

LICEO SCIENTIFICO NICCOLÒ COPERNICO

ELABORATO PER L'ESAME DI STATO

A.S. 2020/2021

---

# Crittografia Classica e Quantistica

---

Alessandro MINISINI

Classe 5E



# Indice

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduzione</b>  | <b>2</b>  |
| <b>2</b> | <b>Crittografia a chiave privata: Cifrario di Cesare</b>         | <b>2</b>  |
| <b>3</b> | <b>Crittografia a chiave pubblica</b>                            | <b>5</b>  |
| 3.1      | Algoritmo RSA . . . . .  | 6         |
| 3.2      | Sicurezza del sistema RSA . . . . .                              | 9         |
| <b>4</b> | <b>La crittografia quantistica</b>                               | <b>10</b> |
| 4.1      | Protocollo BB84 . . . . .  | 11        |
| 4.2      | Principi fisici che garantiscono il funzionamento del BB84 . . . | 11        |
| 4.3      | Procedimento operativo . . . . .                                 | 12        |
| 4.4      | Le leggi che garantiscono la sicurezza del protocollo . . . . .  | 14        |
| <b>5</b> | <b>Conclusioni</b>   | <b>16</b> |

# 1 Introduzione

La crittografia è l'arte di creare "scritture nascoste", ovvero trovare dei metodi che rendano i messaggi incomprensibili alle persone non autorizzate a leggerlo. La crittografia è da sempre usata in ambito militare, diplomatico, bancario e commerciale, per garantire la giusta riservatezza alle comunicazioni tra le varie parti. I primi sistemi di crittografia, che prendono il nome di *tecniche steganografiche*, sono originari dell'antica Grecia: venivano utilizzate delle aste di legno per codificare messaggi su cinture di cuoio, oppure si tendeva a nascondere il messaggio sotto ai capelli del corriere.

Col tempo le varie tecniche si sono evolute: vennero creati dei cifrari a *chiave privata* (la chiave per la cifratura permette di decifrare il messaggio): un esempio è rappresentato dal cifrario di Cesare, descritto in questo elaborato. Il problema principale di questi sistemi è la distribuzione della chiave, che necessita un canale sicuro tra i due interlocutori, non sempre realizzabile. Per ovviare a questo fatto, sono stati inventati dei metodi a *chiave pubblica* (la conoscenza della chiave di cifratura non permette la decifrazione) che fanno leva sulle nostre limitate capacità computazionali: uno di questi è l'algoritmo RSA, inventato da Whitfield Diffie e Martin Hellman nel 1976.

Con la teorizzazione e la creazione dei primi computer quantistici, è sorta la necessità di trovare nuovi sistemi che non possano essere aggirati: nasce quindi negli anni ottanta la *crittografia quantistica*.

## 2 Crittografia a chiave privata: Cifrario di Cesare

Il *Cifrario di Cesare* è uno dei più antichi cifrari di cui si abbia traccia storica: veniva utilizzato dal generale romano in ambito militare per comunicare con i generali sparsi per l'impero.

Inizialmente Cesare, come scrive lui stesso nel *De bello Gallico*, cifrava i messaggi scambiando le lettere dell'alfabeto latino con quelle greche:

*ibi ex captivis cognoscit, quae apud Ciceronem gerantur quantoque in periculo res sit. Tum cuidam ex equitibus Gallis magnis premiis persuadet, uti Ciceronem epistulam deferat. Hanc Graecis conscriptam litteris mittit, ne intecepta epistula nostra ab hostibus consilia cognoscantur.* (De Bello Gallico V, 48.2-4)

Ai tempi di Cesare, questo sistema poteva essere considerato sicuro, poiché lo studio del greco era riservato solo alla classe dirigente, che poteva permettersi un precettore privato e viaggi di studio in Grecia.

Cesare ricorreva spessissimo a cifrari come quello appena descritto, anche per comunicare coi parenti. Sappiamo, a proposito, che la cifratura era talmente tanto spesso utilizzata da Cesare, che Valerio Probo dedicò ai suoi cifrari un intero trattato, che purtroppo è andato perduto. Ci sono, tuttavia, altre testimonianze, tra le quali Svetonio. Si legge, infatti, nel *De vita Caesarum*:

*Extant et ad Ciceronem, item ad familiares domesticis de rebus, in quibus, si qua occultius perferenda erant, per notas scripsit, id est sic structo litterarum ordine, ut nullum verbum effici posset: quae si qui investigare et persequi velit, quartam elementorum litteram, id est D pro A et perinde reliquas commutet.*

(Vita di Cesare, 56)

Come ci dice Svetonio, il cifrario di Cesare era molto semplice da utilizzare: ogni lettera del testo in chiaro viene sostituita con una lettera che si trova ad un numero fissato di posizioni successive. Questi tipi di cifrario si dicono a sostituzione o scorrimento.

