COSIC

iMinds CONNECT.INNOVATE.CREATE

KU LEUVEN

Public key cryptography: The next 4 decades

Bart Preneel

COSIC KU Leuven and iMinds, Belgium
Bart.Preneel(at)esat.kuleuven.be
April 2015

Thanks to Christiane Peters, Alan Szepieniec, and Frederik Vercauteren

Outline

- Public-key cryptography today
- Risks of current public-key cryptography
- The future: post-quantum crypto
- The future: more than algorithms

Diffie-Hellman'75 Merkle'75

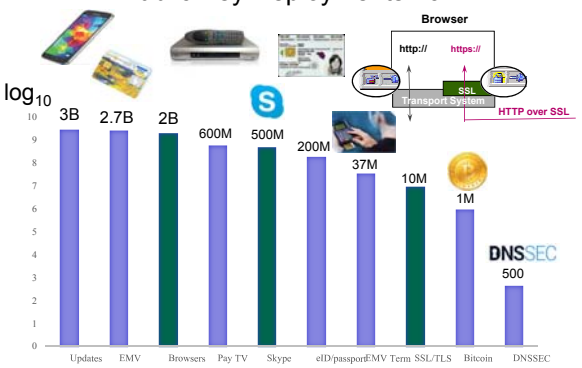
- Can two people who have never met have a private conversation?*
- Is it possible to digitally sign documents?

* Ellis – non-secret encryption (unpublished '69)

Deployment of cryptography

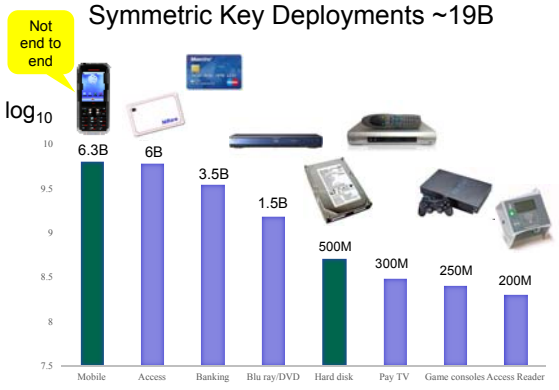
- most crypto in volume and market serves for data and entity authentication
  - code updates
  - payments: credit/debit/ATM/POS and SSL/TLS
  - access tokens
- confidentiality
  - government/military secrets
  - DRM/content protection
  - telco: not end-to-end or with a backdoor
  - hard disk encryption: backdoored?
  - ehealth (growing market)
  - most data in the cloud is not encrypted

Public Key Deployments ~9B



Missing: IPsec, SSH

Symmetric Key Deployments ~19B



Status of Cryptography

COMSEC:

- limited fraction (a few %) of traffic is protected.
- very small fraction of traffic is protected **end-to-end** with a **high security level** and without a backdoor (email/voice)
  - New solutions emerging: Silent Circle, Blackphone
- need authenticated encryption/secure channels
  - reordering, replay, deletion of packets
- protection of **meta-data is very hard**

7

Status of Cryptography

COMPUSEC

data at rest: key management problem

- hard disk encryption
- cloud: FHE is not a panacea

secure configuration/boot/execution

the Internet of Things/Everything in 2020 (~ 20-50B)

Cryptography is **NOT** (yet) used to protect Alice and Bob but to protect the (intellectual) property of corporations

8

"Advanced" Cryptography

- Identity based encryption (key escrow problem)
- Threshold cryptography:
  - root keys of Visa and DNS, some voting protocols
- e-auctions
  - beetroots in Denmark
- e-voting
  - Helios
- Searchable encryption
- Functional encryption
- (Somewhat) Fully Homomorphic Encryption (FHE)
- Multi-party computation (MPC)
  - private matching for e-government in Estonia

9

All widely used public-key systems rely on three problems from algebraic number theory

Integer factorization: RSA (n = p.q)
Discrete **LOG**arithm : Diffie-Hellman, DSA: $y = \alpha^x$
Elliptic Curve **D**iscrete **LOG**arithm, ECDSA: Q = x.P

