UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
Corso di Laurea Magistrale in Ingegneria Informatica

Relazione per il Corso di Sicurezza Informatica

QUANTUM RANDOM NUMBER GENERATORS

Applicazioni di tecnologie quantiche a sistemi crittografici tradizionali

Professore: Chiar.mo Prof. Federico Cerutti

Studenti:
Alessandro Grassi 719064
Danila Turra 720837

Anno Accademico 2021/2022

Indice

1 Introduzione **4**

2 Prerequisiti **5**
2.1 Superposition 5
2.2 Casualità della misurazione 5
2.3 Entanglement 6
2.4 Bell inequality 6
2.5 Casualità 6

3 Random Number Generators **7**
3.1 Pseudorandom-number generators 8
3.2 True Random Number Generators 8

4 Quantum Random Number Generators **9**
4.1 Trusted device QRNG 9
4.1.1 Decadimento radioattivo 9
4.1.2 Rumore elettronico 10
4.1.3 QRNG che utilizzano rilevatori di singoli fotoni 10
4.1.4 QRNG che utilizzano misuratori macroscopici di fotoni 12
4.2 Self testing QRNG 13
4.2.1 Self-testing Randomness expansion 13
4.2.2 Randomness amplification 14
4.3 Semi self testing QRNG 14
4.3.1 Source indipendent QRNG 14
4.3.2 Measurement-device-independent QRNG 15

5 Applicazioni dei QRNG nei protocolli crittografici tradizionali **16**
5.1 Sistemi di crittografia 16
5.2 Crittosistemi basati su RNGs 18
5.2.1 Algoritmi di cifratura 18
5.2.1.1 BloStream 18
5.2.1.2 Present 18
5.2.1.3 Chaos-based AES (CCAES) 19
5.2.2 Algoritmi di scambio chiavi 19
5.2.2.1 Chaos-based hybrid RSA (CRSA) 19

5.2.2.2 Diffie-Hellman con QRNG 20
5.3 Analisi costi/benefici dell'utilizzo di QRNG nei sistemi crittografici 21
5.3.1 Vantaggi 21
5.3.2 Costi 23
5.3.3 Considerazioni finali 23

6 Conclusione **25**

Elenco delle figure

3.1 Tipologie di generatori di numeri casuali. 7

4.1 QRNG con beam splitter o polarized beam splitter 11
4.2 QRNG basato sulla misurazione dello shot noise nello stato di vuoto utilizzando una rilevazione omodina 13

5.1 Key Schedule Algorithm (KSA) e fase di cifratura di un sistema crittografico basato su RNG [5]. 17
5.2 (a) Crittografia simmetrica; (b) Crittografia asimmetrica [5]. 17

Capitolo 1

Introduzione

I numeri casuali hanno un ruolo importante in diversi campi, come la crittografia, le simulazioni scientifiche o le lotterie. Ogni giorno, spesso inconsciamente, utilizziamo gli output di generatori di numeri casuali. Basti pensare alle One-time password (OTP), ai pin inseriti per operazioni bancarie, o ai Captcha che spesso ci viene richiesto di identificare per poter accedere ad un sito. Il loro utilizzo si basa sull'impredicibilità del loro valore, caratteristica che spesso non può essere garantita con i generatori attualmente utilizzati. In informatica, i generatori di numeri casuali (RNGs, Random Number Generators) sono basati su algoritmi di generazione pseudo-casuali, che creano il numero "espandendo" un seme in modo deterministico. Nonostante infatti la sequenza (binaria) in uscita da questi generatori sia costituita in modo bilanciato da 0 e 1, una correlazione tra il numero generato e il seme, seppur di difficile identificazione, esiste. Questo determinismo va ovviamente a diminuire la sicurezza della crittografia tradizionale, in quanto un attaccante, conoscendo sia l'algoritmo utilizzato che il seme e con a disposizione sufficienti risorse computazionali, sarebbe in grado di predire i numeri pseudo-casuali generati.

La soluzione per ottenere dei veri numeri casuali sarebbe quella di sfruttare la casualità associata a particolari fenomeni naturali come rumore atmosferico, rumore termico, sistemi caotici o rumore in circuiti elettrici. Tali fenomeni sono talmente complicati che possono essere sfruttati per generare, almeno apparentemente, numeri casuali non deterministici. Tuttavia, la qualità di tali valori è di difficile quantificazione e non risulta possibile scartare l'ipotesi che, in un futuro non troppo lontano, sia sviluppata una teoria che modelli precisamente questi fenomeni naturali, rendendo i numeri generati predicibili.

