciphertext becomes random string: "normal" crypto does not encrypt a credit card number into a (valid) credit card number
- Does **not** hide existence of plaintext (requires steganography)
- Does **not** hide that Alice is talking to Bob (e.g. Tor)
- Does **not** hide traffic volume (requires dummy traffic)

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

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) the **destination** of the information
  - (optionally) **timeliness** and **sequence**

There are no applications that require encryption without data authentication
(but this can still be found in legacy applications with as excuse performance)

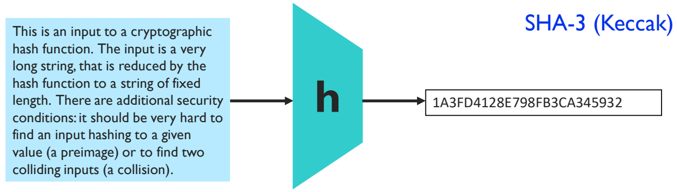
Data authentication: the problem

- problem of replay of messages needs to be addressed at higher layer (e.g. transaction counter in financial systems)
- specific challenges:
  - very long streams
  - versioning systems
  - noisy data
  - ,....

Data authentication: hash function

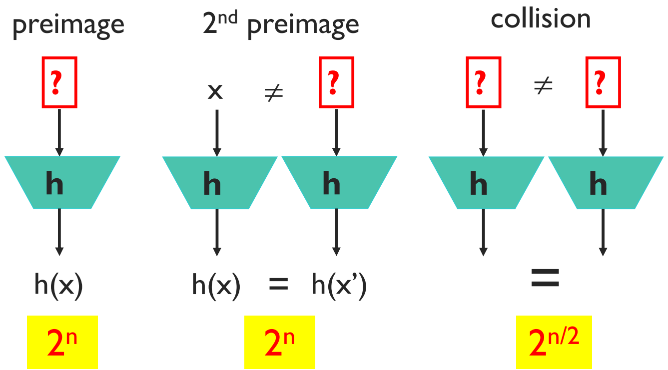
- MDC (manipulation detection code)
- Protect short hash value rather than long text

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



Shift authenticity of file to authenticity of short hash value

Hash function: security requirements (n-bit result)



Data authentication: hash function

- preimage resistance: for given y, hard to find input x such that h(x) = y (2^n operations)
- 2^{nd} preimage resistance: hard to find x'≠ x such that h(x') = h(x) (2^n operations)
- collision resistance: hard to find (x,x') with x' ≠ x such that h(x') = h(x) ($2^{n/2}$ operations)

Widely used hash functions

- MD5
  - (2^{nd}) preimage 2^{128} steps (improved to 2^{123} steps)
  - collisions 2^{64} steps

shortcut: Aug. '04: 2^{39} steps; '09: 2^{20} steps

- SHA-1:
  - (2^{nd}) preimage 2^{160} steps
  - collisions 2^{80} steps

0.15 M$ for 1 year in 2021

shortcut: Aug. '05: 2^{69} steps

Feb. 2017: 2^{61} steps

- SHA-2 family (2002)
- SHA-3 family (2013) – Keccak (Belgian design)
  - (2^{nd}) preimage 2^{256} .. 2^{512} steps
  - collisions 2^{128} .. 2^{256} steps

37

KU LEUVEN

37

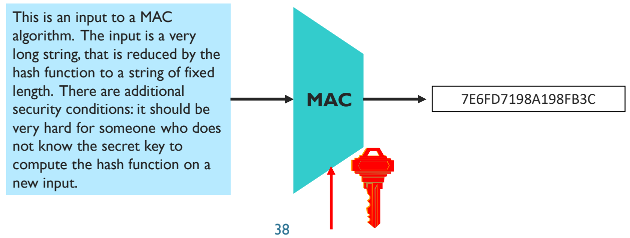
Data authentication: MAC algorithms

- Replace protection of authenticity of (long) message by protection of secrecy of (short) key
- Append MAC to the plaintext

CBC-MAC (CMAC/LMAC)

HMAC

GMAC

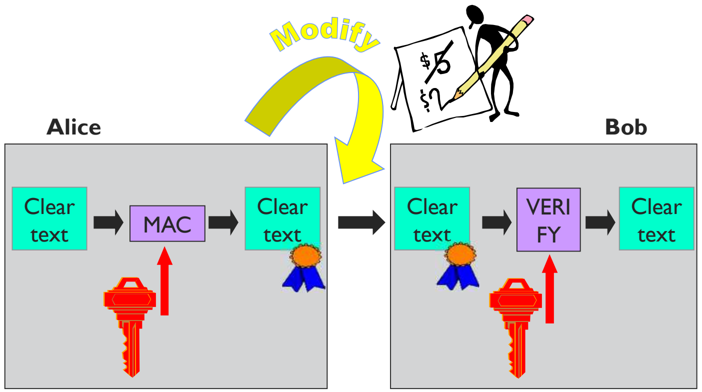


38

KU LEUVEN

38

Data authentication: MAC algorithms



39

KU LEUVEN

39

Data authentication: MAC algorithms

- typical MAC lengths: (32)..64..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

