Length of cryptographic keys

Zdeněk Říha

Security of RSA

- We choose randomly 2 primes and compute n and φ(n) :
  - **p, q**
  - **n** = **p·q**
  - **φ(n) = (p-1)(q-1).**
- **e** is chosen such that **gcd(e, φ(n)) = 1.**
- We compute **d = e^{-1} (mod φ(n)).**
- Public key: **n, e.**
  Private parameters: **p, q, d.**
  Private key: **d.**

- Security of RSA cryptosystem is based on the problem of factoring large numbers
- If public **n** can be factored into **p** and **q**, we can calculate **φ(n)** and derive d from e.
- Integer factorization is taught at primary schools
- But when integers are very big it takes very long time even for fast computers to factor the number

Computational Security

- Unconditional vs. computational security
- Security based on a hard problem
- The problem is solvable, but it takes impractically long time to solve
- The attacker cannot wait thousands/millions of years to break the encryption
- Our expectations can change:
  - Progress in the speed of HW
  - Progress in the efficiency of algorithms

History of RSA Security

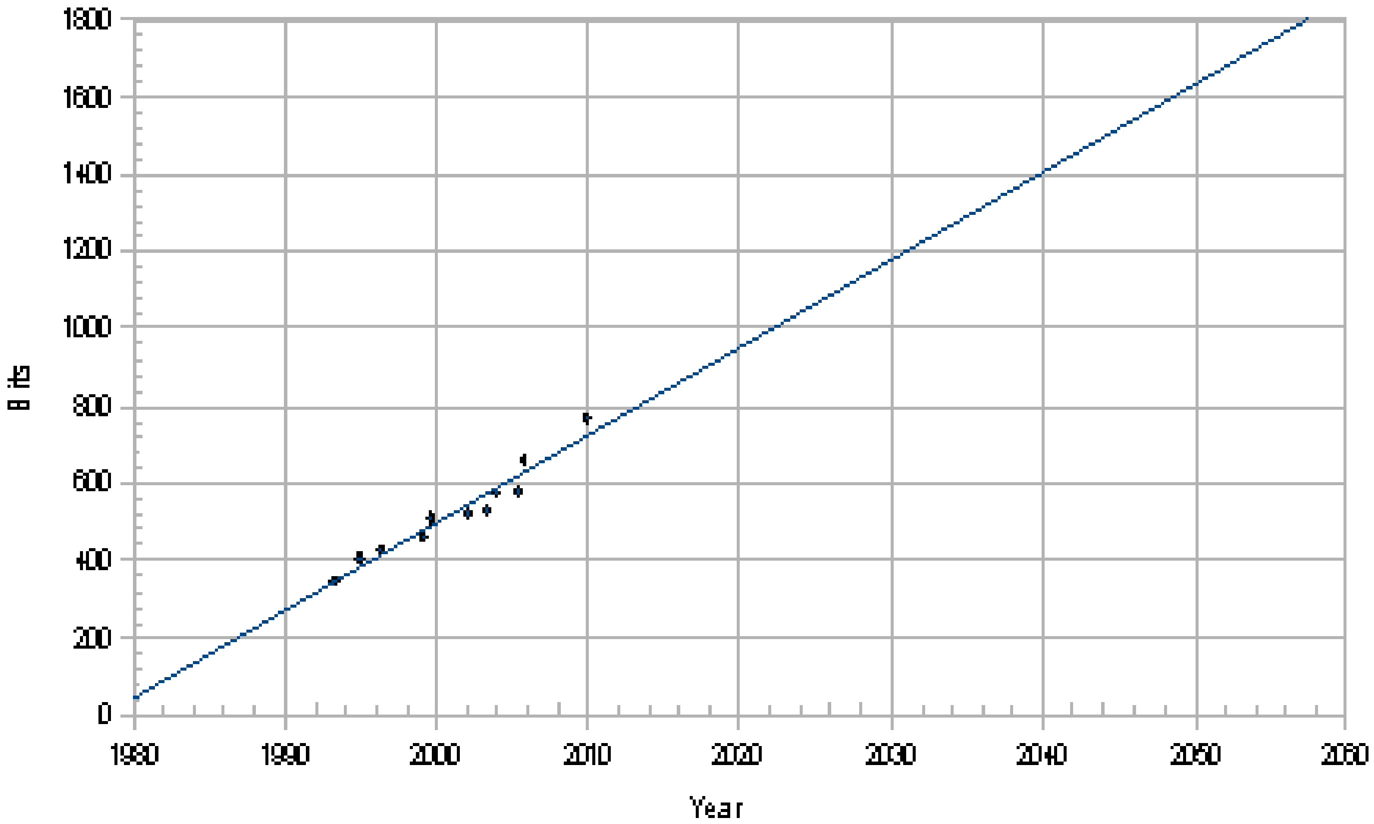
- RSA is considered secure
  - But the key size does matter
- 1977: published in “Scientific American”
  - RSA-129 (129 decimal digits of modulus n)
  - Challenge of 100 dollars
  - 40 quadrillion years estimated to factor …
  - Factored in 1994
    - “The magic words are squeamish ossifrage.”