Il suo funzionamento può essere descritto tramite l'aritmetica modulare: la funzione che descrive lo scambio della lettera in posizione  $x$  è la seguente

$$f(x) \equiv x + y \pmod{m}$$

dove  $m$  è il numero di lettere dell'alfabeto.

Da questa si può ricavare facilmente la funzione inversa, che garantisce la validità della cifratura:

$$\begin{aligned} f(x) &\equiv x + y \pmod{m} \\ f(x) - x &\equiv y \pmod{m} \\ x &\equiv f(x) - y \pmod{m} \end{aligned}$$

dunque

$$f^{-1}(x) \equiv x - y \pmod{m}$$

Questi tipi di cifratura sono ormai considerati obsoleti, poiché la potenza di calcolo di un computer permette di provare ogni singola combinazione di lettere in un tempo irrisorio.

In realtà, questo tipo di cifratura è stato superato già nel Medioevo.

Un matematico arabo chiamato al-Kindi (IX secolo d.C.) elaborò un metodo di decrittazione basato sull'*analisi delle frequenze*: le lettere più frequenti nel messaggio criptato devono corrispondere con le lettere più usate dalla lingua del messaggio originale. Questo fatto permette di ridurre drasticamente il numero di tentativi necessari, permettendo di decrittare il messaggio anche a mano.

Il cifrario di Cesare però costituisce la base di molti altri codici conosciuti, come il *cifrario di Vigenère*, sviluppato nel 1586 dall'omonimo diplomatico francese.

Questo cifrario introdusse per la prima volta il concetto di *chiave di criptazione*: per poter criptare e decriptare un messaggio, infatti, era necessario essere a conoscenza di una sequenza di caratteri chiamata *verme*.

L'evoluzione di questi sistemi ha portato alla nascita dell'unico sistema crittografico considerato *perfetto*, poiché la sua inviolabilità è stata comprovata da una dimostrazione matematica.

Questo cifrario è noto col nome *One Time Pad*, ed il suo funzionamento è molto semplice.

Per prima cosa ci serve il messaggio  $M$ , trasformato in forma numerica, e una chiave  $K$  tale che il numero di cifre di  $M$  sia minore o uguale al numero di cifre di  $K$ .

Per ottenere il messaggio cifrato  $C$  è sufficiente sommare  $M$  e  $K$  senza tenere conto degli eventuali riporti intermedi.

Per decrittare il messaggio sarà sufficiente sottrarre cifra per cifra  $C$  e  $K$ , sommando dieci agli eventuali risultati negativi.

In seguito è stato riportato un esempio di criptazione e di decrittazione utilizzando questo cifrario:

*Criptazione*

|                      |   |   |   |   |   |
|----------------------|---|---|---|---|---|
| Messaggio $M$ :      | 4 | 5 | 3 | 0 | 0 |
| Chiave casuale $K$ : | 7 | 7 | 7 | 4 | 6 |
| <hr/>                |   |   |   |   |   |
| Risultato $C$ :      | 1 | 2 | 0 | 4 | 6 |

*Decriptazione*

|                                      |     |     |     |   |   |
|--------------------------------------|-----|-----|-----|---|---|
| Messaggio $C$ :                      | 1   | 2   | 0   | 4 | 6 |
| Chiave casuale $K$ :                 | 7   | 7   | 7   | 4 | 6 |
| Primo risultato dopo la sottrazione: | -6  | -5  | -7  | 0 | 0 |
| Eventuale somma 10:                  | +10 | +10 | +10 |   |   |
| <hr/>                                |     |     |     |   |   |
| Risultato $M$ :                      | 4   | 5   | 3   | 0 | 0 |

E' importante citare questo cifrario perché verrà poi utilizzato in combinazione con la distribuzione della chiave quantistica.

### 3 Crittografia a chiave pubblica

I sistemi di crittografia odierni sono detti a chiave pubblica: ogni persona che vuole parlare in modo cifrato possiede due chiavi, una pubblica e una privata.

