

Post-quantum Cryptography

Considera il potenziale impatto dei computer quantistici
sulla sicurezza dei sistemi

Teorico: dalla metà degli anni '90 impatto noto

Concreto: computer quantistici large-scale general purpose
ancora inesistenti (A.D. 2024)

Post-quantum Cryptography

Intenso sforzo volto al progetto di criptosistemi
"post-quantistici" che possono restare sicuri anche contro
(poly-time) algoritmi quantistici

- Call per algoritmi post-quantistici
a chiave pubblica
- Pubblicato ad Agosto 2024

Post-quantum

cryptography



sistemi interamente
classici ma
progettati per resistere ad
adversari che dispongono
di computer e algoritmi
quantistici

~~F~~

Quantum

cryptography



sistemi che sono
implementati usando
computer quantistici,
fenomeni della meccanica
quantistica e canali di
comunicazione quantistici

Cifografia simmetrica Post-quantistica

(più o meno salva ...)

Algoritmo di Grover e lunghezza delle chiavi

Problema: Adhv ha accesso oracolare a $f: D \rightarrow \{0, 1\}$
Deve trovare un $x \in D$ tale che $f(x) = 1$.

Se ne esiste uno soltanto scelto unif. a caso
un algoritmo classico richiede $O(|D|)$
query (ricerca esauriva su D)

Algoritmo di Grover

1996 - Grover mostrò un algoritmo quantistico che trova x con $O(|D|^{1/2})$ query
(speed-up quadratico)

Successivamente è stato provato ottimale.

Impatto su un cifrario a blocchi

$$F : \{0,1\}^n \times \{0,1\}^l \rightarrow \{0,1\}^l \quad y_i = F_n(x_i)$$

Aolv cerca $K \in \{0,1\}^n$ dato un # costante di coppie $\{(x_i, y_i)\}_i$.

Problema: $f(K) = 1 \iff F_n(x_i) = y_i$ per ogni i .

Alg. classico (ricerca esauritiva) $O(2^n)$ passi

Alg. Grover quantistico $O(2^{n/2})$ passi

Implicazione: per avere un livello di sicurezza $P = n$ bit

- mondo classico, $|K| = \underline{n}$ bit lunghezza
- mondo quantistico, $|K| = \underline{2 \cdot n}$ bit della chiave

Nota: assumendo che gli attacchi di ricerca esauritiva siano il meglio che possiamo fare contro il critosistema

Funzioni hash

$$h : \{0,1\}^* \rightarrow \{0,1\}^n$$

Mondo classico : attacco del complotto $O(2^{n/2})$

Mondo quantistico : alg. di Grover $O(2^{n/3})$

Implicazione : per ottenere un livello di sicurezza $p = n$ bit

- mondo classico : $\{0,1\}^{\underline{2n}}$ \leftarrow lunghezza del

- mondo quantistico : $\{0,1\}^{\underline{3n}}$ digest

↪ Perdi ? Seguono dettagli ...

Sia $l < n$, $C \in D$ sottoinsiemi di $\{0,1\}^n$

$$|C|=e \quad |D|=l^2$$

Aolv: per ogni $x_i \in C$, calcola $H(x_i) = y_i$. Sia $C' = \{y_i\}_{i=1}^e$

(Se $\exists i, j : y_i = y_j$, collisione trovata)

Altimenti, sia $g: D \rightarrow \{0,1\}$ tale che

$$g(x) = 1 \iff H(x) \in C'$$

L'algor. di browser richiede $O(|D|^{1/2})$ valutazioni di H

Valutazioni totali: $O(l + \sqrt{l^2}) = O(l)$

$$\begin{array}{c} \downarrow \quad \downarrow \\ |C| \quad |D| \end{array}$$

Nota de:

$$\text{Prob} (x \in D : H(x) \in C') = \frac{\ell}{2^n}$$

$$\text{Prob} (x \in D : H(x) \notin C') = 1 - \frac{\ell}{2^n}$$

$$\text{Prob} (\forall x \in D : H(x) \notin C') = \left(1 - \frac{\ell}{2^n}\right)^{\ell^2} \quad (1-x) \leq e^{-x}$$

$$\text{Prob} (\exists x \in D : H(x) \in C') = \left(1 - \left(1 - \frac{\ell}{2^n}\right)^{\ell^2}\right) \geq 1 - e^{-\frac{\ell^3}{2^n}}$$

$$\text{Per } \ell = O(2^{n/3}) \Rightarrow 1 - e^{-\frac{\ell^3}{2^n}} = 1 - e^{-\frac{(2^{4/3})^3}{2^n}} = 1 - \frac{1}{e}$$

Conclusioni: (prob. collisione costante)

Hash sicuro inietto ad Adv quantistici richiede
una lunghezza del 50% in più iniett. all hash classico

Allo stato attuale delle conoscenze

i microsistemi simmetrici sono

essenzialmente salvi nel mondo

post-quantistico.

