# Crittografia Post-Quantum

Olivieri Michele

December 18, 2025

# Contents

# 1 Introduzione

## 1.1 Contesto e motivazioni

Per comprendere l'importanza della crittografia post-quantistica, è fondamentale analizzare il contesto in cui essa si inserisce e le motivazioni che ne hanno guidato lo sviluppo.

La crittografia classica, come visto a lezione, pone le sue fondamenta su problemi computazionalmente difficili per i quali non esistono algoritmi efficienti in grado di risolverli in tempi polinomiali.

Tuttavia, l'emergere dei computer quantistici ha introdotto una nuova dimensione nel panorama della sicurezza informatica. Questi dispositivi sfruttando i principi della meccanica quantistica sono in grado di eseguire calcoli in modo radicalmente diverso rispetto ai computer classici, rendendoli potenzialmente capaci di risolvere in tempi polinomiali quei problemi matematici che risultano intrattabili per le macchine convenzionali.

## 1.2 Minaccia dei computer quantistici: Perché è necessaria?

L'ipotesi di una minaccia quantistica emerge già nel ventesimo secolo, quando nel 1994 l'algoritmo di Shor dimostra che un computer quantistico sufficientemente potente potrebbe fattorizzare grandi numeri interi e calcolare logaritmi discreti in tempo polinomiale. Questo rende vulnerabili algoritmi come RSA, ECC e DSA, che costituiscono la base della sicurezza informatica moderna.

Per molti anni il problema è rimasto solo teorico, perché il calcolo quantistico non aveva applicazioni pratiche. Lo scenario è cambiato con i recenti progressi nel settore e con lo sviluppo dei primi prototipi di calcolatori quantistici da parte di aziende come Google e Microsoft, che hanno riportato risultati significativi con i loro progetti: Myorana 1 e Willow. Questi sviluppi hanno aumentato le preoccupazioni sulla solidità degli schemi crittografici attuali.

Sebbene le macchine quantistiche siano ancora in fase sperimentale, diversi scienziati ritengono che la loro costruzione su larga scala sia ormai una sfida principalmente ingegneristica, e alcuni prevedono la loro maturazione entro i prossimi vent'anni. Considerando che l'attuale infrastruttura crittografica ha richiesto quasi due decenni per essere implementata, risulta necessario iniziare da subito la transizione verso sistemi progettati per resistere al calcolo quantistico.

## 1.3 Obiettivi della crittografia post-quantistica

L'obiettivo della crittografia post-quantistica è quindi sviluppare algoritmi crittografici sicuri sia contro i computer quantistici che classici, garantendo così un futuro sicuro anche in un'era dominata dai computer quantistici.

La crittografia post-quantistica non si limita a sostituire gli algoritmi vulnerabili, ma mira a costruire un'infrastruttura di sicurezza robusta e duratura, capace di adattarsi alle sfide tecnologiche future mantenendo la compatibilità con i protocolli e le reti esistenti.

# 2 Fondamenti della crittografia classica

Per comprendere le sfide poste dai computer quantistici alla crittografia moderna, è fondamentale conoscerne i principi di base dietro i principali algoritmi crittografici attualmente in uso.

## 2.1 Crittografia a chiave pubblica: RSA, ECC

La crittografia a chiave pubblica, introdotta nel 1976 da Diffie, Hellman e Merkle, costituisce una svolta fondamentale nel panorama della sicurezza informatica moderna. A differenza dei sistemi simmetrici, che vincolano mittente e destinatario alla condivisione di un unico segreto, i protocolli asimmetrici impiegano una coppia di chiavi: una pubblica $k_{\text{pub}}$, liberamente distribuibile, e una privata $k_{\text{prv}}$, mantenuta segreta dal proprietario. La sicurezza di questo sistema si basa sulla difficoltà di risalire alla chiave privata a partire da quella pubblica. Le funzioni di cifratura $C$ e decifratura $D$ sono note a tutti, e per ogni messaggio $m$ deve valere:

$$D(C(m; k_{\text{pub}}); k_{\text{prv}}) = m.$$

Il funzionamento di questo sistema si basa sulle funzioni one-way trapdoor: operazioni matematiche semplici da eseguire in una direzione, ma computazionalmente intrattabili da invertire senza la conoscenza di un'informazione specifica (la "trappola").