40

KU LEUVEN

40

MAC algorithms

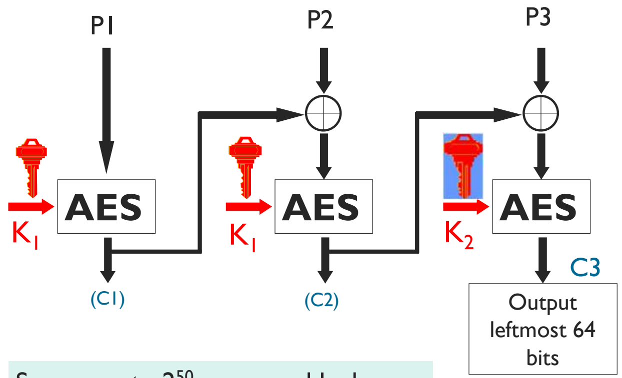
- Banking: CBC-MAC based on triple-DES
- Internet: HMAC and CBC-MAC based on AES
- information theoretic secure MAC algorithms (authentication codes): GMAC/Poly1305
  - rather efficient
  - part of the key refreshed per message

41

KU LEUVEN

41

CBC-MAC based on AES (LMAC)



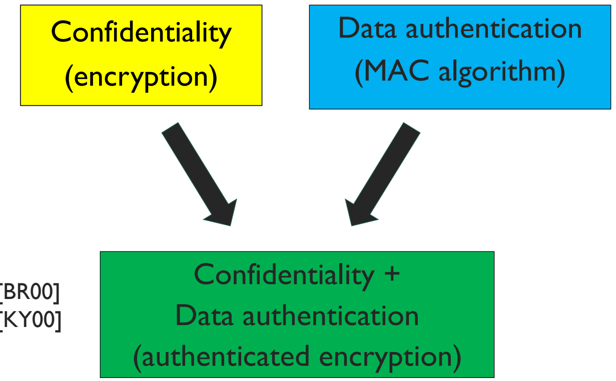
Secure up to 2^{50} message blocks

42

KU LEUVEN

42

Authenticated Encryption



[BR00]
[KY00]

43

KU LEUVEN

43

Authenticated Encryption

Generic composition [BN'00][NRS'14]

- Encrypt-then-MAC with 2 independent keys
  - IPsec, TLS 1.2, 1.3
- MAC-then-Encrypt with 2 independent keys
  - TLS 1.1 and older, 802.11i WiFi
- MAC-and-Encrypt with 2 independent keys

Design "from scratch"

- Integrated authenticated encryption schemes: combined operation with 1 key: see next slide

44

KU LEUVEN

44

Authenticated Encryption: properties wish list

- Associated Data: Authenticated Encryption with Associated Data (AEAD)
- Parallelizable
- Online for encryption
- Security reduction
- Resistance to nonce reuse
- Incremental tags
- Fragmentation
- No release of unverified plaintext
- Key committing
- Flexible implementation sizes
- Performance: speed/size
- Secure implementations: constant time/power analysis/EM analysis/fault attacks…

45 KU LEUVEN

45

Widely used block ciphers

- GCM
- CCM
- GCM-SIV: more robust
- OCB2: faster than GCM
- Aegis: fast
- Ascon: lightweight

46 KU LEUVEN

46

Outline

- Symmetric cryptology
  - confidentiality
  - data authentication
  - authenticated encryption
- Public key cryptology (asymmetric cryptology)
- Hybrid cryptology
- Distributing public keys
- Applications

47 KU LEUVEN

47

Public-key cryptology

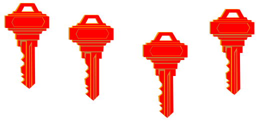
- the problem
- public-key encryption
- digital signatures
- Diffie-Hellman
- RSA

48 KU LEUVEN

48

Limitation of symmetric cryptology

- Reduce security of information to security of keys

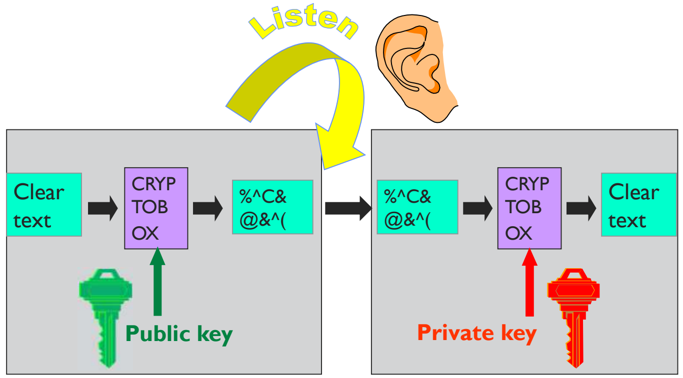


- But: how to establish these secret keys?
  - cumbersome and expensive
  - or risky: all keys in 1 place
- Do we really need to establish secret keys?