Definiamo  $E$  l'algoritmo di crittazione,  $D$  l'algoritmo di decrittazione e  $M$  il messaggio in chiaro da trasmettere. Ci sono quattro algoritmi essenziali che definiscono un sistema a chiave pubblica:

- (a) Decifrare un messaggio cifrato ritorna il messaggio originale

$$D(E(M)) = M$$

- (b) Anche il procedimento opposto deve ritornare il messaggio  $M$

$$E(D(M)) = M$$

- (c) Sia  $E$  che  $D$  devono essere sufficientemente facili da calcolare

- (d) La chiave pubblica non deve compromettere la segretezza della chiave privata

### 3.1 Algoritmo RSA

Questo sistema utilizza degli algoritmi  $E$  e  $D$  che si basano sull'aritmetica modulare. Per poterlo utilizzare dobbiamo trasformare il messaggio  $M$  in forma numerica (ad esempio convertendo le lettere tramite la tabella UNICODE). Supponiamo che  $M$  sia un numero intero tra 0 e  $n - 1$ . Se il messaggio è troppo lungo possiamo dividerlo in più parti e criptarle separatamente. Siano  $e, d, n$  degli interi positivi e definiamo la coppia  $(e, n)$  come chiave pubblica e  $(d, n)$  la chiave privata.

Ora, per criptare il messaggio, sarà sufficiente elevare  $M$  alla  $e$  modulo  $n$ , ottenendo il messaggio cifrato  $C$ . Per decrittare il messaggio invece si eleva  $C$  alla  $d$ -esima potenza modulo  $n$ , ottenendo di nuovo  $M$ .

Formalmente, otteniamo le seguenti definizioni per  $E$  e  $D$ :

$$C \equiv E(M) \equiv M^e \pmod{n} \quad (1)$$

$$M \equiv D(C) \equiv C^d \pmod{n}$$

Ora ci concentriamo sul metodo da seguire per creare le due chiavi (privata e pubblica).

Prima di tutto scegliamo due numeri primi  $p, q$  sufficientemente grandi e li moltiplichiamo per ottenere  $n = pq$ . Anche se  $n$  fa parte della chiave pubblica, la segretezza di  $p$  e  $q$  è garantita dal fatto che ricavarli da  $n$  è un'operazione computazionalmente complessa, impraticabile con i metodi odierni.

Ora rimangono da generare  $e$  e  $d$ .

Scegliamo un  $d$  sufficientemente grande in modo che sia coprimo con  $\varphi(n)$ .

**Definizione** (Funzione  $\varphi$  di Eulero). *La funzione  $\varphi$  di Eulero, detta anche toziente, è una funzione definita, per ogni intero  $n$ , come il numero di interi compresi tra 1 ed  $n$  che sono coprimi con  $n$ .*

*Una delle proprietà fondamentali della funzione è la seguente:*

$$\varphi(ab) = \varphi(a) \cdot \varphi(b) \quad (2)$$

*Notare inoltre che per ogni primo  $p$ ,  $\varphi(p) = p - 1$ .*

Per concludere, troviamo  $e$  utilizzando i valori  $d, p$  e  $q$ , in modo che  $e$  sia l'inverso moltiplicativo di  $d$  modulo  $\varphi(n)$ . Ciò significa soddisfare la seguente congruenza

$$e \cdot d \equiv 1 \pmod{\varphi(n)}$$

ovvero

$$e \cdot d = k \cdot \varphi(n) + 1$$

per un  $k \in \mathbb{Z}$ .

Dalla (2), otteniamo facilmente  $\varphi(n)$ :

$$\begin{aligned}\varphi(n) &= \varphi(p) \cdot \varphi(q) \\ &= (p-1)(q-1) \\ &= n - (p+q) + 1\end{aligned}$$

Per l'aritmetica modulare, l'inverso moltiplicativo di  $a$  modulo  $m$  esiste se e solo se  $a$  ed  $m$  sono coprimi. Siccome  $d$  e  $\varphi(n)$  sono coprimi per ipotesi, allora esiste sicuramente un  $e$  tale che sia l'inverso moltiplicativo di  $d$  e che  $1 \leq e \leq \varphi(n)$ .

A questo punto, dobbiamo verificare che l'algoritmo descritto con le chiavi appena generate soddisfi le condizioni imposte inizialmente.