(Dicembre 2024)

Criptografia a chiavi pubblica

Algoritmo di Shor : speed-up esponenziale nella risoluzione
di alcuni problemi della teoria dei numeri

↳ (factoring, discrete log)

Poly-time quantum algorithms !

Problema astratto : $f : \mathbb{H} \rightarrow \mathbb{R}$ periodica
↳ gruppo abeliano $\exists \delta \in \mathbb{H}$ tale che
 $f(x) = f(x + \delta)$

Calcolare il periodo δ

Nota : * f è periodica di periodo δ , lo è anche di $2\delta, 3\delta \dots$ etc.

Nel mondo classico non si conoscono algoritmi efficienti per il calcolo del periodo f dato soltanto accesso oracolare a f
↳ (ed anche la verifica...)

Nel 1994 Shor mostrò come farlo in poly-time usando un algoritmo quantistico, per certi gruppi \mathbb{H}

Implicazioni: Factoring

$N = p \cdot q$ (primi grandi). Per ogni x , $f_{x,N} : \mathbb{Z} \rightarrow \mathbb{Z}_N^*$

definita come $f_{x,N}(z) = [x^z \bmod N]$

è PERIODICA di periodo $\varphi(N)$.

Precisamente, $\forall z \in \mathbb{Z}$

$$g_{x,N}(z + \varphi(N)) = [x^{2+\varphi(N)} \bmod N] = [x^2 \cdot x^{\varphi(N)} \bmod N] = [x^2 \bmod N]$$

$\underbrace{\hspace{1cm}}$
1 (Te Euler)

Pertanto, per qualsiasi $x \in \mathbb{Z}_N^*$, possiamo

- usare l'algoritmo di Shor per calcolare $\varphi(N)$
- usare $\varphi(N)$ per fattorizzare N (alg. classico)

L'algoritmo di Shor per il calcolo del periodo può essere usato anche per il calcolo del logaritmo discreto

(G, \cdot, g) gruppo ciclico, g generatore. Sia $h \in G$

\hookrightarrow ordine, primo

Sia $f_{g,h} : \mathbb{Z}_q \times \mathbb{Z}_q \rightarrow G$ definita come segue:

$$f_{g,h}(a, b) = g^a \cdot h^{-b}$$

Per $x = \log_g h$, $f(x, b) = g^x \cdot h^{-b} = g^{\log_g h} \cdot h^{-b} = h \cdot h^{-b}$ $\hookrightarrow^{b=1} h \cdot h^{-1} = 1$

Pertanto $(x, 1)$ è periodo per $f_{g,h}$. Infatti, $\forall a, b$:

$$f_{g,h}(a+x, b+1) = g^{a+x} \cdot h^{-(b+1)} = g^a \cdot g^x \cdot h^{-b} \cdot h^{-1} = g^a \cdot h^{-b} \cdot g^x \cdot h^{-1} = g^a h^{-b} = f_{g,h}(a, b)$$

$\boxed{1}$

Inoltre sappiamo che per ogni periodo (x^i, y^i) risulta

$$g^{x^i} \cdot h^{-y^i} = 1 = g^0 \cdot h^0$$

x^i y^i

Applicando gli stessi passi usati nella prova di sicurezza della funzione hash resistente a collisione (lesioni teoria) sui numeri si calcola il logaritmo discreto di h .

Conclusione: tutti gli schemi a chiave pubblica basati su Factoring e Discrete log, soccombono nel mondo post quantistico.

Scenari a chiavi pubbliche post-quantistici

Notazione : $\lfloor x \rfloor$ più grande intero $\leq x$

$\mathbb{Z}_q = \{0, \dots, q-1\}$ lo vediamo come $\left\{ -\lfloor \frac{(q-1)/2}{q} \rfloor, \dots, 0, \dots, \lfloor \frac{q-1}{2} \rfloor \right\}$

Elemento "piccolo": vicino a zero

Problema LWE (learning with errors)

$$B \in \mathbb{Z}_q^{m \times m}, \quad s \in \mathbb{Z}_q^m.$$

Fatto! Basta
usare l'algebra
lineare!