49 KU LEUVEN

49

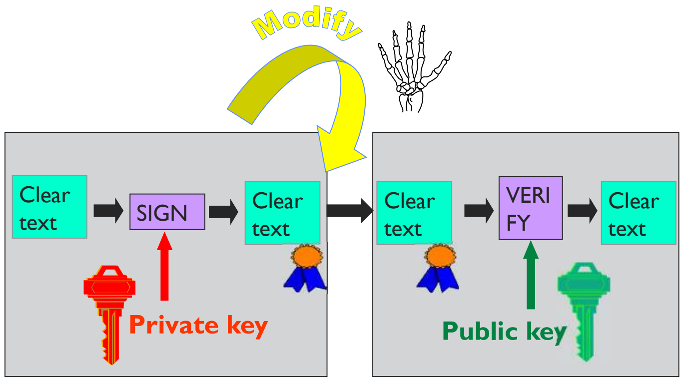
Public key cryptology: encryption



50 KU LEUVEN

50

Public key cryptology: digital signature



51 KU LEUVEN

51

A public-key agreement protocol: Diffie-Hellman

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

generate x	α^{x} $\longrightarrow$	*generate* y
compute α^{x}	$\longleftarrow$ α^{y}	*compute* α^{y}
compute $k=(\alpha^{y})^{x}$		*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} (the discrete logarithm problem)

52 KU LEUVEN

52

RSA ('78)

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

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

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

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

try to factor 2419

53

53

If a large quantum computer can be built

public-key cryptography algorithms have to be replaced [Shor'94]

RSA, Diffie-Hellman (including elliptic curves)

Breaking RSA-2048 requires 4096 ideal qubits or 20 million real qubits

symmetric crypto: key sizes: x2 [Grover'96]

but huge quantum devices needed

54

54

How to continue?

- Pre-Quantum era
  - RSA / ECC (Elliptic Curve Cryptography)
- Hybrid era
  - RSA / ECC + Post-Quantum cryptography
- Post-Quantum Era
  - Once confidence in post-quantum is high enough

First draft standards published August 2023
Ongoing work on other algorithms – standardization work will likely be completed by 2027

55

Advantages of public key cryptology

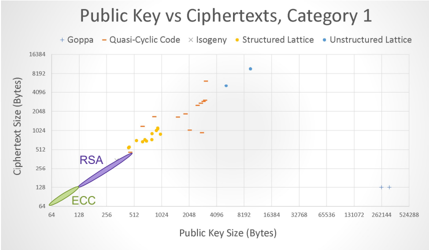
- Reduce protection of information to protection of authenticity of public keys
- Confidentiality without establishing secret keys
  - extremely useful in an open environment
- Data authentication without shared secret keys: digital signature
  - sender and receiver have different capability
  - third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

56

56

Disadvantages of public key cryptology

- Calculations in software or hardware two to three orders of magnitude slower than symmetric algorithms
- Longer keys: 64-512 bytes rather than 10..32 bytes
- What if factoring is easy or if a large quantum computer can be built?
- Post-quantum cryptography



KU LEUVEN

57

Outline

- Symmetric cryptology
  - confidentiality
  - data authentication
  - authenticated encryption
- Public key cryptology (asymmetric cryptology)
- Hybrid cryptology
- Distributing public keys
- Applications

58

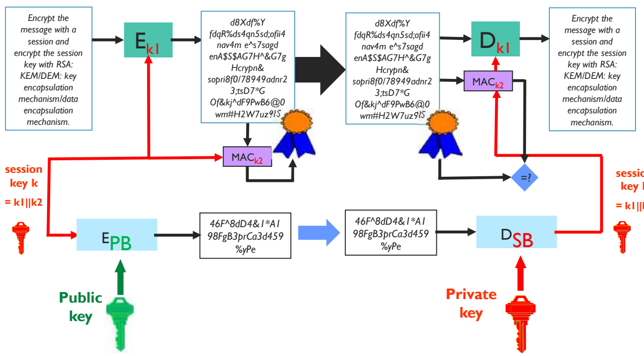
KU LEUVEN

58

RSA encryption for long messages (KEM/DEM)