Innanzitutto notiamo che le condizioni (a) e (b) sono verificabili mediante una sola condizione, infatti:

$$D(E(M)) \equiv (E(M))^d \equiv (M^e)^d \pmod{n} = M^{e \cdot d} \pmod{n} \quad (3)$$

$$E(D(M)) \equiv (D(M))^e \equiv (M^d)^e \pmod{n} = M^{e \cdot d} \pmod{n}$$

Siccome  $e \cdot d = k \cdot \varphi(n) + 1$ , possiamo sostituire l'espressione nelle equazioni precedenti, ottenendo:

$$M^{e \cdot d} \equiv M^{k \cdot \varphi(n) + 1} \pmod{n}$$

Affinché l'algoritmo funzioni, vogliamo che questa espressione valga esattamente  $M$ . Per provarlo, ricordiamo un'altra proprietà della funzione di Eulero: per ogni intero  $M$  coprimo con  $n$ , vale che

$$M^{\varphi(n)} \equiv 1 \pmod{n}$$

Siccome abbiamo definito  $0 \leq M \leq n$ , sappiamo che  $M$  non è coprimo con  $n$  se e solo se  $p \mid M$  o  $q \mid M$ . Di questi due casi ce ne occupiamo in seguito. Per tutti gli altri casi possibili, vale che:

$$M^{e \cdot d} \equiv M^{k \cdot \varphi(n) + 1} \equiv (M^{\varphi(n)})^k \cdot M \equiv 1^k \cdot M \pmod{n} = M$$



Come volevasi dimostrare.

Ora analizziamo il caso particolare in cui  $p \mid M$  o  $q \mid M$ . Siccome  $M < n$ , le due espressioni non possono essere contemporaneamente vere. Senza perdita di generalità, supponiamo che valga  $p \mid M$ . Questa assunzione ci permette di dire che

$$MCD(M, p) = p$$

ovvero

$$M = pt \text{ con } t \in \mathbb{Z}$$

e che

$$MCD(M, q) = 1$$

poiché  $q$  è primo.

Per il piccolo teorema di Fermat, sappiamo che:

$$M^{q-1} \equiv 1 \pmod{q} \quad (4)$$

Ora, eleviamo i due membri della (4) alla  $k(p-1)$ , ottenendo la seguente congruenza (equivalente alla prima):

$$\begin{aligned} M^{k(q-1)(p-1)} &\equiv 1 \pmod{q} \\ M^{k(q-1)(p-1)} &= 1 + hq \end{aligned} \quad (5)$$

per  $h \in \mathbb{Z}$ .

Ora moltiplichiamo i due membri della (5) per  $M$ , ottenendo:

$$M^{1+k(q-1)(p-1)} = M + Mqh$$

Ora sostituiamo alcune parti dell'equazione con alcuni risultati ottenuti sopra, nello specifico sappiamo che  $(q-1)(p-1) = \varphi(n)$  e che  $M = pt$ . Otteniamo che:

$$\begin{aligned} M^{1+k \cdot \varphi(n)} &= M + \overbrace{pq}^n th \\ M^{1+k \cdot \varphi(n)} &= M + nth \\ M^{1+k \cdot \varphi(n)} &\equiv M \pmod{n} \end{aligned}$$

Sostituiamo  $e \cdot d = 1 + k \cdot \varphi(n)$ :

$$M^{e \cdot d} \equiv M \pmod{n}$$

Che è proprio ciò che volevamo verificare.

La dimostrazione quindi garantisce la validità del sistema di criptazione RSA per ogni  $M$ .

### 3.2 Sicurezza del sistema RSA

La sicurezza dell'algoritmo RSA si basa sul fatto che per calcolare  $\varphi(n)$  è necessario fattorizzare  $n$ , siccome  $p$  e  $q$  non sono noti a chi non crea le due chiavi.

Tuttavia, la scomposizione di un numero in fattori primi è un'operazione che richiede molto tempo: tra gli algoritmi più efficienti, quello di Richard Schroepel fattorizza  $n$  in circa

$$e^{\sqrt{\ln n \cdot \ln \ln n}}$$

operazioni. La seguente tabella, stilata dai creatori dell'RSA nel 1978, rappresenta i tempi di calcolo al variare delle cifre di  $n$  (assumendo che un'operazione impieghi in media un microsecondo per essere eseguita):

| Cifre | Numero di operazioni | Tempo impiegato           |
|-------|----------------------|---------------------------|
| 50    | $1.4 \times 10^{10}$ | 3.9 ore                   |
| 75    | $9.0 \times 10^{12}$ | 104 giorni                |
| 100   | $2.3 \times 10^{15}$ | 74 anni                   |
| 200   | $1.2 \times 10^{23}$ | $3.8 \times 10^9$ anni    |
| 300   | $1.5 \times 10^{29}$ | $4.9 \times 10^{15}$ anni |
| 500   | $1.3 \times 10^{39}$ | $4,2 \times 10^{25}$ anni |

Nell'algoritmo RSA vengono solitamente usati  $n$  con circa 200 cifre: questo ci garantisce che la fattorizzazione di  $n$  non sia praticabile neanche ai giorni nostri, sebbene i computer siano notevolmente migliorati dal punto di vista computazionale.