Dati B e $t = B \cdot s \bmod q$, trovare $s' \in \mathbb{Z}_q^m$: $B \cdot s' \equiv t \bmod q$

Sia e (vettore degli errori) $\in \mathbb{Z}_q^m$

norma
euclidea

↪ vettore corto $e = (e_1, \dots, e_m)$, $\|e\| = \sqrt{\sum_i e_i^2}$

lunghezza

$$t = (B \cdot s + e) \bmod q$$

Dati B e t trovare s' tale che $[t - Bs' \bmod q]$ è corto

Quando i parametri vengono scelti opportunamente il problema sembra essere difficile, anche per algoritmi quantistici.

Versione decisionale del problema LWE

Distinguere t generato dal processo precedente da $t \in_{\beta} \mathbb{Z}_q^m$

↑
Uniforme

DEFINITION 14.1 We say the decisional $\text{LWE}_{m,q,\psi}$ problem is quantum-hard if for all quantum polynomial-time algorithms \mathcal{A} there is a negligible function negl such that

$$\left| \Pr[\mathbf{B} \leftarrow \mathbb{Z}_q^{m \times n}; \mathbf{s} \leftarrow \psi^n; \mathbf{e} \leftarrow \psi^m : \mathcal{A}(\mathbf{B}, [\mathbf{Bs} + \mathbf{e} \bmod q]) = 1] - \Pr[\mathbf{B} \leftarrow \mathbb{Z}_q^{m \times n}; \mathbf{t} \leftarrow \mathbb{Z}_q^m : \mathcal{A}(\mathbf{B}, \mathbf{t}) = 1] \right| \leq \text{negl}(n).$$

Come usare il problema per progettare
uno schema di crittografia a chiave pubblica?

Idea: scambio Diffie-Hellman like (non sicuro)

Alice

$$B \in \mathbb{Z}_q^{m \times n}$$

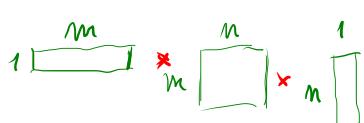
$$s \in \mathbb{Z}_q^m$$

$$t_A = B \cdot s \bmod q$$

$$K_A = [t_B^T \cdot s \bmod q]$$

$$(B, t_A) \rightarrow$$

$$\leftarrow t_B$$



$$K_A = E_B^T \cdot s = \hat{s}^T \cdot \underbrace{B \cdot s}_{\text{L}} = \hat{s}^T \cdot t_A = K_B \quad (\text{stessa chiave})$$

Bob

$$\hat{s} \in \mathbb{Z}_q^m$$

trasposti

$$E_B^T = \hat{s}^T \cdot B \bmod q$$

$$K_B = [\hat{s}^T \cdot t_A \bmod q]$$

Naturalmente lo scerma non è nero perdi un AdL
in ascolto può usare l'algebra lineare per calcolare
 s o \hat{s} (o entrambi) e recuperare la stessa chiave!

Aggiungendo opportunamente errori (noise/rumore)
e assumendo la difficoltà del problema LWE
decisionale, è possibile costruire uno schema di
cifratura a chiave pubblica El-Gamal style CPA-sicuro
(anche rispetto ad AdL quantistici!)

CONSTRUCTION 14.3

Let m, q, ψ be as in the text. Define a public-key encryption scheme as follows:

- **Gen:** on input 1^n choose uniform $\mathbf{B} \leftarrow \mathbb{Z}_q^{m \times n}$ as well as $\mathbf{s} \leftarrow \psi^n$ and $\mathbf{e} \leftarrow \psi^m$. Set $\mathbf{t} := [\mathbf{B} \cdot \mathbf{s} + \mathbf{e} \bmod q]$. The public key is $\langle \mathbf{B}, \mathbf{t} \rangle$ and the private key is \mathbf{s} .
- **Enc:** on input a public key $pk = \langle \mathbf{B}, \mathbf{t} \rangle$ and a bit b , choose $\hat{\mathbf{s}} \leftarrow \psi^m$ and $\hat{\mathbf{e}} \leftarrow \psi^{n+1}$, and output the ciphertext

$$\mathbf{c}^T := \left[\hat{\mathbf{s}}^T \cdot [\mathbf{B} \mid \mathbf{t}] + \hat{\mathbf{e}}^T + \underbrace{[0, \dots, 0, b \cdot \lfloor \frac{q}{2} \rfloor]}_{n+1} \bmod q \right].$$