L'unico modo per essere certi della casualità dei valori generati è basarsi su fonti con casualità intrinseca ad esse associata. Come verrà descritto in questa trattazione, la meccanica quantistica è non deterministica di natura, quindi un sistema quantico risulta essere una fonte ideale per generare numeri realmente casuali. Inoltre, le leggi della meccanica quantistica possono essere utilizzate per quantificare la qualità dei valori generati. In questo documento verranno presentate le differenti categorie di generatori quantici di numeri casuali (Quantum Random Number Generators) e verranno esaminate possibili applicazioni degli stessi nel campo della crittografia, mostrando come il loro utilizzo potrebbe rinforzare la sicurezza di protocolli classici.

Capitolo 2

Prerequisiti

Nel seguente capitolo verranno illustrati i concetti e le proprietà chiave della meccanica quantistica e della casualità, necessari per comprendere al meglio il funzionamento dei generatori quantici di numeri casuali.

2.1 Superposition

Nella meccanica classica un oggetto fisico può avere un solo preciso valore per ogni sua proprietà fisica. Per esempio, un oggetto può essere in un posto nello spazio o in un altro, non in entrambi allo stesso momento. In termini di teoria dell'informazione, questo implica che un bit può solo possedere il valore 0 o il valore 1 in un dato istante, non entrambi.
Nella meccanica quantistica, invece, tale proprietà non risulta essere valida. Lo stato di un sistema fisico è descritto da un vettore e qualsiasi *superposition*, o più tecnicamente, qualsiasi combinazione lineare di due vettori o stati, descrive uno stato fisico perfettamente valido.

2.2 Casualità della misurazione

Effettuando una misurazione di una proprietà di un sistema quantico, il risultato sarà un singolo valore appartenente ad un insieme di valori discreti. Per esempio, in meccanica quantistica, è possibile misurare la polarizzazione di fotoni rispetto a diverse direzioni, utilizzando un filtro che permette al fotone di attraversarlo solo se questo è polarizzato nella stessa direzione del filtro. In particolare, per esempio, un filtro orizzontale permetterebbe solo ai fotoni polarizzati orizzontalmente (fotoni nello stato $|H\rangle$) di passare, mentre bloccherebbe quelli polarizzati verticalmente (fotoni nello stato $|V\rangle$). Preparando un fotone nello stato $(|H\rangle + |V\rangle)/\sqrt{2}$ e misurando il suo stato con un filtro, esso avrà esattamente il il 50% di possibilità di passare. Tale probabilità dipende dai coefficienti della combinazione lineare degli stati H e V. In questo caso la casualità della misurazione è genuina, nel senso che non esiste un modo di predire il risultato della misurazione prima di averla effettuata. Questo indeterminismo è il concetto chiave della meccanica quantistica e quello che permette, come vedremo, la costruzione di generatori di numeri realmente casuali.

2.3 Entanglement

Si definisce entanglement il fenomeno attraverso il quale si viene a creare una speciale connessione tra due elementi quantici. Due oggetti legati da questo rapporto presenteranno sempre una forte correlazione tra loro, anche a grandi distanze spaziali. Per esempio, misurando una proprietà di due fotoni "entangled", questa avrà lo stesso valore per entrambi. Gli scienziati fanno ancora fatica a spiegare questo strano fenomeno, ma numerosi esperimenti continuano a confermare l'esistenza di tale funzionamento nel mondo quantico.

2.4 Bell inequality

Il teorema di Bell incapsula diversi risultati ottenuti in fisica i quali portano alla conclusione che la meccanica quantistica non è compatibile con la teoria delle variabili locali nascoste valida per il mondo classico. Il termine "locale" si riferisce al principio di località, secondo cui una particella può essere influenzata solamente da elementi ad essa vicini e l'interazione mediante campi fisici può avvenire solamente a velocità inferiori a quella della luce. Lo stesso Bell, studiando il concetto di Entanglement osservato nel mondo quantico, ha dimostrato come le variabili di particelle di questo tipo devono essere "non locali" e che quindi esse violano il vincolo definito come Bell inquality secondo cui deve esistere una correlazione matematica tra gli output delle misurazioni di due particelle correlate quantisticamente.