## 4 La crittografia quantistica

Abbiamo capito che un computer tradizionale non è in grado di scomporre numeri grandi in un tempo ragionevole. Questa incapacità viene sfruttata da moltissimi algoritmi di criptazione e questi vengono utilizzati per proteggere i nostri dati più sensibili, come le credenziali bancarie.

Nel 1980 i fisici Paul Benioff e Yuri Manin proposero il primo modello di computer quantistico, basato su dei bit<sup>1</sup> che non sono rappresentati da stati binari (0 e 1), ma da delle particelle subatomiche (chiamate *qbit*).

Queste particelle, a differenza dei bit classici, possono essere forzate in uno stato particolare chiamato *stato di sovrapposizione quantistica*, che permette di immagazzinare nel qbit un quantitativo di informazioni enorme.

Queste informazioni, grazie alle leggi della meccanica quantistica, possono essere processate in un tempo considerevolmente inferiore rispetto ai computer classici.

Al giorno d'oggi la realizzazione di un computer quantistico è molto complessa (sono necessarie temperature estremamente basse, vicine allo zero assoluto, e strumenti che generino onde e le indirizzino verso posizioni molto precise), ma non impossibile: la IBM, Google e altri centri di ricerca hanno già sviluppato dei computer quantistici formati da circa 50 qbits.

Per quanto riguarda l'algoritmo di fattorizzazione, nel 1994 l'informatico Peter Shor ha scritto una procedura che riesce a compiere il calcolo in un tempo irrisorio (nello specifico, il numero di operazioni è dell'ordine di  $\log^2 N \cdot \log \log N \cdot \log \log \log N$ , molto inferiore al tempo dell'algoritmo di Schroepel); l'unico problema prettamente pratico di questo algoritmo è che necessita di all'incirca un milione di qbits per essere eseguito. Con le tecnologie attuali, è impossibile realizzare un computer quantistico così potente, garantendo di fatto la validità degli algoritmi classici ancora per molti anni. Tuttavia, un calcolatore del genere potrebbe venir costruito in futuro, quindi sorge comunque la necessità di creare dei sistemi crittografici che siano in grado di resistere ai tentativi di attacco da parte di questi computer.

---

<sup>1</sup>In informatica, il bit è l'unità di misura elementare dell'informazione.

Questa nuova branca della crittografia è chiamata *Crittografia Quantistica* e si concentra sul trovare un modo per trasmettere una chiave in modo sicuro sfruttando le leggi della meccanica quantistica.

I principali algoritmi quantistici sono il BB84, descritto in seguito, che si basa sulla polarizzazione dei fotoni, e l'ERBE, che sfrutta alcune proprietà che particelle quantistiche "gemelle" possiedono.

## 4.1 Protocollo BB84

Il protocollo BB84 è stato sviluppato da Charles H. Bennet e Gilles Brassard nel 1984.

È stato il primo metodo di crittografia quantistica mai inventato ed è utilizzabile come metodo per comunicare in modo privato una chiave segreta tra due utenti. La comunicazione poi può proseguire tramite il protocollo OTP, descritto in precedenza.

## 4.2 Principi fisici che garantiscono il funzionamento del BB84

Per sfruttare questo sistema si utilizza la luce. Per la meccanica quantistica la luce è composta da quantità discrete di energia, chiamate fotoni. Ogni fotone possiede anche un carattere ondulatorio che gli conferisce un proprio angolo di polarizzazione, definito come l'angolo tra il piano in cui essi oscillano e il piano di propagazione degli stessi fotoni.

Normalmente una sorgente di luce produce fotoni a polarizzazione arbitraria. Per far assumere una particolare polarizzazione ad un fotone si utilizza un filtro polarizzatore, che permette solo ai fotoni con una determinata polarizzazione di proseguire il loro cammino. Ogni polarizzatore ha un angolo ben preciso di polarizzazione che chiameremo  $\theta$ .