RSA-1024 ~ DLOG-1024 ~ ECC-146
RSA-2048 ~ DLOG-2048 ~ ECC-206
RSA-4096 ~ DLOG-4096 ~ ECC-282

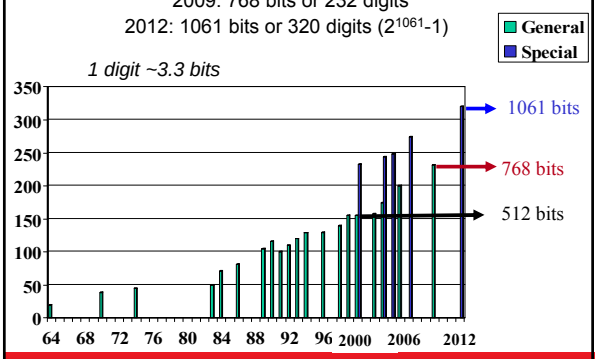
Are these problems hard?

A hard problem is a problem that nobody works on (James L. Massey)

10

Factorisation records (RSA)

2009: 768 bits or 232 digits
2012: 1061 bits or 320 digits (2^{1061}-1)



11

The Cryptocalypse?

2013 breakthrough for DLOG in group of special form

12

Public key crypto security

$L(\alpha)=\exp((\log_2 n)^{\alpha}(\log_2\log_2 n)^{1-\alpha})$

polynomial (weak) — (strong) exponential

- ***L(0)***
- ***L(1/2)*** — 1981 Factoring and DLOG
- ***L(1)*** — best ECC DLOG solvers

Recent progress

- ***L(1/3)*** — 1984 Factoring and (Non-ECC) DLOG stay here for 30 years
- ***L(1/4)*** — DLOG special numbers [Joux Feb'13]
- with restriction on the groups [Barbulescu et al. Jun'13]

Special form DLOG record: 9234 bits [Granger+'13]
Supersingular binary curves 59-bit security << 128 [Granger+'13]

If a large quantum computer can be built...

all schemes based on factoring (RSA) and DLOG are insecure [Shor'94]

- including elliptic curve cryptography

symmetric key sizes: x2 [Grover]

hash sizes: unchanged (for collisions)

News in Jan. 2014: NSA has spent 85 M$ on research to build a quantum computer

14

Upgrade problem: what if large quantum computers arrive?

Data and entity authentication can survive with defense-in-depth

- re-signing or on-line verification
- examples: EMV, Pay TV

Problem is larger for confidentiality:

– require lead time determined by data life time

Upgrades are slow and painful

– probably a few banks are still using single DES
– EMV upgrade from RSA to ECC: 2014-2030
– embedded environments are harder (shellshock)

15

Outline

- Public-key cryptography today
- Risks of current public-key cryptography
- The future: post-quantum crypto
- The future: more than algorithms

16

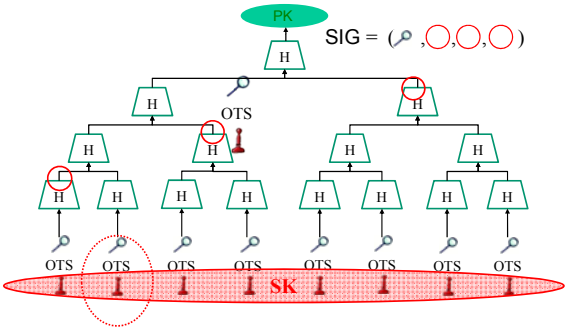
Post-Quantum Cryptography

- Go back to the 1970s
  - digital signatures based on one-way functions
  - public-key encryption based on Error Correcting Coding [McEliece'78] and extensions to rank metrics
  - public key encryption based on lattices (inspired by knapsack problems) (Euclidean distance)

- Go back to the 1980s:
  - multivariate polynomial equations
- So far no good quantum algorithms known to break these systems

17

Hash-Based Signatures



Slide credit: Andreas Hülsing

18

Hash-Based Signatures: variant XMSS

C Implementation using OpenSSL on Intel(R) Core(TM) i5-2520M CPU @ 2.50GHz with Intel AES-NI [BDH'11]