History of RSA Security II

- 1999
  - 512 bit integer was factorized
- 2005
  - 663 bit integer was factorized
- January 2010
  - 768 bit integer was factorized

- 1024 bit integers are (probably) factorable at the moment by large organizations

Security of RSA



Source: P. Layland, RSA Security and Integer Factorization: The Thirty Years War from 1990 to 2020, IS2 2010, Praha

Key size

- Algorithms are public & keys must be secret
- Key must be large enough that a brute force attack is infeasible
- Depending on the algorithm used it is common to have different key sizes for the same level of security
  - Representing the level of security – number of combinations needed for the brute force attack
  - E.g. 1024 bit RSA key equivalent to 80 bit symmetric encryption key

Comparable strengths of cryptosystems

Security Strength	Symmetric key algorithms	FFC (e.g., DSA, D-H)	IFC (e.g., RSA)	ECC (e.g., ECDSA)
≤ 80	2TDEA[21]	$L = 1024$ $N = 160$	$k = 1024$	$f = 160\text{-}223$
112	3TDEA	$L = 2048$ $N = 224$	$k = 2048$	$f = 224\text{-}255$
128	AES-128	$L = 3072$ $N = 256$	$k = 3072$	$f = 256\text{-}383$
192	AES-192	$L = 7680$ $N = 384$	$k = 7680$	$f = 384\text{-}511$
256	AES-256	$L = 15360$ $N = 512$	$k = 15360$	$f = 512+$

Source:
NIST S 7

Security strengths of hash functions

Security Strength	Digital Signatures and hash-only applications	HMAC[22], Key Derivation Functions[23], Random Number Generation[24]
$\leq$ 80	SHA-1[25]	
112	SHA-224, SHA-512/224, SHA3-224	
128	SHA-256, SHA-512/256, SHA3-256	SHA-1
192	SHA-384, SHA3-384	SHA-224, SHA-512/224
$\geq$ 256	SHA-512, SHA3-512	SHA-256, SHA-512/256, SHA-384, SHA-512, SHA3-512

Source:
NIST SP8

Recommended key sizes

Security Strength		Through 2030	2031 and Beyond
< 112	Applying	Disallowed	
	Processing	Legacy-use	
112	Applying	Acceptable	Disallowed
	Processing		Legacy use

Security Strength		Through 2030	2031 and Beyond
128	Applying/Processing	Acceptable	Acceptable
192		Acceptable	Acceptable
256		Acceptable	Acceptable

Source:
NIST SP8

Recommended key sizes

"Acceptable" indicates that the algorithm or key length is not known to be insecure.