Ruotando opportunamente il filtro polarizzatore in modo che permetta il passaggio solo ai fotoni con polarizzazione  $\theta$ ; tutti i fotoni con polarizzazione diversa da  $\theta$  possono essere fermati oppure oltrepassare il filtro con una polarizzazione  $\theta$ .

Le leggi della meccanica quantistica ci dicono che un fotone inizialmente polarizzato con un angolo  $\phi$  oltrepassa il filtro con la seguente probabilità:

$$p_{\theta}(\phi) = \cos^2(\phi - \theta) \tag{6}$$

La probabilità che venga respinto invece è:

$$1 - p_\theta(\phi) = \sin^2(\phi - \theta)$$

Per semplicità, supponiamo di utilizzare raggi luminosi che contengono un singolo fotone polarizzato.

Per i nostri scopi, utilizzeremo solo quattro angoli di polarizzazione:  $0^\circ, 45^\circ, 135^\circ, 180^\circ$ .

I fotoni verranno trasmessi da un capo all'altro della comunicazione tramite fibra ottica, mentre per misurare la polarizzazione del fotone si utilizza un cristallo di calcite. Quando il fotone attraversa il cristallo infatti, possono accadere due cose:

- il fotone lo attraversa in linea retta ed emerge polarizzato perpendicolarmente rispetto all'asse ottico del cristallo
- il fotone viene traslato ed emerge polarizzato parallelamente rispetto all'asse ottico.

Se il fotone è inizialmente polarizzato ortogonalmente rispetto all'asse ottico allora il suo stato non subirà modifiche. Se invece non è così, il fotone seguirà uno dei due cammini con eguale probabilità, subendo un'opportuna ripolarizzazione. Un comportamento del tutto casuale si ha quando la polarizzazione è a metà fra le due direzioni, quindi forma un angolo di  $45^\circ$  o  $135^\circ$  col piano ottico del cristallo: in tal modo ogni informazione sullo stato iniziale viene completamente persa.

Per questo motivo, nel protocollo BB84 vengono utilizzati gli angoli precedentemente indicati: se per le misure ortogonali non è necessario apportare modifiche al cristallo, per le misure diagonali è sufficiente ruotare il cristallo originale di  $45^\circ$ .

### 4.3 Procedimento operativo

Ipotizziamo che Alice e Bob vogliano comunicare in maniera cifrata e che utilizzino il protocollo BB84 per trasferire la chiave di crittazione. Ognuno dei bit che compongono la chiave è rappresentato da un fotone polarizzato come nella tabella seguente:

| R                 | D          | Bit |
|-------------------|------------|-----|
| $\leftrightarrow$ | $\nearrow$ | 1   |
| $\updownarrow$    | $\searrow$ | 0   |

Supponiamo, per semplicità, che ogni impulso contenga un solo fotone.  
Il protocollo è il seguente:

1. Alice sceglie una sequenza casuale di bit ed una sequenza casuale di basi di polarizzazione (rettilinea o diagonale) e manda a Bob una sequenza di fotoni, ognuno rappresentante un bit della sequenza nella base scelta.
2. Bob sceglie casualmente per ogni fotone mandatogli da Alice (e indipendentemente dalle scelte fatte da Alice) se misurare la polarizzazione rettilinea o diagonale e interpreta ogni risultato come 0 o 1, a seconda dell'esito della corrispondente misura. Bob ottiene quindi dati significativi solo dal 50% dei fotoni che ha misurato (quelli per i quali ha indovinato la corretta base di polarizzazione) supponendo che non vi siano state alterazioni dovute da eventuali intrusi.
3. Bob annuncia pubblicamente le basi con cui ha analizzato i fotoni.
4. Alice comunica pubblicamente a Bob se per ciascun fotone che egli ha ricevuto ha eseguito il tipo giusto di misurazione. Si scartano tutte le posizioni dei bit per le quali Bob ha eseguito un tipo di misurazione sbagliato o per le quali non è stato rilevato alcun fotone (Ciò può capitare per svariate ragioni, dovute sia ad un eventuale origliatore che all'errore degli strumenti di misura).
5. Alice e Bob, per verificare se le loro risultanti sequenze di bit sono identiche, confrontano pubblicamente un sottoinsieme casuale di bit correttamente ricevuti da Bob, cioè con la base esatta. Se tutti i fotoni concordano, Alice e Bob possono concludere che la trasmissione quantistica è stata sicura, per cui i rimanenti bit segreti possono costituire la chiave privata del protocollo OTP. Se invece vi è stato un notevole origliamento, la trasmissione è scartata e si riprova con un nuovo gruppo di fotoni.