encryption: $c = m^e \bmod n$

decryption: $m = c^d \bmod n$



59

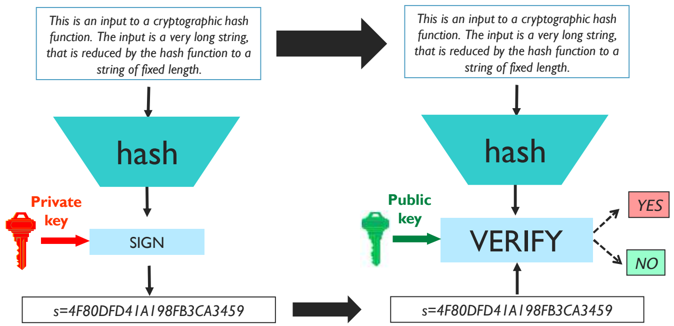
KU LEUVEN

59

RSA signatures for long messages (with appendix)

signature: $s = h(m)^d \bmod n$

verification: $h(m) = s^e \bmod n$



60

KU LEUVEN

60

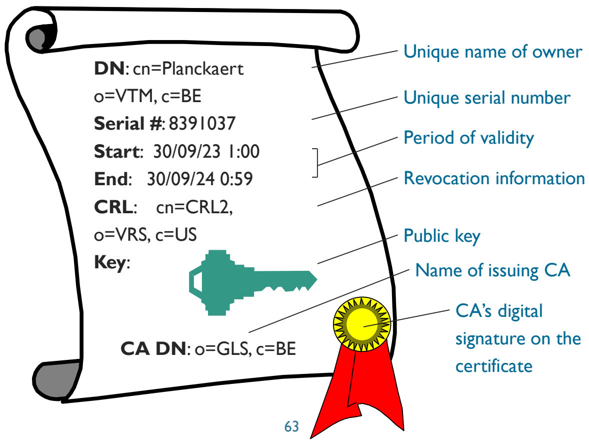
Outline

- Symmetric cryptology
  - confidentiality
  - data authentication
  - authenticated encryption
- Public key cryptology (asymmetric cryptology)
- Hybrid cryptology
- Distributing public keys
- Applications

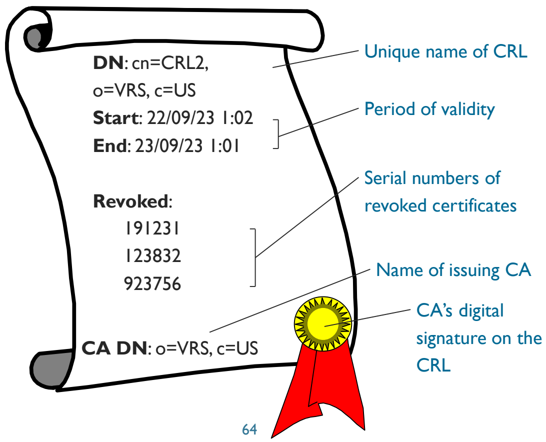
How to distribute public keys?

- Problem sounds trivial but it is highly nontrivial at a large scale
  - unique names
  - centralization/hierarchy
  - revocation
  - privacy

What is a Public-Key Certificate?



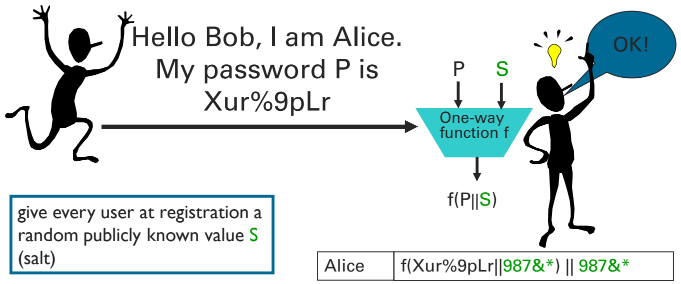
What is a Certificate Revocation List?



Outline

- Symmetric cryptology
  - confidentiality
  - data authentication
  - authenticated encryption
- Public key cryptology (asymmetric cryptology)
- Hybrid cryptology
- Distributing public keys
- Applications

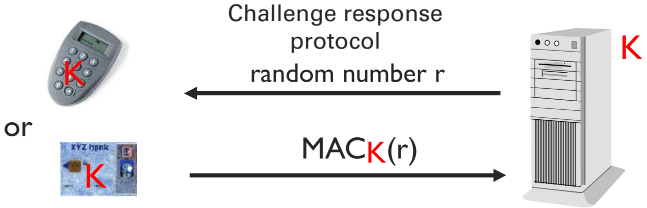
Entity authentication with passwords



Bob stores f(P,S) || S rather than Alice's secret P

motivation for salt S: makes it harder to attack the passwords of all users simultaneously

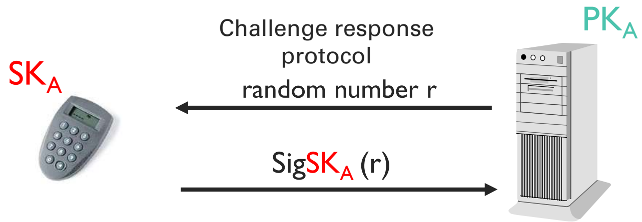
Entity authentication with symmetric token



- Eavesdropping no longer effective
- Bob still needs secret key K
- IETF RFC 4226 HOTP (2005) HMAC-based One Time Password

Detects whether Alice is alive!