- **Dec:** on input a private key \mathbf{s} and a ciphertext \mathbf{c}^T , first compute $k := [\mathbf{c}^T \cdot \begin{bmatrix} -\mathbf{s} \\ 1 \end{bmatrix} \bmod q]$. Then output 1 if k is closer to $\lfloor \frac{q}{2} \rfloor$ than to 0 (see text), and 0 otherwise.

Lo schema permette di cifrare un messaggio di 1 bit b

La decifratura potrebbe essere scorretta (prob. errore)

ma per parametri opp. saliti è corretta con alta probabilità.

$$c^T = \hat{s}^T \cdot [B|t] + \hat{e}^T + b^T \quad b^T = [0, 0, \dots, 0, b, \left\lfloor \frac{q}{2} \right\rfloor]$$

$$\kappa = c^T \cdot \begin{bmatrix} -s \\ 1 \end{bmatrix}$$

$$= (\hat{s}^T \cdot [B|t] + \hat{e}^T + b^T) \cdot \begin{bmatrix} -s \\ 1 \end{bmatrix} = -\hat{s}^T B \cdot s + \hat{s}^T t + \hat{e}^T \begin{bmatrix} -s \\ 1 \end{bmatrix} + b \left\lfloor \frac{q}{2} \right\rfloor$$

$$= \hat{s}^T e + \hat{e}^T \cdot \begin{bmatrix} -s \\ 1 \end{bmatrix} + b \left\lfloor \frac{q}{2} \right\rfloor \quad (\text{usando } t = B \cdot s + e)$$

Fino a quando $|\hat{s}^T e + \hat{e}^T \cdot \begin{bmatrix} -s \\ 1 \end{bmatrix}| < \frac{(q-1)}{4}$ la decifratura
è (a' \boxed{b} correttamente)

Firme digitali tramite funzioni hash

Può sembrare sorprendente ma schemi di firme digitali possono essere ottenuti usando funzioni hash crittografiche, senza la necessità di assunzioni di teoria dei numeri.

Schemi di Lamport: sicuri per un uso singolo (one-time signature)

Signing $m = 011$:

$$sk = \begin{pmatrix} x_{1,0} & x_{2,0} & x_{3,0} \\ x_{1,1} & \boxed{x_{2,1}} & \boxed{x_{3,1}} \end{pmatrix} \Rightarrow \sigma = (x_{1,0}, x_{2,1}, x_{3,1})$$

Verifying for $m = 011$ and $\sigma = (x_1, x_2, x_3)$:

$$pk = \left(\begin{array}{ccc} y_{1,0} & y_{2,0} & y_{3,0} \\ y_{1,1} & \boxed{y_{2,1}} & \boxed{y_{3,1}} \end{array} \right) \right\} \Rightarrow \begin{aligned} H(x_1) &\stackrel{?}{=} y_{1,0} \\ H(x_2) &\stackrel{?}{=} y_{2,1} \\ H(x_3) &\stackrel{?}{=} y_{3,1} \end{aligned}$$

FIGURE 12.3: The Lamport scheme used to sign the message $m = 011$.

One-time signature

The one-time signature experiment $\text{Sig-forge}_{\mathcal{A}, \Pi}^{\text{1-time}}(n)$:

1. $\text{Gen}(1^n)$ is run to obtain keys (pk, sk) .
2. Adversary \mathcal{A} is given pk and asks a single query m' to its oracle $\text{Sign}_{sk}(\cdot)$. \mathcal{A} then outputs (m, σ) with $m \neq m'$.
3. The output of the experiment is defined to be 1 if and only if $\text{Vrfy}_{pk}(m, \sigma) = 1$.