2.5 Casualità

Al fine di comprendere il funzionamento e l'utilità di generatori di numeri casuali è prima necessario definire cosa sia esattamente una fonte di casualità. Per tale definizione saranno considerate sequenze di numeri, in particolare sequenze di 0 e 1. La casualità è strettamente legata all'impossibilità di predire i valori generati. Un semplice test per verificare se una sequenza di numeri è realmente casuale è quello di comprimere il valore con la compressione zip. Se la sequenza compressa risulta essere di dimensione inferiore rispetto a quella di partenza, allora lo strumento ha trovato un pattern ricorrente all'interno della stessa e si può quindi affermare che i valori generati non siano causali. Al fine di ottenere una fonte di valori realmente random essi dovrebbero quindi essere indipendenti gli uni dagli altri. Il problema con questa valutazione è che, considerando una stringa di lunghezza 5, le sequenze 00000, 11111 e 10011 sono tutte equamente probabili. Risulta perciò complicato essere certi della casualità di una serie di valori. I test attualmente utilizzati per misurare la casualità misurano diversi attributi delle sequenze generate, come la presenza di pattern ricorrenti o la distribuzione dei valori all'interno della serie di numeri, che dovrebbe essere uniforme.

Capitolo 3

Random Number Generators

In questo capitolo saranno descritti brevemente i generatori di numeri casuali attualmente utilizzati. La figura 3.1 riassume le varie tipologie di generatori di numeri casuali esistenti.

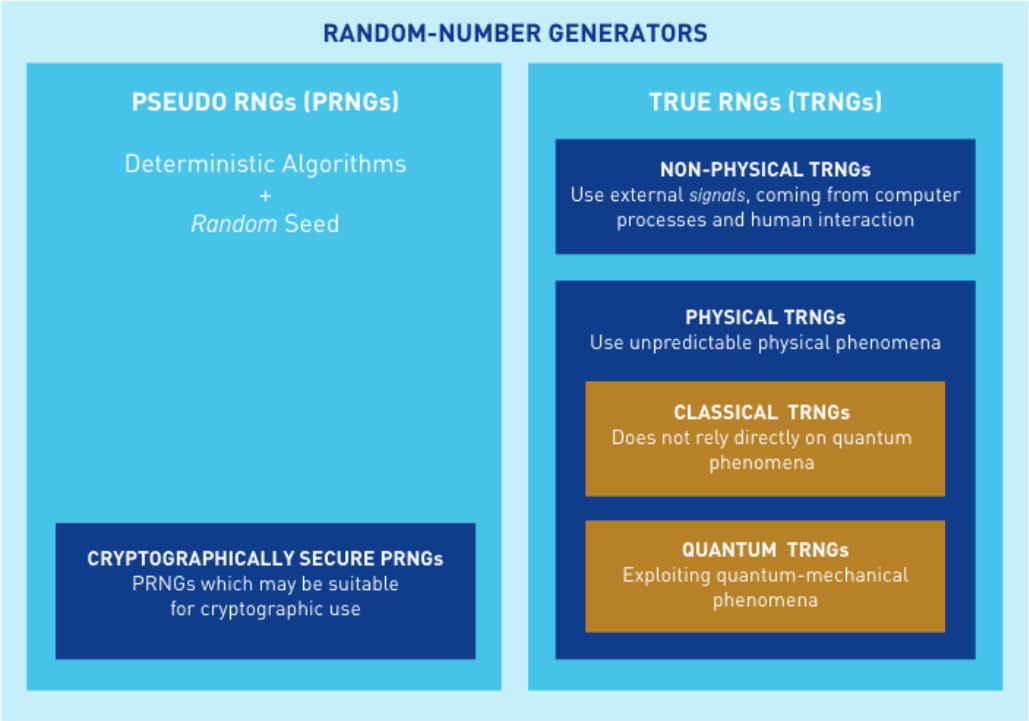

Figura 3.1: Tipologie di generatori di numeri casuali.

3.1 Pseudorandom-number generators

Un generatore di numeri pseudocasuali (PRNG) è un programma o una funzione che espande una breve sequenza di cifre (seme) in una sequenza di valori apparentemente casuali di lunghezza N, con N scelto a piacere. L'espansione del seme viene tuttavia eseguita attraverso algoritmi deterministici e risulta quindi possibile predire i valori generati se si conosce il seme utilizzato. Per questo motivo, i PRNG sono perfetti per utilizzi dove la sicurezza non è importante, come simulazioni scientifiche o previsioni meteo, ma rappresentano fonti di vulnerabilità quando utilizzati in crittografia o altre applicazioni dove la segretezza dei valori generati è un fattore chiave.