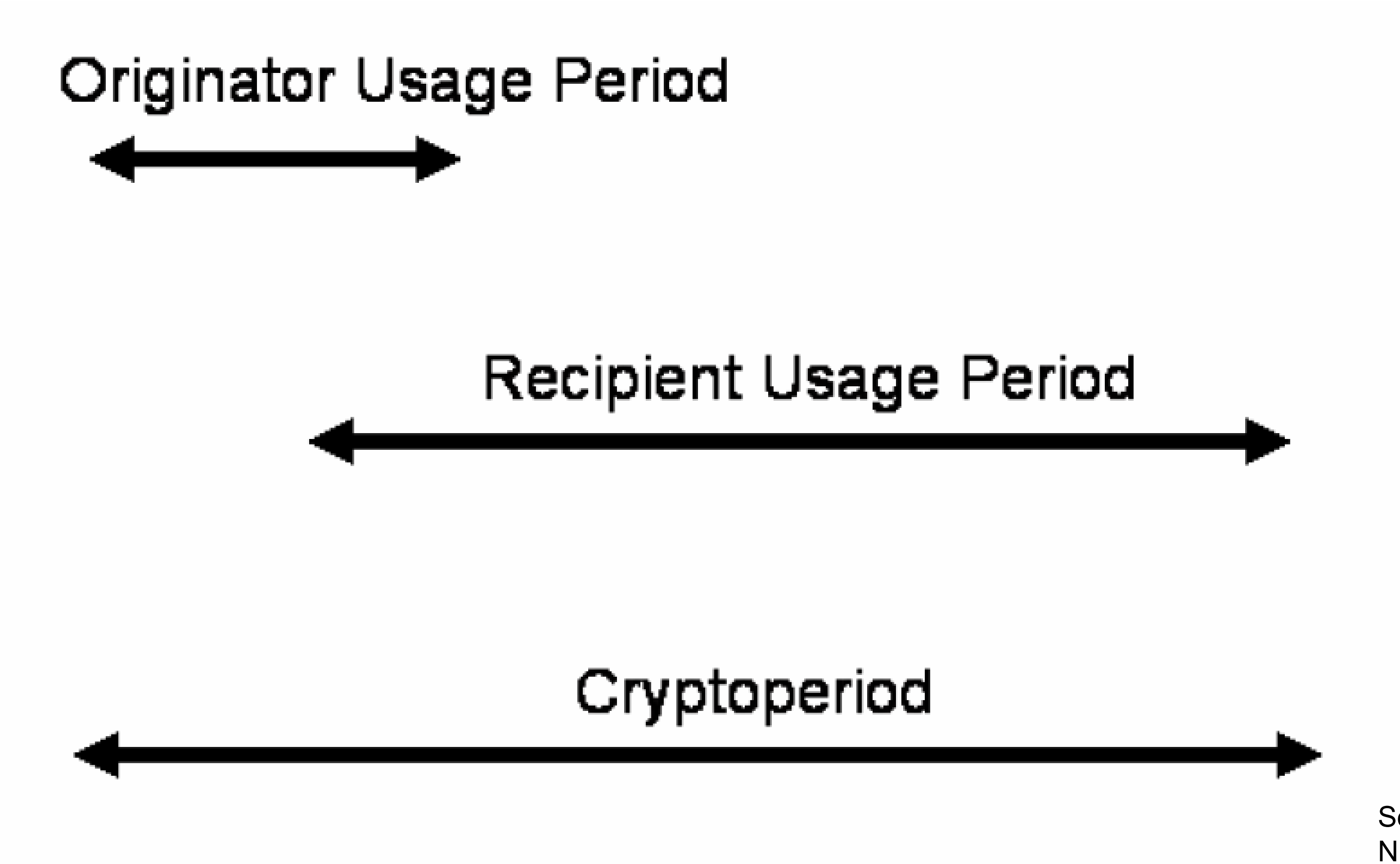
"Deprecated" means that the use of an algorithm or key length that provides the indicated security strength may be used if risk is accepted

"Legacy use" means that an algorithm or key length may be used because of its use in legacy applications

"Disallowed" means that an algorithm or key length **shall not be used for** applying cryptographic protection.

