

TRANSITIONING TO A QUANTUM-RESISTENT PUBLIC KEY INFRASTRUCTURE

PQCrypto 2017
Utrecht, The Netherlands
06/26/2017

Nina Bindel

Udyani Herath

Matthew McKague

Douglas Stebila





$\frac{1}{7}$ chance of breaking RSA-2048
(Michele Mosca – Nov 2015)

$\frac{1}{2}$ chance of breaking RSA-2048
(Michele Mosca – Nov 2015)

Start
NIST
PQ project

Universal quantum computer
(Quantum Manifesto)



MS started to stop support of SHA-1

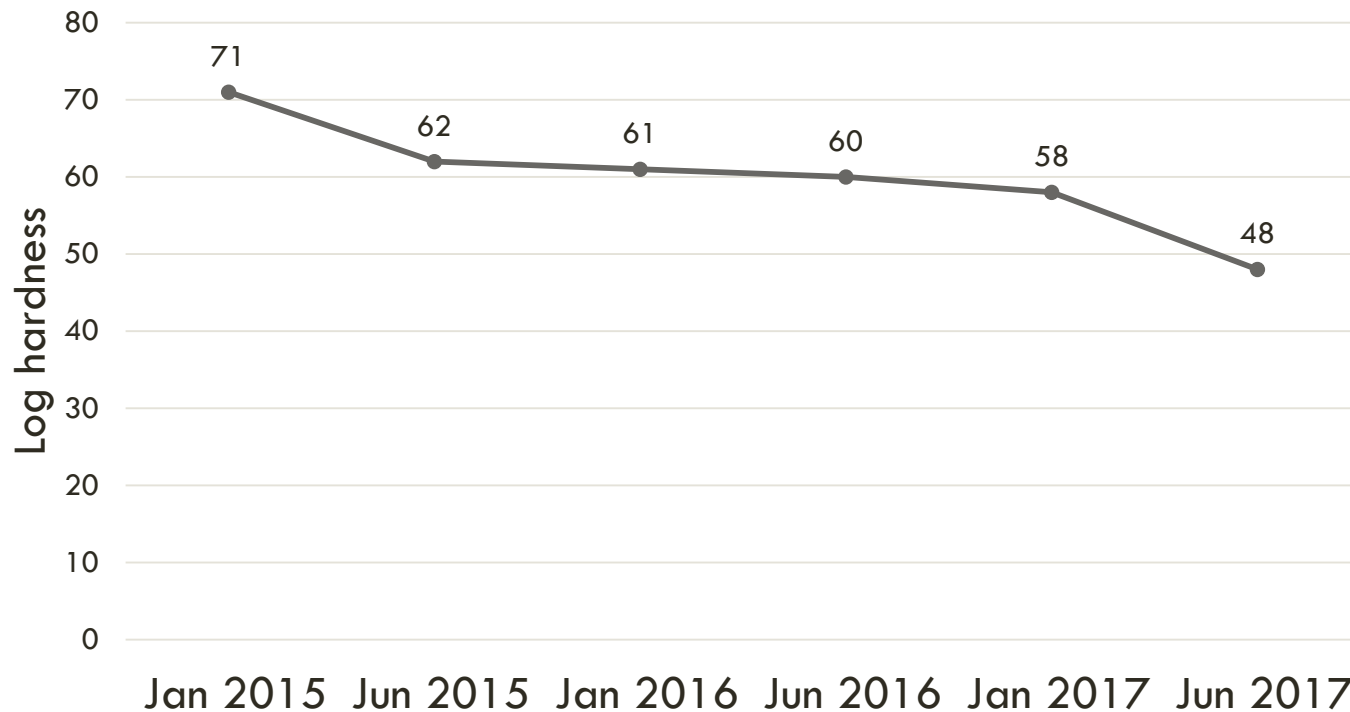
15 years

18 years

Best: start transition now ?