	Sign (M cycles)	Ver fy (M cycles)	Signature (Kbit)	Public Key (Kbit)	Secret Key (Kbit)	Bit Security	Comment
XMSS-SHA-2	89	5.0	16.3	13.3	26.3	157	h = 20, w = 64,
XMSS-AES-NI	1.3	0.18	19.2	7.2	13.2	84	h = 20, w = 4
RSA 2048	7.7	0.23	≤ 2	2	2	87-103	

Slide credit: Andreas Hülsing

19

McEliece (1978): code-based public-key crypto

Public key

a random-looking binary linear code given by a matrix H
weight w

Private key

random-looking code is a disguised Goppa code with error-correction capability w

Encryption

encode a plaintext as weight-w word e and send syndrome $s=H \cdot e$

Decryption

after conversion use standard Goppa-code decoders to determine low-weight solution e

Slide credit: Christiane Peters

20

McEliece security notions

Private key security
Relies on the difficulty of retrieving inner code from public matrix H and thus getting access to efficient decoding

Message security
decryption security relies on NP-hardness of the syndrome-decoding problem foro random code - assuming that structure of H does not leak
(best known algorithms take exponential time)

Slide credit: Christiane Peters

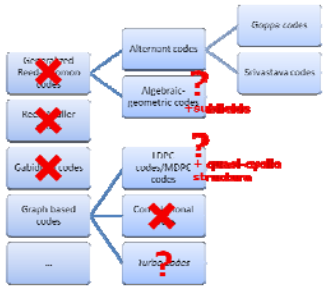
21

McEliece: suitable codes don't have too much structure

McEliece's original proposal Goppa codes is still holding up

large key sizes: 187kB for 128-bit security

Need to randomize plaintext!



Slide credit: Christiane Peters

22

Performance McEliece

C Implementation on Intel Core i5-3210M, Ivy Bridge

	Decrypt (M cycles)	Encrypt (M cycles)	Public Key (Kbit)	Secret Key (Kbit)	Bit Security	Comment
RSA-2048	10.4	0.37*	2	2	87-103	
DH binary ECC	0.077	0.078*	0.5	0.5	127	
McEliece	0.060	0.073*	1496	1496	128	(n,w) = (212,41)

* estimated

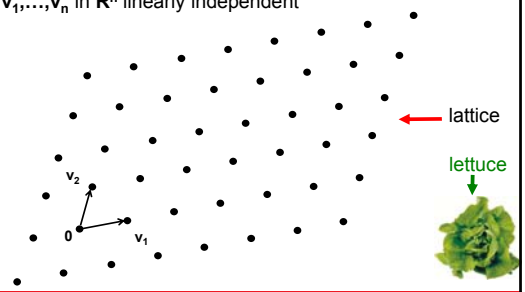
23

Lattices

A **lattice** is a set of points

$$L=\{a_1v_1+\ldots+a_nv_n \mid a_i \text{ integers}\}$$

with $v_1,\ldots,v_n$ in R^n linearly independent



Slide credit Frederik Vercauteren

24

Short-Integer-Solution (SIS)

- $\mathbf{Z_q^n}$ = n-dimensional vectors modulo q
- Given n x m matrix A, find small non-zero vector **z** such that

$$\begin{pmatrix} \mathbf{a_1} & \mathbf{a_2} & \dots & \mathbf{a_m} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{pmatrix} = \begin{pmatrix} \mathbf{0} \end{pmatrix} \mod q$$

Slide credit Frederik Vercauteren 25

Learning With Errors (LWE)

- $\mathbf{Z_q^n}$ = n-dimensional vectors modulo q, error rate $\alpha << 1$
- Given m vectors $\mathbf{a_1},\dots,\mathbf{a_m}$ in $\mathbf{Z_q^n}$
- **Search:** find secret vector **s** in $\mathbf{Z_q^n}$ given "noisy" inner products

$$\mathbf{b_1} = < \mathbf{a_1} , \mathbf{s} > + \mathbf{e_1}$$
$$\mathbf{b_2} = < \mathbf{a_2} , \mathbf{s} > + \mathbf{e_2}$$
$$\dots.$$
$$\mathbf{b_m} = < \mathbf{a_m} , \mathbf{s} > + \mathbf{e_m}$$

- Errors e_i are taken from Gaussian over **Z** with **deviation** αq

- **Search LWE =** noisy linear algebra modulo q
- m x n matrix **A** with rows $\mathbf{a_i}$: $\mathbf{A\ s^t = b^t + e^t}$