La teoria dei numeri e l'algebra modulare forniscono il substrato matematico necessario per generare tali funzioni; a seconda del problema matematico sottostante, si distinguono i vari algoritmi di crittografia asimmetrica oggi in uso.

**Richiami di algebra modulare** L'aritmetica modulare è un sistema in cui i numeri si riavvolgono entro un intervallo fissato da un modulo $n$. Quando un valore supera (o scende sotto) questo intervallo, viene riportato all'interno prendendo il resto della divisione per $n$.

Un esempio quotidiano è l'orologio: in un sistema a 12 ore il modulo è 12. Se sono le 10 e aggiungo 4 ore, il risultato non è 14, ma 2, perché:

$$14 \equiv 2 \pmod{12}.$$

Per calcolare $c = a \bmod b$ si considera il resto della divisione intera tra $a$ e $b$, ottenendo un valore sempre compreso tra 0 e $b-1$, esempio: $6 \bmod 4 = 2$.

**Problemi difficili**

- **Fattorizzazione:** dati $p, q$ è facile calcolare $n = pq$; dato $n$ è difficile trovare $p$ e $q$.
- **Radice modulare:** dato $y = x^z \bmod s$ invertire la potenza è difficile senza conoscere $\varphi(s)$.
- **Logaritmo discreto:** data $y = x^z \bmod s$ trovare $z$ è computazionalmente difficile.

### 2.1.1 RSA

Il cifrario RSA, proposto da Rivest, Shamir e Adleman nel 1978, è il sistema crittografico a chiave pubblica più diffuso e studiato. La sua sicurezza si fonda sulla difficoltà computazionale della fattorizzazione di numeri interi molto grandi.

**Generazione delle chiavi** Ogni utente genera la propria coppia di chiavi attraverso i seguenti passaggi:

1. Scelta di due numeri primi $p$ e $q$ molto grandi
2. Calcolo di $n = pq$ e della funzione di Eulero $\phi(n) = (p-1)(q-1)$
3. Scelta di un intero $e$ minore di $\phi(n)$ e coprimo con esso
4. Calcolo dell'intero $d$, inverso moltiplicativo di $e$ modulo $\phi(n)$

La chiave pubblica è la coppia $(e, n)$, mentre la chiave privata è $d$. La cifratura di un messaggio $m$ avviene calcolando $c = m^e \bmod n$, mentre la decifratura richiede il calcolo di $m = c^d \bmod n$.

La correttezza dell'algoritmo è garantita dal teorema di Eulero: poiché $ed \equiv 1 \pmod{\phi(n)}$, si ha $ed = 1 + k\phi(n)$ per qualche intero $k$, e quindi:

$$m^{ed} \bmod n = m^{1+k\phi(n)} \bmod n = m \cdot (m^{\phi(n)})^k \bmod n = m \bmod n$$

**Sicurezza e dimensioni delle chiavi** La sicurezza di RSA dipende dall'impossibilità pratica di fattorizzare $n$ quando questo è sufficientemente grande. Conoscendo la fattorizzazione $n = pq$, un attaccante potrebbe infatti calcolare $\phi(n)$ e di conseguenza la chiave privata $d$.

Attualmente, le dimensioni delle chiavi considerate sicure sono di almeno 2048 bit, con raccomandazioni crescenti verso 4096 bit per applicazioni che richiedono sicurezza a lungo termine. Chiavi di 1024 bit sono considerate obsolete e vulnerabili ad attacchi con risorse computazionali moderne.

### 2.1.2 Crittografia su Curve Ellittiche (ECC)

La crittografia su curve ellittiche, sviluppata indipendentemente da Neal Koblitz e Victor Miller nel 1985, offre un'alternativa matematicamente elegante e computazionalmente efficiente a RSA.

**Fondamenti matematici** Una curva ellittica su un campo finito $\mathbb{Z}_p$ (con $p$ primo e $p > 3$) è definita dall'equazione di Weierstrass in forma normale:

$$y^2 = x^3 + ax + b$$

dove $a, b \in \mathbb{Z}_p$ soddisfano la condizione $4a^3 + 27b^2 \bmod p \neq 0$, che garantisce l'assenza di punti singolari sulla curva.