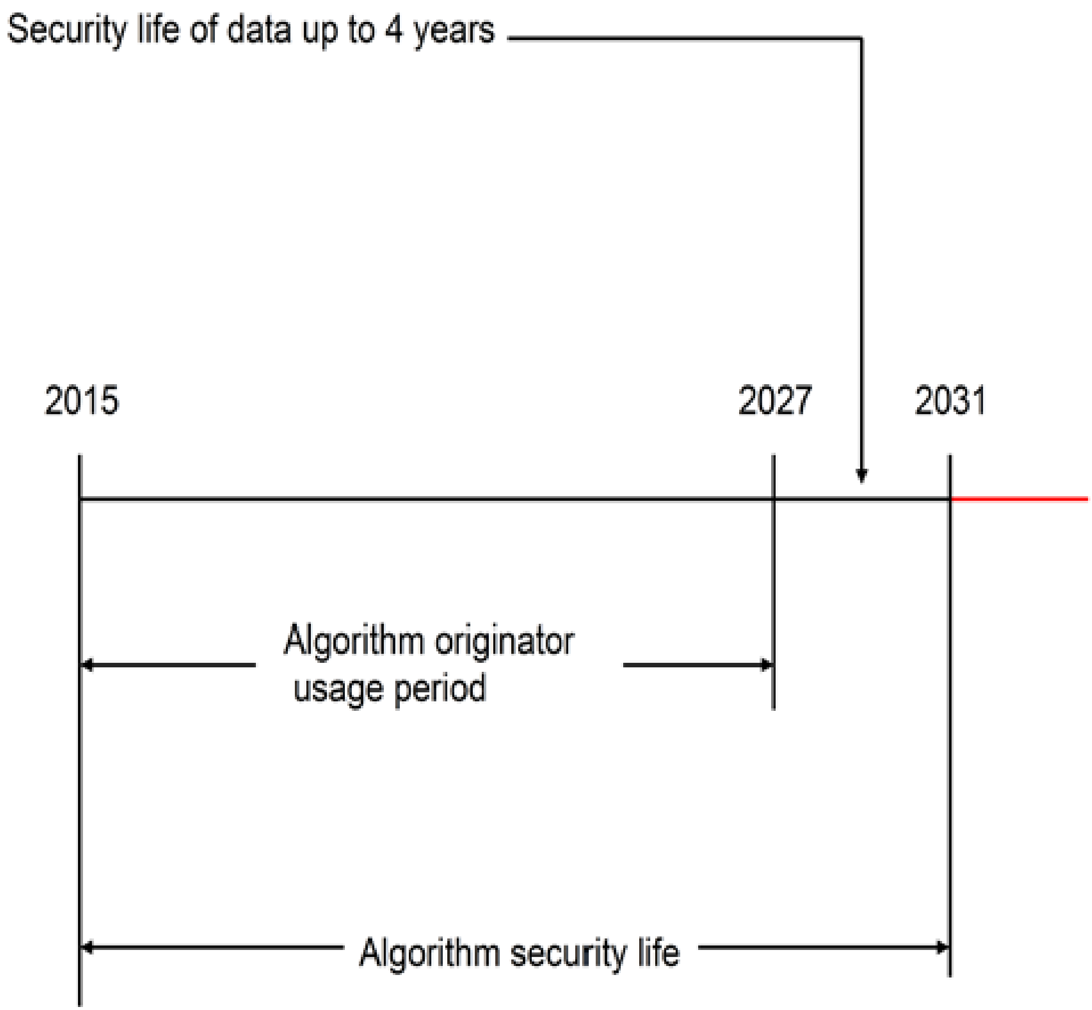
Source:
NIST SP800-57

Crypto period



Source:
NIST SP8

Crypto period example



Source:
NIST SP8(

Recommended crypto periods

Key Type	Crytoperiod	
	Originator-Usage Period (OUP)	Recipient-Usage Period
1. Private Signature Key	1 to 3 years	–
2. Public Signature-Verification Key	Several years (depends on key size)	
3. Symmetric Authentication Key	$\le$ 2 years	$\le$ OUP + 3 years
4. Private Authentication Key	1 to 2 years	
5. Public Authentication Key	1 to 2 years	
6. Symmetric Data Encryption Keys	$\le$ 2 years	$\le$ OUP + 3 years
7. Symmetric Key Wrapping Key	$\le$ 2 years	$\le$ OUP + 3 years
8. Symmetric RBG Keys	See [SP800-90]	–
9. Symmetric Master Key	About 1 year	–
10. Private Key Transport Key	$\le$ 2 years[16]	
11. Public Key Transport Key	1 to 2 years	
12. Symmetric Key Agreement Key	1 to 2 years[17]	
13. Private Static Key Agreement Key	1 to 2 years[18]	
14. Public Static Key Agreement Key	1 to 2 years	
15. Private Ephemeral Key Agreement Key	One key-agreement transaction	
16. Public Ephemeral Key Agreement Key	One key-agreement transaction	