Slide credit Frederik Vercauteren 26

LWE-based Encryption: Parameters

- best BDD solver [Liu-Nguyen'13]
- 128-bit security (2^{128} basic ops):
  - dimension n = 256
  - prime q = 7681
  - parameter of Gaussian error distribution ~ 11 (st. dev. $11/\sqrt{2\pi}$)
- public key: 104 Kbyte
- ciphertext: 416 byte

- public key and ciphertext expansion can be reduced with ring-LWE (structured A instead of random A)
  - hardness related to problems in "ideal" lattices

Slide credit Frederik Vercauteren 27

Key Aspects of Lattice-based Systems

Pros

- efficient and parallizable
  - matrix-vector arithmetic, Fast-Fourier Transform for polynomial multiplication
- worst-case to average-case reductions

Cons

- difficult to find good sampling methods
- difficult to assess exact security
- large keys (except for ring versions)
- probabilistic decryption

Slide credit: Christiane Peters 28

Lattice Cryptography for the Internet [Peikert14]

4.4 Instantiating the Parameters

We now instantiate the parameters to analyze their asymptotic behavior and the underlying (worst-case) hardness guarantees. These calculations work for arbitrary choices of m and error parameter $r \geq 1$, and can therefore be slightly loose by small constant factors. Very sharp bounds can easily be obtained for particular choices of m and r using Lemma 4.2.

Since $\mathrm{rad}(m)/m \leq 1$, by Item 2 of Fact 2.4 we have that each $\|g \cdot s_i\|_2 \leq \hat{m} \cdot (r+1) \cdot \sqrt{n}$ except with probability at most 2^{-n}. Similarly, $r'^2 \leq r^2 + 2\pi$. Therefore, by taking $\omega = \sqrt{\ln(2n/\varepsilon)/\pi}$ and

$$q \geq 8\sqrt{(r^2+2\pi)(2\hat{m}^2 \cdot (r+1)^2 + 1) \cdot n \cdot \omega} = O(\hat{m} \cdot r^2 \cdot \sqrt{n}) \cdot \omega,$$

we obtain a probability of decryption failure bounded by $\approx \varepsilon$. Thus we may take $q = O(r^2 \cdot n^{3/2} \log n)$ in the typical case where $\hat{m} = O(n)$ and, say, $\varepsilon = 2^{-128}$.

To apply Theorem 2.7 for $\ell = 2$ samples, we let $r = \xi q$ and $\xi = \alpha \cdot (3n/\log(3n))^{1/4}$, where

- $r = (3n/\log(3n))^{1/4} \cdot \omega(\sqrt{\log n})$ to guarantee $\alpha q \geq \omega(\sqrt{\log n})$, and
- $q = O(r^2 \cdot n^{3/2} \log n) = \tilde{O}(n^2)$ is a sufficiently large prime congruent to one modulo m.

Then we obtain that R-DLWE$_{q,\chi}$ is hard (and hence the KEM is IND-CPA secure, by Lemma 4.1) assuming that SVP on ideal lattices in R is hard to approximate to within $\tilde{O}(\sqrt{n}/\alpha) = \tilde{O}(\sqrt{n} \cdot q) = \tilde{O}(n^{5/2})$ factors for quantum algorithms.

Alternatively, we may conjecture that the search version of ring-LWE with error distribution $\psi = D_r$ is hard for $r \geq \omega(\sqrt{\log n})$ (or even $r \geq 1$), which by [LPR10, Theorem 5.3] implies that R-DLWE$_{q,\chi}$ is hard

29

TU Darmstadt Lattice Challenge

1. Ajtai: Generating Hard Instances of Lattice Problems, STOC 1996
2. Buchmann, Lindner, Rückert: Explicit Hard Instances of the Shortest Vector Problem, PQCrypto 2008

HALL OF FAME

Position	Dimension	Euclidean norm	Contestant
1	825	120.37	Yuanmi Chen Phong Nguyen
2	800	106.60	Yuanmi Chen Phong Nguyen
3	775	100.14	Yuanmi Chen Phong Nguyen
4	750	87.76	Yuanmi Chen Phong Nguyen
5	725	80.65	Yuanmi Chen Phong Nguyen