In seguito è riportato un esempio di trasmissione della chiave quantistica da parte dei due interlocutori.

|     |            |                |            |                   |                |                |                   |                   |            |            |                |            |            |            |                |
|-----|------------|----------------|------------|-------------------|----------------|----------------|-------------------|-------------------|------------|------------|----------------|------------|------------|------------|----------------|
| 1a. | 0          | 1              | 1          | 0                 | 1              | 1              | 0                 | 0                 | 1          | 0          | 1              | 1          | 0          | 0          | 1              |
| 1b. | D          | R              | D          | R                 | R              | R              | R                 | R                 | D          | D          | R              | D          | D          | D          | R              |
| 1c. | $\nearrow$ | $\updownarrow$ | $\searrow$ | $\leftrightarrow$ | $\updownarrow$ | $\updownarrow$ | $\leftrightarrow$ | $\leftrightarrow$ | $\searrow$ | $\nearrow$ | $\updownarrow$ | $\searrow$ | $\nearrow$ | $\nearrow$ | $\updownarrow$ |
| 2a. | R          | D              | D          | R                 | R              | D              | D                 | R                 | D          | R          | D              | D          | D          | D          | R              |
| 2b. | 1          |                | 1          |                   | 1              | 0              | 0                 | 0                 |            | 1          | 1              | 1          |            | 1          | 0              |
| 3.  | R          |                | D          |                   | R              | D              | D                 | R                 |            | R          | D              | D          |            | D          | R              |
| 4a. |            |                | OK         |                   | OK             |                |                   | OK                |            |            |                | OK         |            | OK         | OK             |
| 4b. |            |                | 1          |                   | 1              |                |                   | 0                 |            |            |                | 1          |            | 0          | 1              |
| 5a. |            |                |            |                   | 1              |                |                   |                   |            |            |                |            |            | 0          |                |
| 5b. |            |                |            |                   | OK             |                |                   |                   |            |            |                |            |            | OK         |                |
| 5c. |            |                | 1          |                   |                |                |                   | 0                 |            |            |                | 1          |            |            | 1              |

| <i>Trasmissione quantistica</i> |  |
|---------------------------------|--|
| 1a.                             | Bit scelti da Alice  |
| 1b.                             | Basi scelte da Alice   |
| 1c.                             | Fotoni polarizzati spediti sul canale quantistico                  |
| 2a.                             | Basi scelte da Bob per misurare i fotoni                           |
| 2b.                             | Bit ricevuti da Bob  |
| <i>Discussione pubblica</i>     |  |
| 3.                              | Bob dichiara le basi che ha utilizzato                             |
| 4a.                             | Alice dice a Bob quali basi erano corrette                         |
| 4b.                             | Bit validi per la chiave di criptazione                            |
| 5a.                             | Bob rivela alcuni bit della chiave ottenuta                        |
| 5b.                             | Alice conferma i bit inviati, verificando la validità della chiave |
| <i>Risultato</i>                |  |
| 5c.                             | Bit rimanenti utilizzati nella chiave                              |

#### 4.4 Le leggi che garantiscono la sicurezza del protocollo

Una delle classiche strategie di intercettazione utilizzate è il cosiddetto *Man-in-the-middle*: L'origliatore, tipicamente chiamato Eva, intercetta le comunicazioni e ritrasmette al destinatario ciò che riceve.

In questo caso, Eva intercetta ogni fotone in una delle due basi e spedisce a Bob un fotone nello stato che ha rilevato: in questo modo le informazioni correttamente captate saranno il 50% di quelle totali.