Esistono diversi metodi per generare valori pseudocasuali e altrettanti test statistici per quantificarne la qualità. Nel tempo sono stati sviluppati anche i cosiddetti CSPRNG, ossia dei generatori pseudocasuali più robusti per utilizzi crittografici. Sebbene alcuni di questi generatori siano considerati sicuri (affermazione supportata da numerose evidenze sperimentali), tale sicurezza si basa sulla complessità computazionale, metrica che perde valore con l'avvento della tecnologia quantica. Come già detto, il punto debole di questi generatori è il seme iniziale, che, se in possesso dell'attaccante, gli permetterebbe di predire tutti i valori generati. Per limitare questo problema, la sequenza che farà da seme è spesso ottenuta utilizzando un altro tipo di generatore di numeri casuali, un *True Random Number Generator*, che utilizza l'entropia derivante da diversi componenti del computer per generare il valore casuale.

3.2 True Random Number Generators

Questa categoria di generatori di numeri casuali si può suddividere, come mostrato in 3.1 in due macro aree: TRNG non fisici e TRNG fisici. I primi si basano su segnali provenienti da diversi componenti del computer o dall'interazione dell'utente con esso. Alcuni esempi di fonti di entropia in questo caso sono i movimenti del mouse, l'attività elettrica del disco fisso o valori istantanei dell'orologio di sistema. I TRNG fisici si possono invece ulteriormente suddividere in due gruppi: quelli classici e quelli quantici. I TRNG fisici classici utilizzano come fonte di entropia dei fenomeni fisici di complessa natura (non quantica), impossibili da modellare con gli strumenti e le conoscenze attualmente a disposizione. Alcuni esempi di tali fenomeni sono il lancio del dado, il rumore atmosferico, le radiazioni cosmiche o il rumore in circuiti elettrici. Questa classe di generatori presenta tuttavia un grande problema: la casualità dei valori generati si basa sulla mancanza di informazioni a disposizione, o, come i generatori pseudo-casuali, sulla limitata capacità computazionale a disposizione degli attaccanti. Quantificare la qualità di tali generatori è dunque complesso e nulla assicura che un più matura teoria fisica in futuro riesca a modellare perfettamente questi fenomeni. La peculiarità dell'ultima classe di generatori, quelli quantici, riguarda proprio questo aspetto: la casualità delle fonti quantiche è intrinseca e non si basa su assunzioni di complessità o mancanza di informazioni.

Capitolo 4

Quantum Random Number Generators

I generatori quantici di numeri casuali nascono dalla necessità di avere degli strumenti per ottenere numeri realmente random, dove la fonte di casualità non dipenda da una mancanza di informazioni sul sistema o fenomeno sfruttato ma sia genuina. Come sarà descritto in questo capitolo, con i QRNG risulta inoltre possibile verificare la fonte di casualità grazie alle leggi del mondo quantico.

I QRNG si possono suddividere in varie categorie in base alla fonte di casualità sfruttata e alla modalità di misurazione utilizzata. In primo luogo, però, i generatori quantici di numeri casuali si possono dividere in Trusted device QRNG, self-testing QRNG e semi self-testing QRNG, descritti di seguito. Per comprendere al meglio le prossime sezioni è opportuno fare una premessa: un QRNG è costituito da due componenti principali, una fonte di casualità e uno strumento per estrarre quest'ultima e trasformarla in un valore casuale.

4.1 Trusted device QRNG

I trusted device QRNG assumono che sia lo strumento utilizzato per l'estrazione dei valori, sia la fonte di casualità siano ben definiti e che rispettino alla perfezione il modello matematico a cui si riferiscono. La loro semplicità di funzionamento e di implementazione li rende adatti ad applicazioni pratiche, ma se una di queste due assunzioni non è rispettata la qualità dei numeri casuali non è verificata. Questa risulta essere una forte limitazione, soprattutto considerando che controllare sistemi quantici è un'operazione molto complessa.

Di seguito verranno elencate diverse tipologie di trusted device QRNG, discutendone il funzionamento e analizzandone le proprietà.