Entity authentication with public key token



Eavesdropping no longer effective

Bob no longer needs a secret – only PK_A

Application: EMV

Applications of cryptography: protection of data at rest

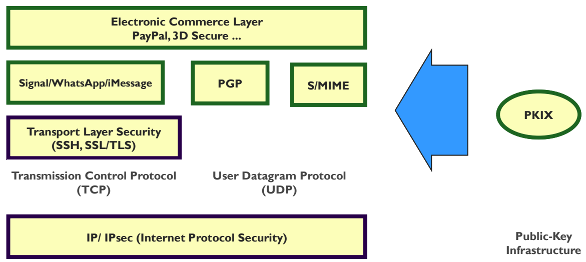
- Hard disk encryption (e.g. Bitlocker, Veracrypt, Ciphershed)
- Database encryption
- File encryption
- Encryption in the cloud

Main question: who manages the decryption keys?

69

69

Applications of cryptography: network security



- security services depend on the layer of integration:
  - the mechanisms can only protect the payload and/or header information available at this layer
  - header information of lower layers is not protected!!

70

70

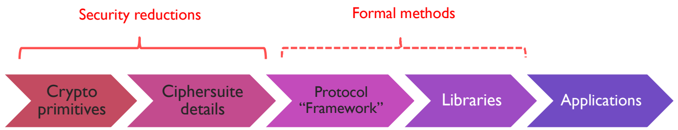
Applications of cryptography: network security (2)

- Data link layer
  - 2G, 3G, 4G, 5G
  - WLAN
  - Bluetooth

71

71

TLS overview [Stebila'19]

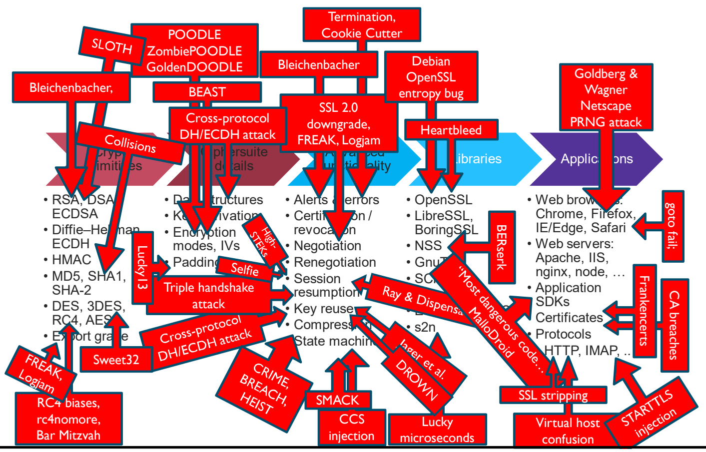


RSA, DSA, ECDSA	Data structures	Alerts & errors	OpenSSL	Web browsers
DH, EC-DH	Key derivation	Certification/revocation	LibreSSL	Web servers
HMAC	Encryption modes and IVs	(Re-)Negotiation	BoringSSL	Application SDKs
MD5, SHA-1, SHA-2	Padding	Session Resumption	NSS	Certificates
DES, 3DES, RC4, AES		Key reuse	GnuTLS	Protocols
Export grade		Compression	SChannel	
		State machine	Java JSSE	
			Everest/miTLS	
			s2n	

72

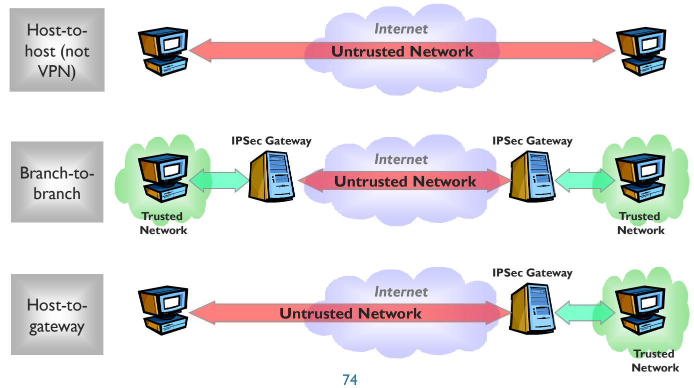
72

Attacks on TLS [Stebila'19]