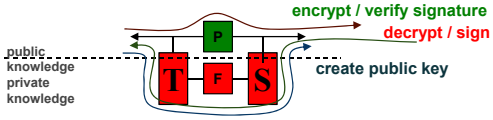
30

Multivariate Quadratic Equations

Public Key:
- system of quadratic polynomials $P : F_q^n \to F_q^m$

Private Key:
- **affine** transformations $T : F_q^m \to F_q^m$ (on output variables) and $S : F_q^n \to F_q^n$ (on input variables)
- central system of **quadratic** polynomials $F : F_q^n \to F_q^m$ (easily invertible)

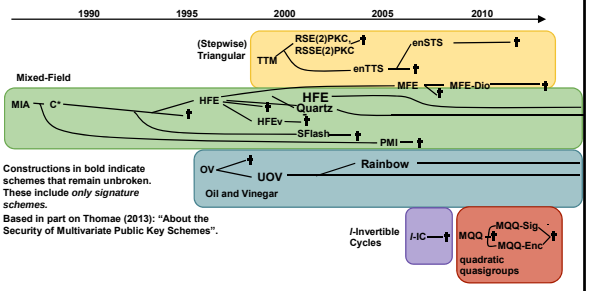
S and T **hide the structure** of F: $P = T \circ F \circ S$



Slide credit: Alan Szepeniec

31

Multivariate Quadratic Equations



Slide credit: Alan Szepieniec

32

Public key crypto

(disclaimer: preliminary results)

Algorithm	Enc. Sign	Speed			Size		
		Enc/ Ver	Dec/ Sign	Key gen	Public key	Private key	Ciphertext/ signature
RSA/DH/DSA	E+S						
ECC	E+S						
Hash-based	S						
Coding-based*	E						
LWE*	E						
SIS*	S						
NTRU	E						
MQ* (e.g rainbow)	S						

* Key sizes can be optimized

33

Outline

- Public-key cryptography today
- Risks of current public-key cryptography
- The future: post-quantum crypto
- The future: more than algorithms

34

Reaction attacks: well known

[Bleichenbacher 98] PKCS #1v1.5 – 1 million chosen ciphertexts
[Bardou+12]: reduced to about 10,000 chosen ciphertexts
- many commercial products vulnerable!

And many more….
[Manger 01] OAEP PKCS #1v2 – a few 1000 chosen ciphertexts
[Bellare-Kohno-Namprempre 02]: SSH
[Vaudenay 02] SSL, IPsec, WTLS...
[Canvel-Hiltgen-Vaudenay-Vuagnoux 03]: SSL/TLS

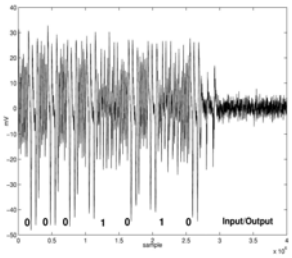
Solution: design new protocols and implementations
- proper KEM/DEM schemes (DEM: symmetric authenticated encryption) – no plug replacement for IKE, TLS [Peikert14]
- don't send error messages (bad engineering practice)

35

Security in Practice

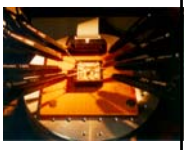
36

Physics trumps Mathematics



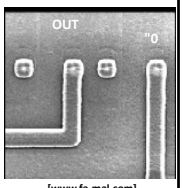
37

Invasive attacks



Passive: micro-probing

Active: modify circuits
- connect or disconnect security mechanism
  - disconnect security sensors
  - RNG stuck at a fixed value
  - reconstruct blown fuses
- cut or paste tracks with laser or focused ion beam



RNG [www.fa-mal.com]

38

Implementation attacks (CHES conference)

Academic
- ever more sophisticated attacks
- broad range of countermeasures: well understood
  - but writing constant time implementations is harder than it seems
- new constructions with security proofs: leakage resilience
- cost in practice: 2-100 times more

Industry
- needs security at cost 20-50% more
- return to security by obscurity
- expensive (but confidential) validation program under Common Criteria

39

Asking for the key

- security letters
- (alleged) examples
  - Lavabit email encryption
  - CryptoSeal Privacy VPN
  - SSL/TLS servers of large companies
  - Truecrypt?