Source:NIST SP800-57

Recommended crypto periods

Key Type	Crytoperiod	
	Originator-Usage Period (OUP)	Recipient-Usage Period
17. Symmetric Authorization Key	$\leq$ 2 years	
18. Private Authorization Key	$\leq$ 2 years	
19. Public Authorization Key	$\leq$ 2 years	

Source:NIST SP800-57

ETSI recommendation (RSA)

Table 6: Recommended parameters for RSA for a resistance during X years

Parameter	1 year	3 years	6 years
Key size ($\log_2(n)$)	≥ 1 900	≥ 1 900	≥ 3 000

- Source: ETSI TS 119 312 V1.3.1 (2019-02)
- Recommended key sizes for RSA for a resistance during X years
- Starting date: 2019

ETSI recommendation (DSA)

Parameter	1 year	3 years	6 years
pLen	2 048	2 048	3 072
qLen	224 or 256	224 or 256	256

- Source: ETSI TS 119 312 V1.3.1 (2019-02)
- Recommended key sizes for DSA
- Starting date: 2019

ETSI recommendation (ECDSA)

Table 8: Recommended parameters for EC-DSA and EC-SDSA-opt for a resistance during X years

Parameter	1 year	3 years	6 years
pLen = *qLen*	256, 384 or 512	256, 384 or 512	256, 384 or 512

- Source: ETSI TS 119 312 V1.3.1 (2019-02)
- Recommended key sizes for ECDSA
- Starting date: 2019

ETSI recommendation (hash functions)

Entry name of the hash function	1 year	3 years	6 years
SHA-224	usable	usable	unusable
SHA-256	usable	usable	usable
SHA-384	usable	usable	usable
SHA-512	usable	usable	usable
SHA3-256	usable	usable	usable
SHA3-384	usable	usable	usable
SHA3-512	usable	usable	usable

- Source: ETSI TS 119 312 V1.3.1 (2019-02)
- Recommended hash functions
- Starting date: 2019

ETSI recommendation

Entry name of the signature suite	1 year	3 years	6 years
sha256-with-rsa	≥ 1 900	≥ 1 900	not recommended
sha512-with-rsa	≥ 1 900	≥ 1 900	not recommended
rsa-pss with mgf1SHA-256Identifier	≥ 1 900	≥ 1 900	≥ 3 000
rsa-pss with mgf1SHA-512Identifier	≥ 1 900	≥ 1 900	≥ 3 000
rsa-pss with mgf1SHA3-Identifier	≥ 1 900	≥ 1 900	≥ 3 000
sha256-with-dsa	2 048	2 048	3 072
sha512-with-dsa	2 048	2 048	3 072
sha224-with-ecdsa	legacy		not recommended
sha2-with-ecdsa	recommended		
sha2-with-ecsdsa	recommended		
sha3-with-ecdsa	recommended		
sha3-with-ecsdsa	recommended		

- Source: ETSI TS 119 312 V1.3.1 (2019-02)
- Recommended signature suites
- Starting date: 2019

ICAO recommendation

- International Civil Aviation Organization
  - Electronic passports
  - Data signed by the issuing country to protect integrity
  - One CA per country, certificates issued for entities producing passports (so called Document Signers).
  - Standard validity of passports: 10 years

ICAO recommendations

- RSA (UK, CZ, France, …)
  - Padding: PKCS#1 v1.5, PSS (recommended)
  - For CA: min 3072 bits
  - For DS: min 2048 bits
- DSA
  - For CA: min 3072/256 bits
  - For DS: min 2048/224 bits
- ECDSA (Germany, Switzerland, …)
  - For CA: min 256 bits
  - For DS: min 224 bits
- Hash functions
  - ~~SHA-1~~, SHA-2

Source: ICAO Doc.

ICAO recommendations

- “It is therefore RECOMMENDED that the maximum period the Document Signer Key is used to sign passport documents be three months. For States that generate large numbers of MRTDs, several current document signing keys MAY be issued at any given time.”

Source: ICAO Doc.