4.1.1 Decadimento radioattivo

Il decadimento radioattivo di particelle è stato uno dei primi fenomeni quantici sfruttati per la generazione di numeri casuali. La radioattività può essere spiegata solamente utilizzando il principio di indeterminazione della meccanica quantistica. I generatori sviluppati utilizzavano un tubo Geiger–Müller (GM) particolarmente sensibile e delle fonti ben definite di radiazioni α, β e γ. Il

tubo GM produce una pulsazione per ogni ionizzazione rilevata. La probabilità di decadimento in un intervallo di tempo dt è data da:

$$P(t)dt = \lambda_m e^{\lambda_m t} dt$$

dove λ_m è la costante di decadimento relativa alla fonte radioattiva considerata. Le pulsazioni formano una distrubuzione di Poisson e la frequenza di decadimento dipende da diversi fattori. I principali metodi per trasformare la rilevazione delle ionizzazioni in numeri casuali sono due: il metodo dell'orologio veloce e quello dell'orologio lento. La differenza tra i due è che, mentre nel primo la frequenza dell'orologio è maggiore di quella media di rilevazione, nel secondo avviene l'opposto. Di conseguenza, anche il funzionamento è differente. Nel metodo dell'orologio veloce ad essere convertito in numero binario è il numero di cicli dell'orologio tra una rilevazione e quella successiva, mentre in quello dell'orologio lento vengono contate le rilevazioni di ionizzazioni in un determinato lasso temporale. Per risolvere il problema della distribuzione di Poisson dei valori, proprietà che ovviamente non è desiderata in un generatore di numeri casuali, sono stati elaborati diversi metodi per renderla uniforme. Il più semplice consiste nel guardare la parità del valore generato per generare il valore realmente casuale (0 se il numero è pari, 1 se dispari). Nonostante siano state sviluppate diverse versioni di tali generatori, ognuno di essi è caratterizzato dalle medesime limitazioni. In primo luogo, controllare fenomeni di decadimento radioattivo comporta delle difficoltà in termini di sicurezza. Inoltre, eventi radioattivi possono influenzare il funzionamento degli strumenti utilizzati, andando ad intaccare la qualità dei valori generati e la frequenza di generazione degli stessi.

4.1.2 Rumore elettronico

Un'altra fonte di entropia individuata è il rumore presente in circuiti elettrici. Sono infatti stati sviluppati dei generatori che utilizzano componenti elettronici come resistenze e diodi come fonte di entropia. Il rumore viene tipicamente generato a causa della natura quantica delle cariche elettriche e il fenomeno quantico sfruttato è anche in questo caso il principio di indeterminazione. Il rumore rilevato viene successivamente amplificato e utilizzato per l'estrazione dei valori, per esempio confrontandolo con un valore soglia per generare bit casuali. Il rumore presente in questi sistemi si può tuttavia suddividere in due categorie: il rumore shot (shot noise), di natura quantica e il rumore termico, dovuto allo spostamento delle cariche elettriche a causa della temperatura ambiente. Idealmente si vorrebbe estrarre la casualità dallo shot noise in quanto fenomeno quantico, ma in pratica risulta difficile isolare quest'ultimo dal rumore termico. La difficoltà nel distinguere le due forme di rumore è il problema principale di questo tipo di generatori.

4.1.3 QRNG che utilizzano rilevatori di singoli fotoni

I QRNG spiegati in questa sezione e quelli illustrati nella successiva fanno parte della classe di generatori basata su fenomeni ottici, che sfruttano quindi la natura dei fotoni come fonte di casualità. I principali vantaggi di tali sistemi, rispetto a quelli finora analizzati, sono la facilità di implementazione e di funzionamento. Le fonti utilizzate sono LASER, LED e fonti di singolo fotone. La luce viene successivamente manipolata con elementi ottici e infine misurata. In questa sezione sono descritti i QRNG che utilizzano rilevatori di singoli fotoni, mentre nella successiva saranno analizzati i QRNG che sfruttano rilevatori macroscopici.

Qubit state
I QRNG basati sul qubit state generano casualità misurando i qubits in superposition. Il principio di meccanica quantistica alla base di questo approccio è quindi il collasso del vettore di stato al momento dell'osservazione. In particolare, come descritto nel capitolo dei prerequisiti, lo stato quantico di un fotone può essere una combinazione lineare di più stati, ossia il fotone può trovarsi in superposition. Per esempio, se un fotone è fatto passare in un beam splitter (BS), esso esisterà nella superposition degli stati *riflesso* ($|R\rangle$) e *trasmesso* ($|T\rangle$). Qualsiasi sistema quantico può esistere in una superposition degli stati della base e risulta relativamente semplice sfruttare questa proprietà e quella del collasso all'osservazione per generare numeri casuali. In base binaria, dove gli stati $|0\rangle$ e $|1\rangle$ rappresentano rispettivamente l'assenza e la presenza del fotone, è possibile definire lo stato $|1\rangle_1|0\rangle_2$ che indica la misurazione del fotone nel primo stato della base e lo stato $|0\rangle_1|1\rangle_2$ che indica la misurazione del fotone nel secondo. La superposition di un fotone sarebbe a questo punto descritta da:

$$\frac{|1\rangle_1|0\rangle_2 + |0\rangle_1|1\rangle_2}{\sqrt{2}}$$

Utilizzando dei rilevatori come mostrato in figura 4.1 è possibile generare un bit casuale ad ogni rilevazione, osservando uno dei due stati, ognuno con il 50% di probabilità.