73

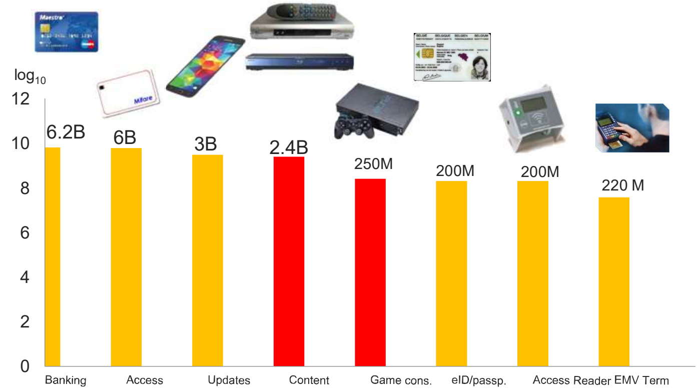
IPsec VPN models: Hosts and Security Gateways



74

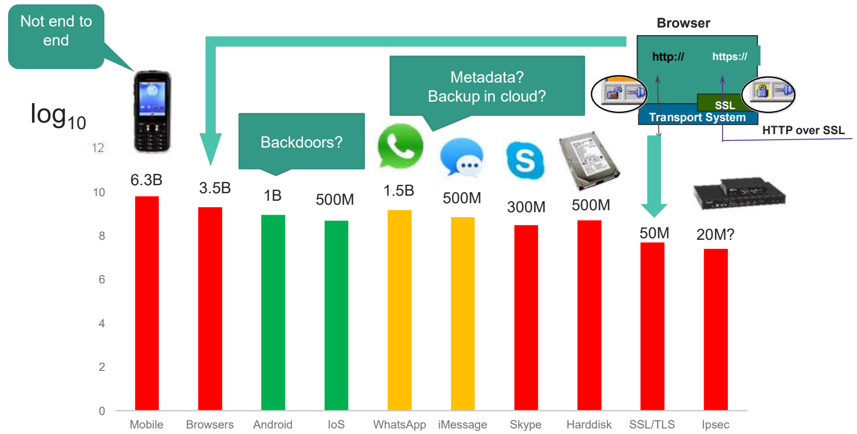
74

Cryptography to protect industry ~60 B



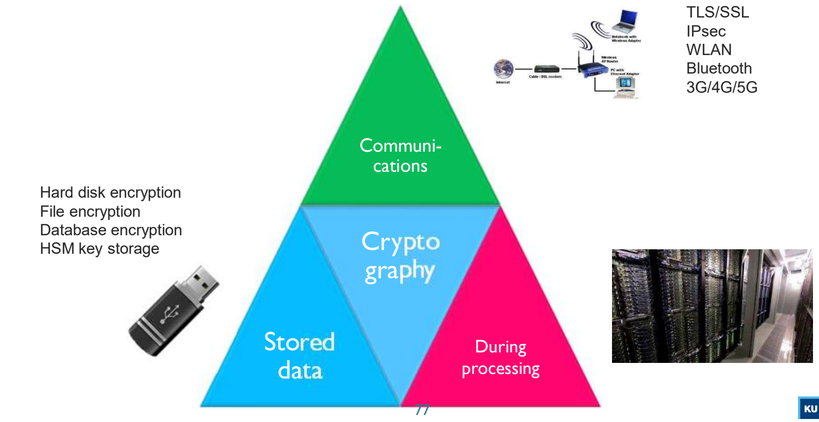
75

Cryptography to protect user data ~30 B



76

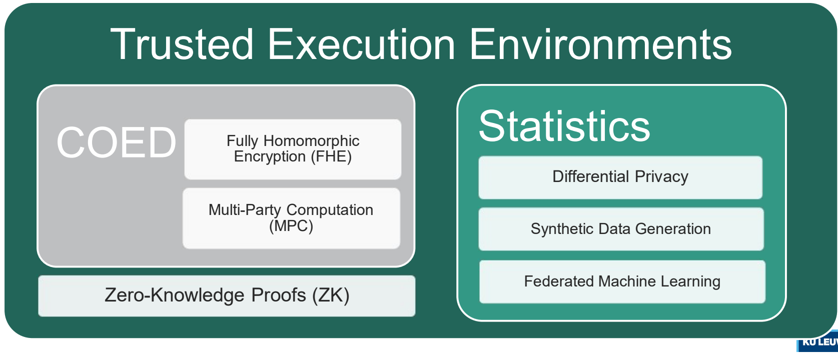
Changing role of cryptography



KU LEUVEN

77

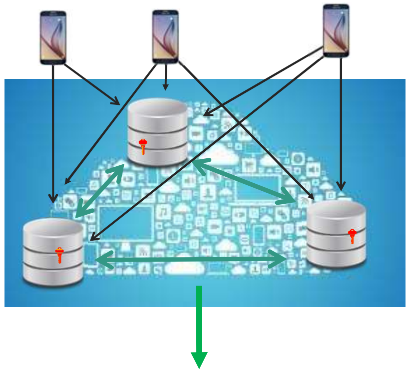
Computing on Encrypted Data (COED)