DEFINITION 12.14 Signature scheme $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ is existentially unforgeable under a single-message attack, or is a one-time-secure signature scheme, if for all probabilistic polynomial-time adversaries \mathcal{A} , there exists a negligible function negl such that:

$$\Pr \left[\text{Sig-forge}_{\mathcal{A}, \Pi}^{\text{1-time}}(n) = 1 \right] \leq \text{negl}(n).$$

CONSTRUCTION 12.15

Let $H : \{0, 1\}^* \rightarrow \{0, 1\}^*$ be a function. Construct a signature scheme for messages of length $\ell = \ell(n)$ as follows:

- Gen: on input 1^n , proceed as follows for $i \in \{1, \dots, \ell\}$:
 1. Choose uniform $x_{i,0}, x_{i,1} \in \{0, 1\}^n$.
 2. Compute $y_{i,0} := H(x_{i,0})$ and $y_{i,1} := H(x_{i,1})$.

The public key pk and the private key sk are

$$pk = \begin{pmatrix} y_{1,0} & y_{2,0} & \cdots & y_{\ell,0} \\ y_{1,1} & y_{2,1} & \cdots & y_{\ell,1} \end{pmatrix} \quad sk = \begin{pmatrix} x_{1,0} & x_{2,0} & \cdots & x_{\ell,0} \\ x_{1,1} & x_{2,1} & \cdots & x_{\ell,1} \end{pmatrix}.$$

- Sign: on input a private key sk as above and a message $m \in \{0, 1\}^\ell$ with $m = m_1 \cdots m_\ell$, output the signature $(x_{1,m_1}, \dots, x_{\ell,m_\ell})$.
- Vrfy: on input a public key pk as above, a message $m \in \{0, 1\}^\ell$ with $m = m_1 \cdots m_\ell$, and a signature $\sigma = (x_1, \dots, x_\ell)$, output 1 if and only if $H(x_i) = y_{i,m_i}$ for all $1 \leq i \leq \ell$.

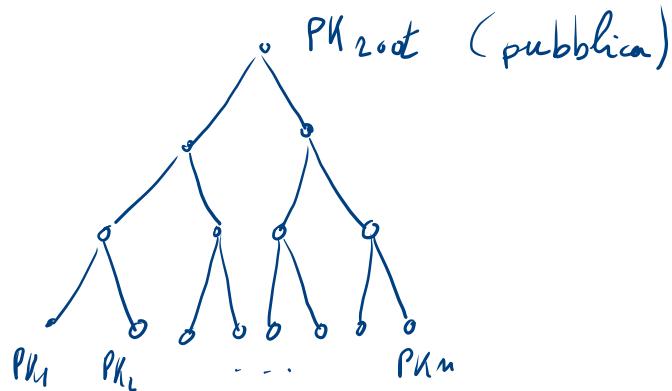
The Lamport signature scheme.

Teorema . Se H è one-way , la costruzione
è sicura .

Firme digitali

Lo schema di Lamport può essere esteso in diversi modi

- t istanze indipendenti per firmare t messaggi
- usare un Merkle tree per avere un tradeoff tra lunghezza chiave pubblica e lunghezza firma



La firma richiede di ri
fornire PK_i (i -esimo uso)
+ path di autenticazione

Altra idea

Sistema one-time usato per

- firmare un messaggio e una nuova chiave pubblica

(PK_0, SK_0) usata per $(m_1, PK_1)_{SK_0}$ (verificato con PK_0)

(PK_1, SK_1) usata per $(m_2, PK_2)_{SK_1}$ (verificato con PK_1)

:

Lo schema di Lamport non può essere usato tout-court

Ma posso usare Lamport su $H(m_i, PK_i)$

Collision resistant hash[↑]

Quantum Key Exchange

Esempio di qui **14** è la cryptography, usa i principi della fisica quantistica.

Metodo che permette a due parti di condividere una chiave segreta

Condividono: canale quantistico + canale classico
(e.g. fibra ottica) (e.g. cavo di rete)