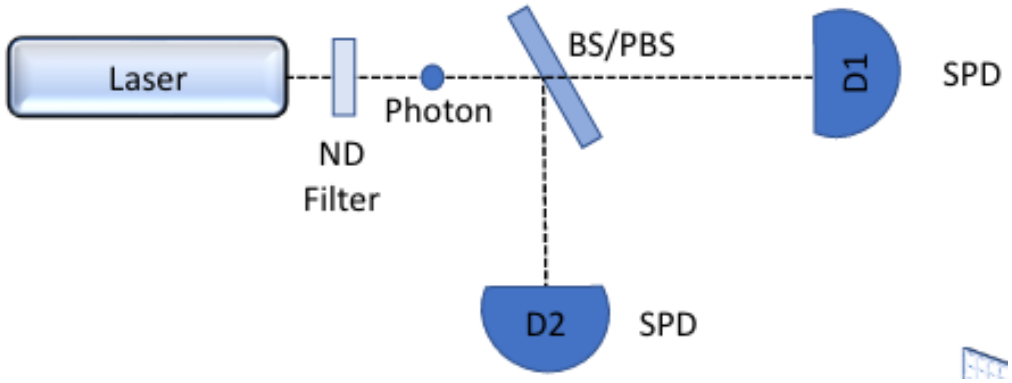

Figura 4.1: QRNG con beam splitter o polarized beam splitter

Diverse implementazioni sono state sviluppate con la tecnica appena presentata, ma i risultati hanno mostrato come la velocità di generazione sia limitata a decine di Mbps [17]. Tale tipologia di generatori ha infatti un problema causato dai rilevatori utilizzati. Questi ultimi sono infatti caratterizzati da un tempo morto a seguito di ogni rilevazione durante il quale non possono rilevare altri fotoni. Questo può portare alla creazione di correlazione tra i bit generati ed aumentare il tempo richiesto per la generazione. Per aggirare queste limitazioni sono stati sviluppati dei generatori che utilizzano combinazioni di un maggior numero di stati, in modo che i fotoni possano seguire più percorsi. A tal punto, la superposition di un fotone sarebbe descritta come di seguito:

$$\frac{|10...00\rangle + |01...00\rangle + ... + |00...01\rangle}{\sqrt{n}}$$

Una misurazione dello stato in un esperimento in cui il fotone può seguire n percorsi risulterebbe in $log_2(n)$ bit casuali. Tuttavia, la creazione e il controllo di un sistema di questo tipo risultano essere molto più complicati rispetto ai dispositivi descritti precedentemente.

Temporal mode
I QRNG temporali generano casualità dal tempo di arrivo dei fotoni. In questo senso, sono molto simili come funzionamento ai QRNG basati sul decadimento radioattivo descritti nella sezione 4.1.1. A differenza di questi ultimi, tuttavia, hanno una maggiore frequenza di generazione, in quanto la produzione di fotoni è più veloce. Una tipologia standard di questi generatori utilizza una fonte di fotoni, dei ricettori e un timer. Viene misurato il tempo impiegato da un fotone per arrivare ($t1$) e il tempo impiegato da quello successivo ($t2$). Tali valori vengono infine confrontati e viene generato un 1 se $t2 > t1$ e uno 0 se $t2 < t1$. La difficoltà principale con questi generatori è la precisione di misurazione del tempo impiegato dai fotoni. Un'alternativa anche in questo caso è utilizzare il metodo dell'orologio lento descritto per il decadimento radioattivo. In ogni caso, questi generatori non sono afflitti dal problema del tempo morto dei ricettori e riescono a raggiungere velocità di generazione di circa 109 Mbps [18].

4.1.4 QRNG che utilizzano misuratori macroscopici di fotoni

In questa tipologia di QRNG vengono misurate delle caratteristiche più classiche come l'intensità, l'ampiezza, la fase, ecc. Creare un modello per tali generatori risulta essere più complesso e deve essere prestata particolare attenzione per assicurarsi che la fonte di casualità utilizzata sia effettivamente quantica. I QRNG sviluppati con queste tecniche non sono tuttavia caratterizzati dal problema dei tempi morti dei SPD e riescono dunque a raggiungere velocità di generazione maggiori.