KU LEUVEN

78

From Big Data to encrypted data
MPC (Multi-Party Computation)



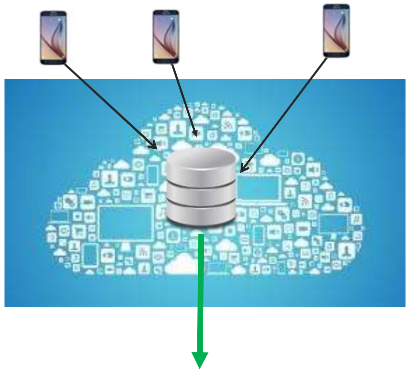
+ secrets shared over multiple servers
+ moderate computation
- high communication overhead

79

KU LEUVEN

79

From Big Data to encrypted data
(somewhat) Fully Homomorphic Encryption



+ single server
+ low communication
- high computation cost
- simple functions: basis statistics, neural networks

80

KU LEUVEN

80

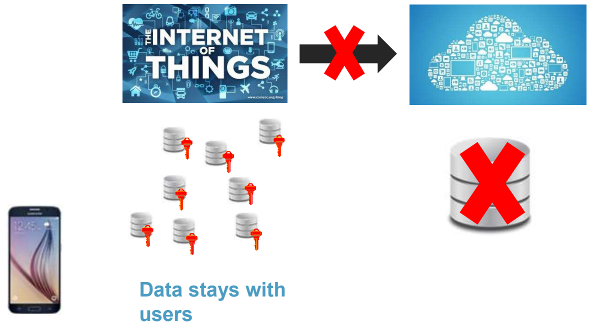
Cool applications

- Elecronic voting
- Cryptocurrencies
- Privacy-preserving contact tracing
- Inference on encrypted data
- AI training on encrypted data
- Dark markets

KU LEUVEN

81

From Big Data to small local data



82 KU LEUVEN

82

Reading material

B. Preneel, Modern cryptology: an introduction

- This text corresponds more or less to the first 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

83 KU LEUVEN

83

Selected books on cryptology and applications

- 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 but outdated– not suited as a first text book. http://www.cacr.math.uwaterloo.ca/hac
- D. Boneh, V. Shoup, A Graduate Course in Applied Cryptography, https://toc.cryptobook.us/ Draft. Very advanced course with interesting applications.
- N. Smart, *Cryptography Made Simple*, Springer, 2015. Solid and up to date but on the mathematical side.
- D. Stinson, M. Peterson, *Cryptography: Theory and Practice*, CRC Press, 4th Ed., 2018. Solid introduction, but only for the mathematically inclined.
- Jonathan Katz and Yehuda Lindell, *Introduction to Modern Cryptography,* Chapman & Hall, 2014. Rigorous and theoretical approach.
- M. Rosulek, The Joy of Cryptography, https://web.engr.oregonstate.edu/~rosulekm/crypto/
- A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, Princeton, 2016. Excellent introduction to the field.
- B. Schneier, *Applied Cryptography*, Wiley, 1996. Widely popular but no longer up to date– make sure you get the errata, online.
- P.C. van Oorschot, Computer Security and the Internet: Tools and Jewels, Springer, 2019. Brief chapters on cryptography, https://link.springer.com/book/10.1007/978-3-030-33649-3
- R. Anderson, Security Engineering, Wiley, 2nd Ed., 2008. Insightful. A must read for every information security practitioner. First edition is available for free at http://www.cl.cam.ac.uk/~rja14/book.html
- W. Stallings, Cryptography and Network Security: Principles and Practice, Pearson, 8th Ed., 2020. Solid background on network security. Explains basic concepts of cryptography.

84 KU LEUVEN

84

Selected books on crypto policy

- G. Greenwald, *No place to hide, Edward Snowden, the NSA, and the U.S. Surveillance State*, Metropolitan Books, 2014
- W. Diffie, S. Landau, *Privacy on the Line. The Politics of Wiretapping and Encryption.* Updated And Expanded Edition, MIT Press, 2010
- S. Landau, *Surveillance or Security? The Risks Posed by New Wiretapping Technologies*. MIT Press, 2013
- S. Landau, *Listening In: Cybersecurity in an Insecure Age*, Yale University Press, 2017
- US National Academies, *Decrypting the Encryption Debate*, 2018, https://www.nap.edu/read/25010/chapter/1