BIT-HARDNESS ESTIMATIONS WITH LWE-ESTIMATOR

[APS15]



Difference of
~20 bit in 2.5 years

LWE Instance - Regev(128)
 $n=128, q=16411, \sigma=29.6$

CURRENT SITUATION

Quantum threat against
RSA- and discrete log

Unstable hardness
estimations of “PQ
assumptions”

HYBRID SIGNATURE SCHEMES

Given: Σ_1 and Σ_2

Construct: Σ_C s.t. Σ_C is secure if Σ_1 or Σ_2 secure

Example:

- Σ_1 PQ scheme and Σ_2 classical scheme
- 2 PQ schemes based on different assumptions



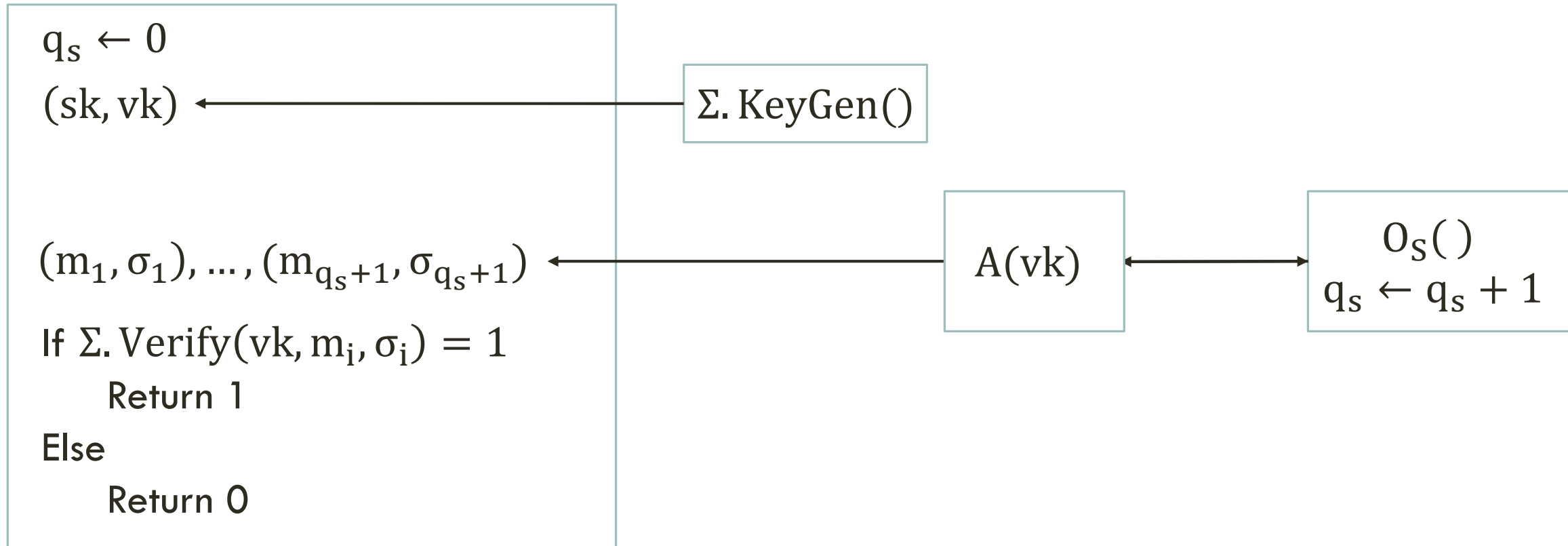
- What means “secure“?
- How to construct Σ_C ?
- Can we use hybrids in current protocols and standards?