Vacuum noise
Questi QRNG utilizzano come fonte di casualità le fluttuazioni di punto zero del campo elettromagnetico. Nell'ottica quantica, lo stato di vuoto è rappresentato da una distribuzione Gaussiana su due dimensioni centrata nell'origine e con incertezza pari a 1/4. Per generare numeri casuali vengono misurate quindi le quadrature dell'ampiezza e della fase del campo elettromagnetico. Le misurazioni possono essere eseguite ripetutamente, generando numeri casuali. Nell'esperimento descritto in [20] viene utilizzata una rilevazione omodina dello shot noise dello stato di vuoto. In particolare, come mostrato in figura 4.2, dopo aver effettuato la prima misurazione dello shot noise il segnale viene manipolato per estrarre i bit casuali.

Questa tipologia di generatori è limitata in velocità dai rilevatori e, sebbene opportune misure siano state implementate per isolare lo shot noise, quest'ultimo non è l'unico rumore presente, fattore che potrebbe influenzare la casualità dei valori generati. Nonostante tali limitazioni esperimenti pratici di tali generatori sono riusciti a raggiungere frequenze di generazione pari a 2 Gbps [30].

Raman Scattering Lo scattering Raman o diffusione Raman consiste nella diffusione inelastica di fotoni e prevede sia uno scambio di energia che un cambio nella direzione della luce. Diversi QRNG sono stati sviluppati utilizzando questo fenomeno, estraendo entropia dall'ampiezza [22] o dalla fase [21] del campo in uscita.

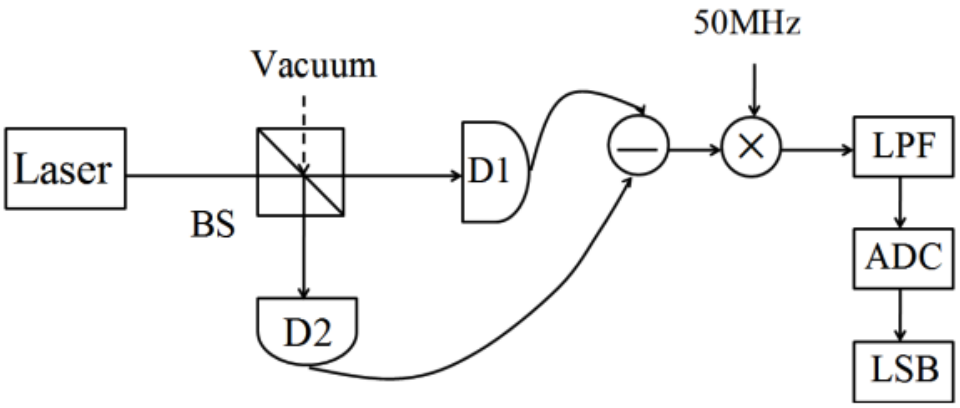

Figura 4.2: QRNG basato sulla misurazione dello shot noise nello stato di vuoto utilizzando una rilevazione omodina

4.2 Self testing QRNG

I metodi finora descritti per generare numeri casuali con tecnologie quantiche si basano su una fiducia assoluta negli strumenti utilizzati. Tuttavia, questi ultimi potrebbero non funzionare correttamente, per difetti tecnici o per manipolazione di attaccanti esterni. Sarebbe quindi opportuno aggiungere qualche forma di controllo per verificare che le sequenze generate siano effettivamente casuali. Ovviamente, anche tali sistemi di controllo potrebbero essere influenzati da entità malevole, non risolvendo quindi il problema nella sua interezza. Tuttavia, supponendo che il generatore non sia stato manipolato, semplici sistemi di controllo permetterebbero di verificare il corretto funzionamento del QRNG. Per esempio, alcuni generatori implementano metodi per verificare la distribuzione del segnale estratto dalla fonte di casualità, utilizzando solo le sequenze che superano il controllo per la generazione di numeri casuali. La distribuzione ottenuta è confrontata con quella che si sa essere esatta in natura. Altri esperimenti hanno anche sfruttato il concetto di entanglement per verificare la non località del sistema osservando violazioni della Bell inequality. Tale approccio permette di scartare la possibilità che il funzionamento del sistema sia deterministico. Sistemi di controllo di questa natura sono tuttavia complicati da implementare ed utilizzare e riducono drasticamente la velocità di generazione dei valori casuali.