Obtaining the key by hacking or using malware

- (alleged) example: Gemalto

40

If you can't get the private key, substitute the public key

fake SSL certificates or SSL person-in-the-middle

- Flame: rogue certificate by cryptanalysis*
- Comodo, Diginotar, Turktrust
- TLS data stored by GCHQ FLYING PIG (Google, Hotmail, Yahoo!)

* *Stevens*, Counter-cryptanalysis, *Crypto 2013*

41

The CA Mess on the web

[Eckersley10] "An observatory for the SSLiverse"

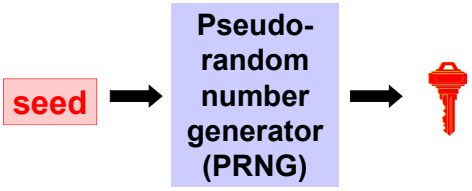
10.8M servers start SSL handshake
4.3M use valid certificate chains
650 CA certs trustable by Windows or Firefox
1.4M unique valid leaf certs
- 300K signed by one GoDaddy cert

80 distinct keys used in multiple CA certs
several CAs sign the IP adr. 192.168.1.2 (reserved by RFC 1918)
2 leaf certs have 508-bit keys
Debian OpenSSL bug (2006-2008)
- resulted in 28K vulnerable certs
- fortunately only 530 validate
- only 73 revoked

42

If you can't get the key

make sure that the key is generated using a random number generator with trapdoor

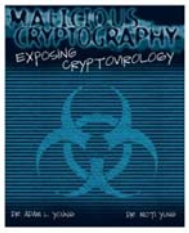


trapdoor allows to predict keys

43

Cryptovirology [Young-Yung]

http://www.cryptovirology.com/cryptovfiles/research.html



Title: Malicious Cryptography – Exposing Cryptovirology

Authors: Adam Young
Moti Yung

Date: February, 2004

Publisher: John Wiley & Sons

44

NSA can (sometimes) break SSL/TLS, IPsec, SSH, PPTP, Skype

- ask for private keys
- implementation weaknesses
- weak premaster secret (IPsec)
- end 2011: decrypt 20,000 secure VPN connections/hour

- http://www.spiegel.de/international/germany/inside-the-nsa-s-war-on-internet-security-a-1010361.html
- http://blog.cryptographyengineering.com/2014/12/on-new-snowden-documents.html

45

COMSEC - Communication Security

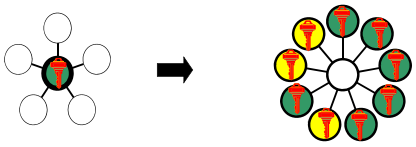
Forward secrecy (e.g. Diffie-Hellman)

Do **not** move problems to the authenticity of a single public key

Do **not** move problems to a single secret key

– solution: threshold cryptography; proactive cryptography

Do protect **meta-data**



46

COMPUSEC - Computer Security

- Simplify to reduce attack surface
- Secure local computation
  - with threshold security
  - Multi Party Computation
  - hardware support: TPM, SMART, Sancus, SGX,…
- Secure cloud computation?
- Secure and open implementations
- Community driven open audit

47

Reconsider every stage

Crypto design	Kleptography
Hardware/software design	Hardware backdoors
Hardware production	
Firmware/sw impl.	Software backdoors
Device assembly	Adding/modifying
Device shipping	hardware backdoors
Device configuration	Configuration errors
Device update	Backdoor insertion

48

Predictions on the Next 40 Years of Public-Key Cryptography

- **????????????:** Computers, communications, storage are all quantum and all classical cryptography disappears
- **Highly unlikely:** public-key cryptography will disappear completely
  - everything online: symmetric cryptography could make a comeback for many applications (e.g. EMV, web security)
- **Probable**: within 5-15 years massive deployment of post-quantum cryptography (hash-based signatures and lattice-based encryption)
- **Probable:** much more sophisticated protocols with distributed crypto and multi-party computation are more widely used
- **Perhaps:** RSA/DLOG/ECC stays around but with much larger key lengths

49