SECURITY DEFINITION

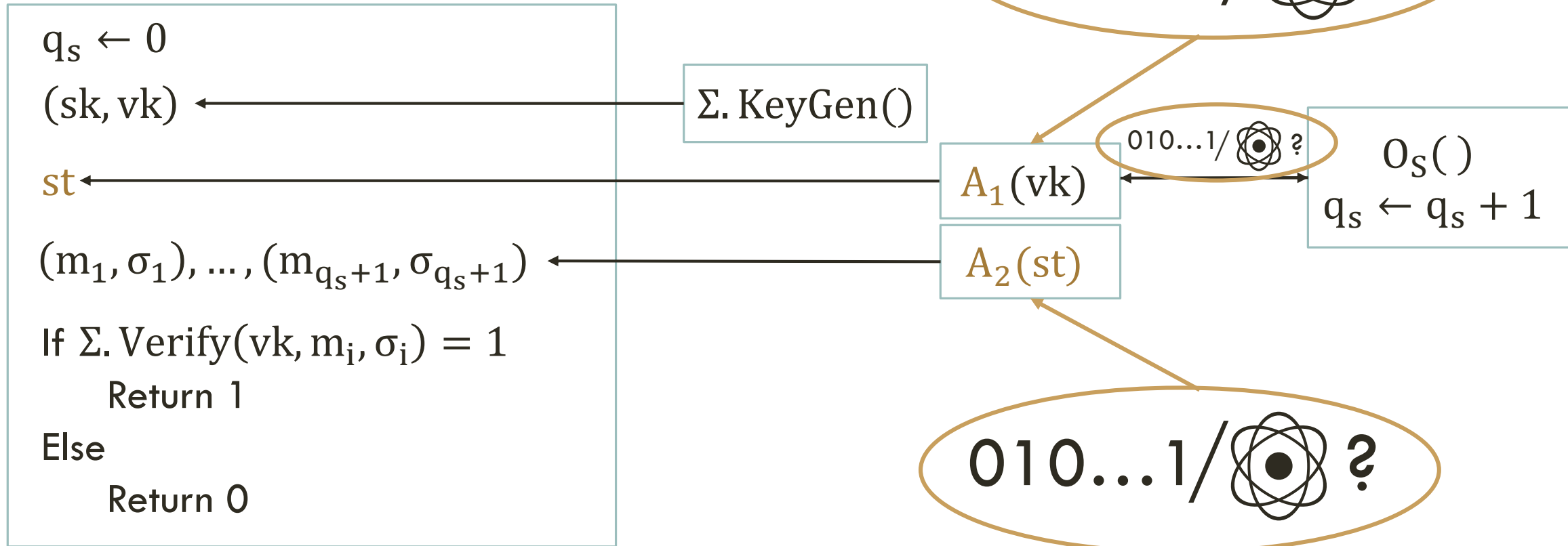
Intuition:

- eUF-CMA with 2-stage adversary $A = (A_1, A_2)$
- A_1, A_2 different access to quantum computer
- A_1 classical/quantum access to sign oracle

$\text{EXPT}_{\Sigma}^{\text{EUF-CMA}}(A)$:



EXPT $_{\Sigma}^{\text{EUF-CMA}}$ (A_1, A_2) :





ADVERSARY MODEL

C^cC - Fully classical (eUF-CMA)

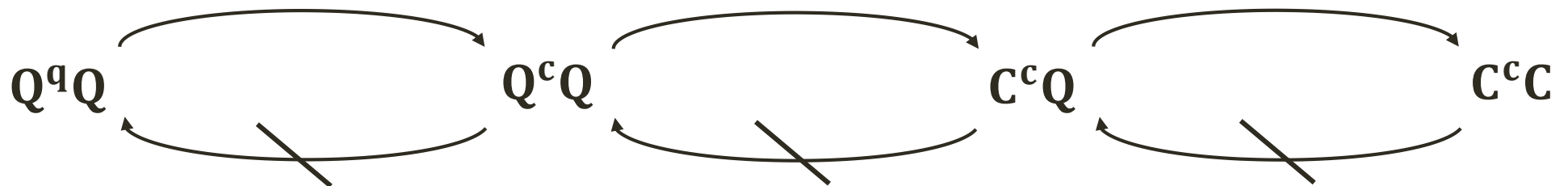
C^cQ - Future quantum

Q^cQ - Quantum adversary

Q^qQ - Fully quantum (also in [BZ13]) ←

- A_1 : 
- A_2 : 
- Access O_S : 