L'insieme dei punti $(x, y)$ che soddisfano questa equazione, insieme al punto all'infinito $\mathcal{O}$, forma un gruppo abeliano additivo. È possibile definire un'operazione di addizione tra punti della curva tale che, dati due punti $P$ e $Q$, la loro somma $P + Q$ sia ancora un punto della curva.

Per punti distinti $P = (x_P, y_P)$ e $Q = (x_Q, y_Q)$, con $P \neq -Q$, si ha:

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}, \quad x_S = \lambda^2 - x_P - x_Q, \quad y_S = -y_P + \lambda(x_P - x_S)$$

dove $S = P + Q = (x_S, y_S)$. Nel caso di raddoppio di un punto ($P = Q$), il coefficiente angolare diventa $\lambda = \frac{3x_P^2 + a}{2y_P}$.

**Il problema del logaritmo discreto** La sicurezza di ECC si basa sulla difficoltà del problema del logaritmo discreto su curve ellittiche (ECDLP): dati due punti $P$ e $Q$ sulla curva, trovare l'intero $k$ tale che $Q = kP$, dove $kP$ denota l'addizione di $P$ con se stesso $k$ volte.

La moltiplicazione scalare $Q = kP$ è computazionalmente efficiente (tempo polinomiale), mentre il problema inverso è considerato intrattabile: tutti gli algoritmi classici noti hanno complessità esponenziale nella dimensione della chiave.

**Vantaggi rispetto a RSA** ECC offre lo stesso livello di sicurezza di RSA con chiavi significativamente più corte. Una chiave ECC di 256 bit fornisce una sicurezza paragonabile a una chiave RSA di 3072 bit. Questo si traduce in:

- Minore occupazione di memoria e larghezza di banda
- Operazioni crittografiche più veloci
- Minore consumo energetico, cruciale per dispositivi mobili e IoT

### 2.1.3 Sicurezza classica

Entrambi gli algoritmi sono considerati sicuri nell'ambito del calcolo classico per dimensioni di chiave appropriate. La loro robustezza deriva dalla complessità computazionale dei problemi matematici sottostanti:

- **Fattorizzazione per RSA**: il miglior algoritmo classico noto è il General Number Field Sieve (GNFS), con complessità sub-esponenziale $O(e^{(64/9)^{1/3}(\ln n)^{1/3}(\ln \ln n)^{2/3}})$
- **ECDLP per ECC**: gli algoritmi più efficienti, come il metodo rho di Pollard, hanno complessità completamente esponenziale $O(\sqrt{n})$, dove $n$ è l'ordine del gruppo

Questa differenza nella complessità degli attacchi spiega perché ECC richiede chiavi più corte per garantire lo stesso livello di sicurezza.

RSA e ECC costituiscono oggi la base dell'infrastruttura di sicurezza digitale globale, utilizzati in TLS/SSL per la sicurezza web, in SSH per l'accesso remoto sicuro, nella firma digitale di documenti e software, e in numerose altre applicazioni critiche.

## 2.2 Limiti rispetto ai computer quantistici

L'avvento dei computer quantistici rappresenta una minaccia fondamentale per la sicurezza di RSA e ECC. La differenza sostanziale tra calcolo classico e quantistico risiede nell'unità fondamentale di informazione: mentre i computer classici utilizzano bit che possono trovarsi negli stati 0 o 1, i computer quantistici operano con qubit che, grazie al principio di sovrapposizione quantistica, possono esistere simultaneamente in una combinazione di entrambi gli stati.

**Vulnerabilità di RSA e ECC** Nel 1994, Peter Shor sviluppò un algoritmo quantistico in grado di fattorizzare numeri interi e risolvere il problema del logaritmo discreto in tempo polinomiale. Più precisamente, l'algoritmo di Shor ha complessità $O((\log N)^3)$ per la fattorizzazione di un numero $N$, rendendo vulnerabili sia RSA che ECC.