85 KU LEUVEN

85

One time pad: Extra material not covered during the lectures

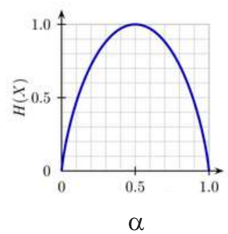
- Generalization from mod 2 to mod L for an integer $L \geq 2$
- p, c, k are strings of length N over the alphabet {0,1, … L-1}
- each key character is generated **uniformly at random** and used **only once**
- Encryption: $c_i = p_i + k_i \mod L \quad i = 1,\dots,N$
- Decryption: $p_i = c_i \ \ k_i \mod L \quad i = 1,\dots,N$

86 KU LEUVEN

86

One time pad: security (Shannon 1948)

- Let X be a finite set and $\{pr(x)\}_{x \in X}$ be a probability distribution on X with $pr(x) \neq 0$
- **entropy H(X)** $= - \Sigma_{x \in X} \ pr(x) \log_2 pr(x)$



Example: binary variable
X = {0,1}, pr(0) = α, pr(1) = 1-α

$H(X) = - \alpha \log_2 (\alpha) - (1- \alpha) \log_2 (1- \alpha)$

If X contains N elements, then $H(X) \leq N \log_2 L$

87 KU LEUVEN

87

One time pad: security (2)

- Let X,Y be finite sets and $\{pr(x)\}_{x \in X}$ and $\{pr(y)\}_{y \in Y}$ be probability distributions on X resp. Y with pr(x), pr(y) $\neq 0$
- Let $\{pr(x,y)\}_{x \in X, y \in Y}$ be a joint probability distribution on the Cartesian product of X and Y
- Then we define:
  - **joint entropy H(X,Y)** $= - \Sigma_{x \in X, y \in Y} \ pr(x,y) \log_2 pr(x,y)$
  - **conditional entropy H(X|Y)** $= - \Sigma_{x \in X, y \in Y} \ pr(x,y) \log_2 pr(x|y)$
- Fact: $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$

Note: take sums over values of x and y for which $pr(x,y) \neq 0$

88 KU LEUVEN

88

One time pad: security (3)

- L=26, N=5 c_i = (p_i + k_i) mod 26; p_i = (c_i - k_i) mod 26
  - with c_i, p_i, $k_i \in$ [0,25]; A=0, B=1, …, Z=25
- consider ciphertext C= XHGRQ
  - with key AAAAA P = XHGRQ
  - with key VAYEK P = CHINA
  - with key EZANZ P = TIGER
  - …
  - with key ZZZZZ P = YIHSR
- conclusion: for every 5-character plaintext there is a 5-character key which maps the ciphertext to that plaintext

89

89

One time pad: security (4)

The one time pad offers perfect secrecy

H(P | C) = H(P) or the ciphertext provides no additional information on the plaintext

Proof

$\forall$ p, c $\exists$ k with c = p + k

pr(c | p) = pr (p | c) = pr(k) = $1/L^N$ - this holds $\forall$ p, c thus independent of probability distribution of p

hence p and c are statistically independent

then H(P,C)=H(C)+H(P) and thus H(P | C) = H(P)

Indeed: H(P, C) = H(C) + H(P | C) = H(P) + H(C | P)

90

90

One time pad uses minimal key size

perfect secrecy: H(P | C) = H(P)

H(P | C) $\leq$ H(P, K | C)
= H(K | C) + H(P | K, C)
= H(K | C)
$\leq$ H(K)

as H(P | C) = H(P) we obtain that H(K) $\geq$ H(P)

If P contains N_P bits: H(P) $\geq$ N_P

Perfect security: this property holds for any distribution on P and thus also for the special case H(P) = N_P

If K contains N_K random bits: H(K) $\leq$ N_K

Thus $N_K \geq N_P$

91

91

What if a cipher does not offer perfect secrecy?

ciphertext leaks information on plaintext and key

in principle key can be recovered with only N_K/r ciphertext bits with r the redundancy per bit

r = 1 - H(P)/N_P

English: r $\cong$ 0.75

for a 64-bit key: 85 bits are sufficient to recover the key unambiguously (= unicity distance)

note: in practice known plaintext is often available; in this case the key can be recovered once N_K ciphertext and corresponding plaintext characters are known

92

92

Lessons learned: redundancy harms confidentiality

- one can remove redundancy by **data compression** before encryption
- if r $\rightarrow$ 0 unicity distance $\rightarrow \infty$ (ideal cipher)
  - but practical compression algorithms are not perfect
  - known plaintext may make scheme very vulnerable anyway
  - timing attacks are also a problem
- other solution: **homophonic coding**
- make plaintext distribution more uniform by introducing randomized redundancy
  - again: known plaintext may make scheme very vulnerable anyway

In practice: non-constant time compression leaks information on the plaintext; but constant time compression is hard

93

KU LEUVEN

93