Bennet e Brassard proposero il primo schema nel 1984

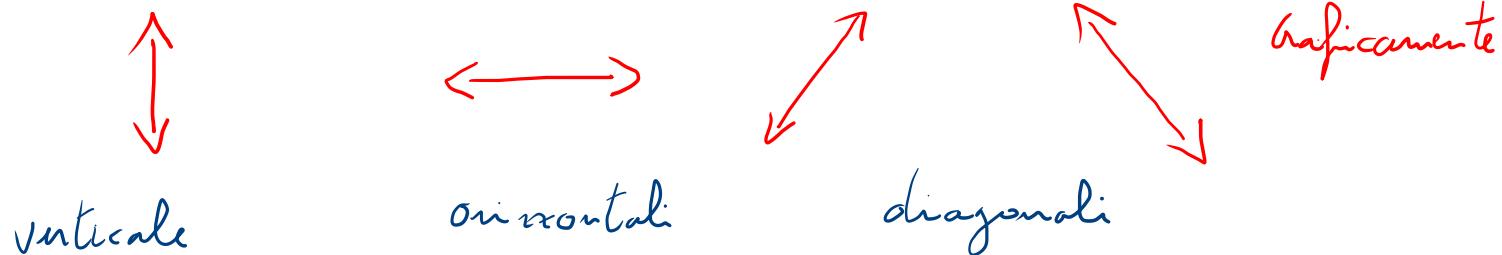
Nota : presentazione informale

La chiave che Alice desidera inviare a Bob è una seq. di bit

Ogni bit è codificato con una proprietà di un fotone.
(e.g. polarizzazione)

Polarizzazione: oscillazione della direzione del suo campo elettrico

Quattro possibili polarizzazioni sono considerate per rappresentare i bit



Alice e Bob concordano che



Un filtro può essere usato per distinguere tra

fotoni orizzontali e verticali (\longleftrightarrow)

Un altro tra fotoni diagonali ($\nwarrow \nearrow$)

Pertanto un filtro permette di leggere un fotone
che codifica zero o uno

Proprietà principale

Quando un fotone passa attraverso il filtro giusto la sua polarizzazione non cambia.

Viceversa, se passa attraverso il filtro stagliato, cambia

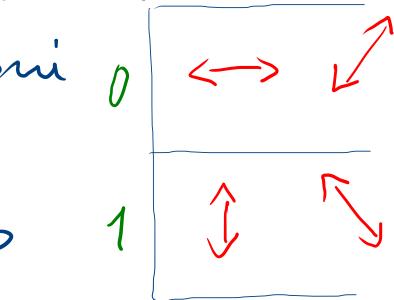
E.g. un fotone verticale o orizzontale che passa attraverso il filtro di distingue verticali ed orizzontali, non cambia polarizzazione.

Viceversa x passa attraverso il filtro de distingue i diagonali, cambia a caso la sua polarizzazione

Idea dello schema

- Alice, per ogni bit della chiave, sceglie un fotone con una delle due possibili polarizzazioni

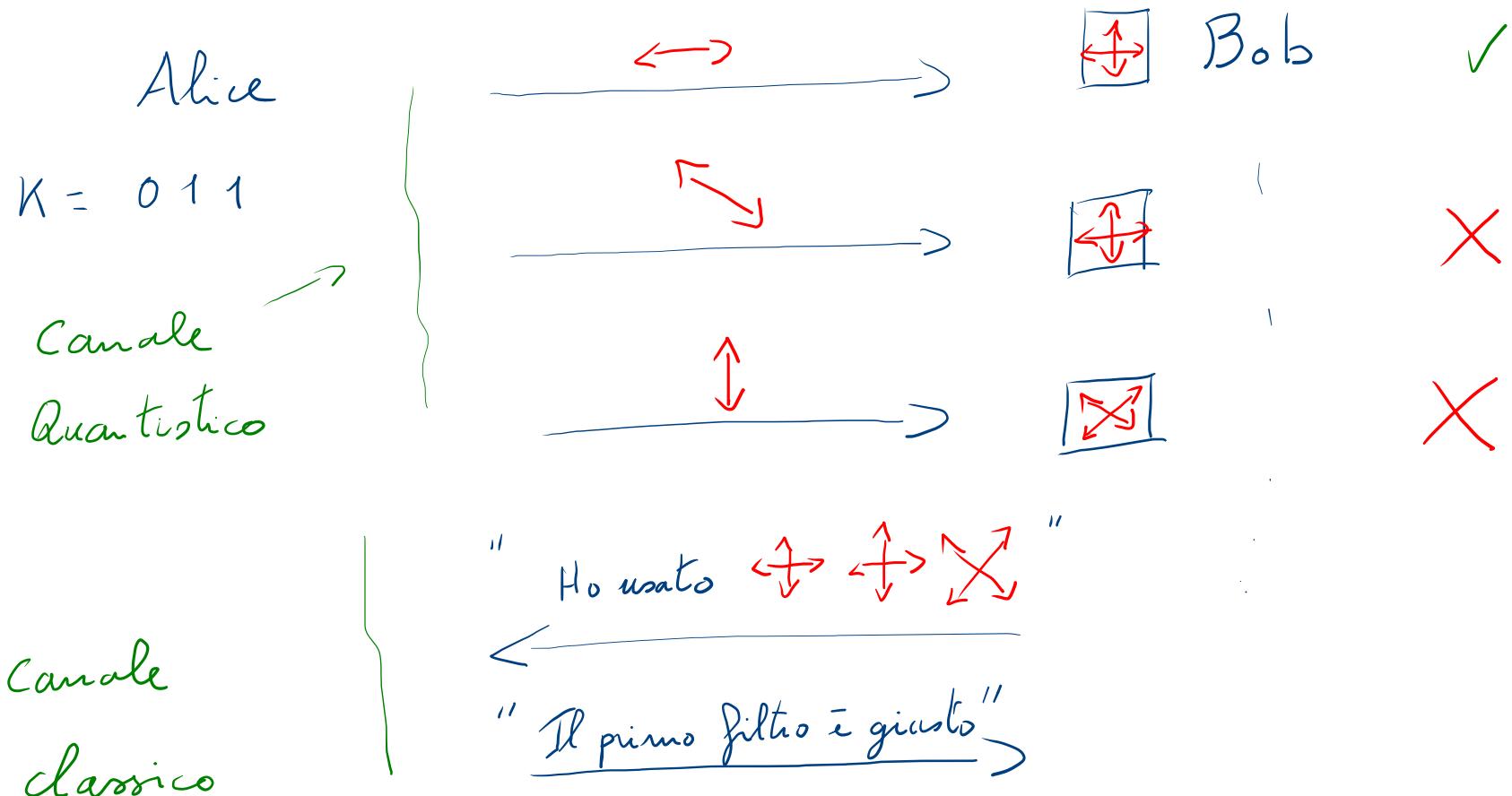
e li trasmette lungo il canale quantistico