Per RSA, un computer quantistico sufficientemente potente potrebbe:

- Fattorizzare $n = pq$ in tempo polinomiale
- Calcolare $\phi(n) = (p-1)(q-1)$
- Determinare la chiave privata $d$ dall'equazione $ed \equiv 1 \pmod{\phi(n)}$

Per ECC, l'algoritmo di Shor si estende naturalmente al problema del logaritmo discreto su curve ellittiche, permettendo di calcolare $k$ da $Q = kP$ con la stessa efficienza.

# 3 Computazione quantistica e impatto sulla crittografia classica

## 3.1 Nozioni base di computazione quantistica

### 3.1.1 Richiami essenziali di meccanica quantistica

La computazione quantistica si basa su alcuni principi della meccanica quantistica che non hanno un analogo diretto nel calcolo classico. In questa sede non interessa una descrizione formale, ma l'introduzione dei concetti che verranno utilizzati nel seguito.

Il primo concetto fondamentale è quello di *stato quantistico*. A differenza del bit classico, che può assumere solo i valori 0 o 1, un sistema quantistico può trovarsi in una sovrapposizione di stati. Questo significa che lo stato di un qubit può essere descritto come una combinazione lineare di $|0\rangle$ e $|1\rangle$.

Un secondo concetto chiave è l'*entanglement*, una correlazione tra più qubit tale per cui lo stato di ciascun qubit non può essere descritto indipendentemente dagli altri. Questa proprietà consente di rappresentare informazioni distribuite su più qubit in modo non separabile.

Infine, la *misura* di uno stato quantistico provoca il collasso della sovrapposizione, restituendo un risultato classico. La probabilità dei diversi esiti dipende dallo stato quantistico prima della misura.

Questi tre elementi — sovrapposizione, entanglement e misura — costituiscono la base della differenza tra computazione classica e computazione quantistica.

### 3.1.2 Qubit e operazioni quantistiche

L'unità fondamentale di informazione in un computer quantistico è il qubit. Dal punto di vista matematico, un qubit è rappresentato come un vettore in uno spazio di Hilbert bidimensionale, mentre un registro di $n$ qubit è descritto in uno spazio di dimensione $2^n$.

Le operazioni sui qubit sono realizzate tramite *porte quantistiche*, che corrispondono a trasformazioni lineari e reversibili. A differenza delle porte logiche classiche, le porte quantistiche agiscono su stati in sovrapposizione, trasformando simultaneamente tutte le componenti dello stato.

Questa caratteristica permette di eseguire operazioni su un numero esponenziale di stati in modo parallelo, fenomeno noto come *parallelismo quantistico*.

### 3.1.3 Modello di calcolo quantistico

Un computer quantistico segue un modello di calcolo diverso da quello classico. Un algoritmo quantistico è tipicamente composto da tre fasi:

- preparazione dello stato iniziale;
- applicazione di una sequenza di porte quantistiche;
- misura finale.

Durante la fase di evoluzione, lo stato del sistema viene trasformato in modo deterministico, mentre l'osservazione del risultato è intrinsecamente probabilistica.

La potenza del modello di calcolo quantistico non deriva dalla possibilità di "provare tutte le soluzioni", ma dalla capacità di *interferenza*, che consente di amplificare le soluzioni corrette e sopprimere quelle errate.

Questo aspetto è centrale per comprendere come algoritmi quantistici specifici, come l'algoritmo di Shor, riescano a risolvere problemi matematici ritenuti computazionalmente difficili nel modello classico.

### 3.1.4 Collegamento con la crittografia

I concetti introdotti in questa sezione costituiscono la base teorica necessaria per analizzare l'impatto della computazione quantistica sulla crittografia moderna. In particolare, la possibilità di sfruttare sovrapposizione e interferenza consente di affrontare in modo efficiente problemi come la fattorizzazione di interi e il calcolo del logaritmo discreto.

Nel prossimo punto verrà analizzato l'algoritmo di Shor, mostrando come il modello di calcolo quantistico renda concretamente possibile la rottura dei protocolli crittografici asimmetrici descritti nei capitoli precedenti.