THEOREM



EXAMPLES OF HYBRID SIGNATURES

Σ_1 XYZ-secure
 Σ_2 UVW-secure

Combiner	$\sigma = (\sigma_1, \sigma_2)$	Unforgeability
$C_{ }$	$\sigma_1 \leftarrow \text{Sign}_1(m)$ $\sigma_2 \leftarrow \text{Sign}_2(m)$	$\max\{XYZ, UVW\}$
C_{nest}	$\sigma_1 \leftarrow \text{Sign}_1(m)$ $\sigma_2 \leftarrow \text{Sign}_2(m, \sigma_1)$	$\max\{XYZ, UVW\}$
$C_{\text{dual-nest}}$	$\sigma_1 \leftarrow \text{Sign}_1(m_1)$ $\sigma_2 \leftarrow \text{Sign}_2(m_1, \sigma_1, m_2)$	XYZ wrt to m_1 , UVW

APPLICABLE TO CURRENT PKI?

- Certificates: X.509v3
- Secure channels: TLS
- Secure email: S/MIME



- (1) How can hybrid combiners be used in current standards?
- (2) What about backwards-compatibility?
- (3) Do large key and signature size raise problems?

HYBRID SIGNATURE IN S/MIME EMAIL

Idea:

- Use concatenation combiner
- S/MIME data structures allow multiple parallel signatures
- Disadvantage: Verification of all signatures
 - backwards-compatibility?

2nd Idea:

- Use nested combiner
- Use optional attributes

HYBRID SIGNATURES IN X.509V3 CERT

Idea:

- Use dual nested combiner
- **PQ** cert = extension of **RSA** cert
- Hybrid software recognizes and processes PQ cert **and** RSA cert
- Older software ignores non-critical ext.

$(sk_{PQ}^{CA}, vk_{PQ}^{CA}), (sk_{RSA}^{CA}, vk_{RSA}^{CA}) \leftarrow \text{KeyGen}_{\text{dual-nest}}$

$(sk_{PQ}^{\text{Sub}}, vk_{PQ}^{\text{Sub}}), (sk_{RSA}^{\text{Sub}}, vk_{RSA}^{\text{Sub}}) \leftarrow \text{KeyGen}_{\text{dual-nest}}$

Certificate c_2 (RSA)

tbsCertificate m_2 :

CA, subject, vk_{RSA}^{Sub}

$c_2 = \text{Sign}_{\text{RSA}}(sk_{RSA}^{CA}, (m_2, vk_{RSA}^{\text{Sub}}, c_1, m_1))$

Extensions:

Ext. id. = non-critical

Certificate c_1 (PQ)

tbsCertificate m_1 :

CA, subject, vk_{PQ}^{Sub}

$c_1 = \text{Sign}_{\text{PQ}}(sk_{PQ}^{CA}, (m_1, vk_{PQ}^{\text{Sub}}))$

COMPATIBILITY OF HYBRID X.509V3 CERTS

	Application	Extension size [KB]				
		1.5	3.5	9.0	43.0	1333.0
Libraries	GnuTLS	✓	✓	✓	✓	✗
	Java SE	✓	✓	✓	✓	✓
	mbedTLS	✓	✓	✓	✗	✗
	NSS	✓	✓	✓	✓	✗
	OpenSSL	✓	✓	✓	✓	✗
Web browsers	Apple Safari	✓	✓	✓	✓	✓
	Google Chrome	✓	✓	✓	✓	✗
	MS Edge	✓	✓	✓	✗	✗
	MS IE	✓	✓	✓	✗	✗
	Mozilla Firefox	✓	✓	✓	✓	✗
	Opera	✓	✓	✓	✓	✗

SUMMARY

- Security experiment with 2-stage adversary
- Adversary model with respect to quantum power
- Construction of hybrid signature schemes
- Compatibility of with current PKI:
 - Nested single message in S/MIME
 - Nested dual message in X.509 cert in applications
- Left out: non-separability