Infatti, dalla (6) possiamo ricavare le probabilità che ogni fotone attraversi correttamente una base:

|                   | 0°  | 90° | 45° | 135° |
|-------------------|-----|-----|-----|------|
| $\leftrightarrow$ | 1   | 0   | 0.5 | 0.5  |
| $\updownarrow$    | 0   | 1   | 0.5 | 0.5  |
| $\nearrow$        | 0.5 | 0.5 | 1   | 0    |
| $\searrow$        | 0.5 | 0.5 | 1   | 0    |

Ad esempio, la probabilità che Eva riesca a captare correttamente un fotone polarizzato a 90° vale:

$$p(90^\circ) = \frac{p_{0^\circ}(90^\circ) + p_{45^\circ}(90^\circ) + p_{90^\circ}(90^\circ) + p_{135^\circ}(90^\circ)}{4} = \frac{0 + 1 + 0.5 + 0.5}{4} = 0.5$$

La probabilità che Eva spedisca correttamente il qbit a Bob è però diversa da  $p(90^\circ)$ , perché per *il principio di indeterminazione di Heisenberg*, la misura comporta il collasso dello stato quantico del fotone: se inizialmente la polarizzazione della particella non è concorde con la base utilizzata, la misura obbliga il fotone ad orientarsi lungo uno dei due assi ottici del filtro utilizzato.

Di conseguenza, la probabilità che Bob ricevi correttamente il bit di Alice è descritta dalla seguente formula:

$$\frac{\sum_{i \in A} \sum_{j \in A} p_i(90^\circ) \cdot p_j(90^\circ)}{16} = 0.25$$

dove  $A = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$ .

L'elevata discrepanza tra le due probabilità di errore permette di stabilire se la chiave è stata trasmessa in modo sicuro. Se Eva decide di intercettare solo una parte dei bit totali, diventa più complesso stabilire se ci sono state interferenze; è stato comunque dimostrato che con un errore del 15% è possibile stabilire con certezza che la trasmissione non è avvenuta segretamente.

Un'altra strategia che Eva potrebbe seguire è copiare il fotone trasmesso da Alice per poi effettuare le misurazioni quando le basi vengono rilevate. Oltre alle difficoltà incontrate nel conservare la particella, è dimostrato che clonare perfettamente una particella quantistica è impossibile (questo principio è chiamato *teorema della non-clonazione*). Tuttavia, dobbiamo tenere in



considerazione che esistono delle macchine che raggiungono un ottimo stato di clonazione compatibili col teorema sopra citato.

Per concludere, non esistono metodi sufficientemente precisi per intercettare correttamente il messaggio senza essere scoperti dai due interlocutori legittimi.

## 5 Conclusioni

Sebbene questi protocolli siano stati elaborati in un tempo recente e la teoria su cui si basano sia ancora in fase di studio, esistono in commercio vari strumenti sviluppati da varie nazioni che permettono lo scambio di una chiave in via quantistica.

Inoltre, sono state realizzate delle reti di distribuzione di chiavi quantistiche in svariate aree del globo: la rete quantistica americana, realizzata tra il 2002 e il 2007, connette le università di Harvard e di Boston con la sede centrale della BBN technologies, una società di consulenza informatica locale.

Ci sono reti a Vienna, che collegano il municipio con la tesoreria comunale, a Tokyo, a Canberra, a Ginevra e Wuhu in Cina; quest'ultima è la più lunga rete realizzata fino ad ora (all'incirca 100 km). Progetti futuri comportano la distribuzione della chiave per via satellitare, siccome l'atmosfera terrestre causa notevoli disturbi.

## Bibliografia

I.V. Volovich, Ya.I. Volovich, *On Classical and Quantum Cryptography*,  
<https://arxiv.org/abs/quant-ph/0108133v1>

Evgeny Milanov, *The RSA Algorithm*,  
<https://pdfdirectory.com/702-tutorial-the-rsa-algorithm.pdf>

Dispense dell'Università di Roma, *Crittografia Quantistica*,  
<http://www.dia.uniroma3.it/dispense/rota/quan1.pdf>

Giuseppe Sammarco, *Breve storia della Crittografia*,  
<http://www.iasi.cnr.it/seminaricrittografia/BreveStoriaCrittografia.pdf>

## Conferenze

Nicola Laurenti, *Crittografia quantistica: la sicurezza informatica del futuro*,  
Orientamento UniPd (Online, 25 febbraio 2021)

Federico Mattei, *Una nuova era dell'informatica: il computer quantistico*,  
TEDxBologna (Bologna, 30 ottobre 2019)

## Sitografia

Wikipedia, *Cifrario di Cesare*, [https://it.wikipedia.org/wiki/Cifrario\\_di\\_Cesare](https://it.wikipedia.org/wiki/Cifrario_di_Cesare)

Wikipedia, *One Time Pad*, [https://it.wikipedia.org/wiki/Cifrario\\_di\\_Vernam](https://it.wikipedia.org/wiki/Cifrario_di_Vernam)

Wikipedia, *BB84*, <https://it.wikipedia.org/wiki/BB84>