### 3.1.5 Computer quantistici reali e sfide di realizzazione

Un computer quantistico reale è un dispositivo composto da qubit fisici, ciascuno dei quali deve essere mantenuto in stato quantistico per durate sufficienti allo svolgimento delle operazioni. Le tecnologie più diffuse utilizzano qubit superconduttori, che operano a temperature estremamente basse, prossime allo zero assoluto, tipicamente dell'ordine di decine di millikelvin, al fine di ridurre il rumore termico e preservare la coerenza quantistica.

Queste condizioni sono ottenute tramite refrigeratori a diluizione, dispositivi complessi e costosi che rendono difficile la scalabilità dei sistemi.

Anche minime interferenze ambientali possono causare la decoerenza, ovvero la perdita delle proprietà quantistiche dei qubit, compromettendo il calcolo.

Un'ulteriore sfida è rappresentata dal controllo delle operazioni quantistiche. Le porte devono essere applicate con estrema precisione, poiché errori anche molto piccoli si accumulano rapidamente all'aumentare del numero di qubit.

Per ottenere un qubit logico affidabile sono necessari molti qubit fisici, sfruttando tecniche di correzione d'errore quantistica. Gli schemi attuali richiedono tipicamente centinaia o migliaia di qubit fisici per realizzare un singolo qubit logico utilizzabile.

Inoltre, i tempi di coerenza dei qubit nei sistemi attuali sono limitati, rendendo impraticabile l'esecuzione di algoritmi complessi su larga scala, come quelli necessari per rompere RSA a 2048 bit.

Per queste ragioni si ritiene che siano necessari ancora diversi anni, se non decenni, prima che computer quantistici in grado di compromettere concretamente i sistemi crittografici attuali diventino disponibili. Tuttavia, i progressi tecnologici e gli investimenti in corso rendono questa prospettiva rilevante dal punto di vista della sicurezza a lungo termine.

## 3.2 Algoritmi quantistici rilevanti: Shor, Grover

## 3.3 Implicazioni per la sicurezza dei sistemi attuali

# 4 Introduzione alla crittografia post-quantistica

## 4.1 Definizione, requisiti e obiettivi

**Requisiti pratici e tempistiche** Sebbene la minaccia sia teoricamente dimostrata, la realizzazione pratica di computer quantistici capaci di violare RSA ed ECC richiede risorse considerevoli. Secondo stime del NIST, per compromettere una chiave RSA-2048 sarebbero necessari diversi milioni di qubit logici affidabili, mentre le implementazioni attuali (2024) operano con centinaia di qubit fisici caratterizzati da elevati tassi di errore.

La transizione da qubit fisici a qubit logici richiede tecniche di correzione degli errori quantistici che impongono un overhead significativo: potrebbero essere necessari da centinaia a migliaia di qubit fisici per realizzare un singolo qubit logico stabile.

Nonostante queste difficoltà tecnologiche, il principio "harvest now, decrypt later" rappresenta una preoccupazione concreta: un attaccante potrebbe intercettare e archiviare oggi comunicazioni cifrate con RSA o ECC, per decifrarle in futuro quando disporrà di computer quantistici sufficientemente

potenti. Questa considerazione è particolarmente rilevante per dati che richiedono riservatezza a lungo termine, come informazioni mediche, segreti industriali o comunicazioni governative.

## 4.2 Ruolo del Nist e processo di standardizzazione

**La risposta: crittografia post-quantistica** Di fronte a questa minaccia emergente, il National Institute of Standards and Technology (NIST) ha avviato nel 2016 un processo di standardizzazione per identificare algoritmi crittografici resistenti agli attacchi quantistici. Nel luglio 2022, il NIST ha annunciato i primi algoritmi selezionati per la standardizzazione, basati su problemi matematici ritenuti difficili anche per computer quantistici, come i reticoli algebrici e i codici correttori di errori.

La migrazione verso la crittografia post-quantistica rappresenta una delle sfide più urgenti per la sicurezza informatica moderna, richiedendo un'attenta pianificazione per sostituire l'infrastruttura crittografica esistente mantenendo retrocompatibilità e garantendo una transizione graduale e sicura.

# 5 Panorama sugli algoritmi post-quantistici