- Bob, per ogni fotone trasmesso, sceglie a caso un filtro per leggere fotoni $\uparrow \leftrightarrow$ oppure $\uparrow \downarrow$

Al termine, comunica ad Alice, lungo il canale classico, il filtro usato in ogni trasmissione.

Idea dello schema



Bit segreti condivisi : primo bit (uguale a 0)

Sicurezza dello schema

E Adr cerca di leggere i fotoni trasmessi da Alice lungo il canale quantistico, mediamente nella metà delle lettura cambierà l'orientamento

Alice e Bob, usando la chiave condivisa, si accorgeranno che Adr ha provato a leggere.

↳ modificando l'orientamento
di una parte dei fotoni

Sicurezza dello schema

Nota che Alice, anche sapendo quale filtro Bob ha usato, non riesce a capire quale bit ha letto, poiché ogni filtro rende possibile la lettura sia di zero che di uno.

Conclusioni: le leggi della natura garantiscono che o Alice risulta la sua presenza sul canale quantistico oppure non riceve alcuna informazione sulla chiave segreta.

Nota: Alice e Bob sono autenticati prima di eseguire il protocollo

Quantum Key Distribution Scheme

1. For each key bit, *Alice* sends a photon, whose polarization is randomly selected. She records these polarizations/orientations.
2. For each incoming photon, *Bob* chooses randomly one of the two filters. He writes down his choice as well as the value he records.
3. After all photons have been transmitted, *Bob* reveals, over a conventional and unsecure channel - the phone line for example - to *Alice* the sequence of filters he used.
4. *Alice* tells *Bob* in which cases he chose the correct filter.
5. *Alice* and *Bob* now know in which cases their bits should be identical (when *Bob* used the correct filter). A subset of these bits will form the final key.
6. Finally, *Alice* and *Bob* check the common sequence of bits they hold. In this step error correcting codes are used and some bits are discarded. The remaining ones constitute the common secret key.

Riferimenti per approfondimenti

- Capitolo 14 di "An Introduction to Modern Cryptography"
Y. Lindell e J. Katz (libro di testo, 3^a edizione)
- Capitolo 9 di "Cryptography: Theory and Practice"
M. Paterson e D. R. Stinson (5^a edizione)

Quantum world

- Capitolo 1 di "Quantum Computation and Quantum Information"
M. A. Nielsen e I. L. Chuang, Cambridge press (10th Anniversary edition)
2010
- (consigliare anche la lettura del secondo capitolo --- del resto, studiando l'algabra lineare)