I self testing QRNG seguono principalmente due approcci: l'espansione della casualità e l'amplificazione della stessa.

4.2.1 Self-testing Randomness expansion

Al fine di ottenere un dispositivo completamente indipendente dalla realizzazione fisica dello stesso, esso deve includere qualche forma di non località. Infatti, se il dispositivo funzionasse seguendo un algoritmo deterministico, l'attaccante dovrebbe solamente utilizzare dei dati scelti con cura al fine di superare i test statistici utilizzati. Dall'altro lato, se il protocollo fosse locale, l'attaccante potrebbe fornire al sistema dei valori di output per ogni input, rendendo il sistema di controllo inutile. Di conseguenza, l'utilizzo di uno stato caratterizzato da non località è necessario per

avviare la generazione di numeri casuali. Proprio per questo, tali protocolli sono visti come metodi di espansione di casualità piuttosto che di generazione della stessa.

Un protocollo di espansione di casualità mira a prendere in ingresso una sequenza casuale e generare una sequenza casuale di maggiore lunghezza, facendo in modo che la sicurezza del protocollo non dipenda dalla conoscenza del funzionamento interno dei dispositivi utilizzati. A tal fine, il protocollo assicura che i dispositivi violino la Bell inequality e quantifica la casualità estraibile dal processo in base alla violazione rilevata.

4.2.2 Randomness amplification

Un dispositivo di espansione della casualità appena descritto necessita di un seme iniziale di casualità. Senza di esso, il protocollo diventerebbe deterministico e un attaccante potrebbe predire i valori generati. Un protocollo di amplificazione di casualità evita il vincolo di possedere un seme uniforme. Una sequenza di valori casuali $x_1, x_2, ..., x_n$ è chiamata una fonte debole ϵ se

$$\epsilon \leq p(x_j|x_1, x_2, ..., x_{j-1}, e) \leq 1 - \epsilon$$

dove $p(x|y)$ è la probabilità condizionata e e rappresenta una variabile classica che influenza potenzialmente x_j. Data una fonte debole è impossibile produrre casualità certificata attraverso un processo classico. In [25], Colbeck e Renner hanno proposto un primo protocollo di amplificazione di casualità utilizzando tecnologie quantiche. Successivamente, tale risultato è stato migliorato mostrando come fosse possibile generare casualità perfetta da una sequenza casuale arbitrariamente debole [26].

Risulta quindi possibile utilizzare l'espansione di casualità per generare il seme perfettamente casuale necessario per avviare il processo di espansione di casualità.

4.3 Semi self testing QRNG

Durante l'implementazione di un QRNG è possibile che alcuni componenti del sistema siano meglio definiti di altri. In pratica, sarebbe possibile fidarsi solo di parte del sistema, soprattutto in situazioni in cui ci si vuole difendere unicamente da malfunzionamenti del sistema e non da ipotetici attacchi provenienti dall'esterno. I semi self testing QRNG sono dei generatori con caratteristiche intermedie tra i trusted QRNG e i self testing QRNG. In particolare, rappresentano un compromesso tra la bassa affidabilità con alte prestazioni dei primi e l'alta affidabilità con prestazioni scarse dei secondi.

4.3.1 Source indipendent QRNG

Nei source independent QRNG, la casualità della fonte è considerata inaffidabile, mentre il funzionamento degli strumenti di misurazione è ritenuto corretto. L'idea alla base di questi generatori, introdotta per la prima volta in [27], è quella di effettuare le misurazioni della fonte di casualità utilizzando due differenti basi, tra loro mutualmente indipendenti. Sfruttando queste ultime e un seme casuale iniziale è possibile ottenere un limite inferiore alla casualità generata. Negli ultimi anni sono stati sviluppati diversi modelli di SI-QRNG [28, 29], dimostrando come sia possibile raggiungere frequenze di generazione molto elevate fidandosi solo parzialmente del sistema utilizzato.

4.3.2 Measurement-device-independent QRNG

Alternativamente, esistono dei casi in cui la fonte di casualità è ben definita mentre gli strumenti di misurazione utilizzati sono considerati inaffidabili. In tali sistemi vengono utilizzati stati casuali per verificare il corretto funzionamento degli strumenti di misurazione. Il vantaggio di tali QRNG è la loro tolleranza nei confronti di malfunzionamenti dei sistemi di misurazione e a potenziali attacchi agli stessi. Lo svantaggio rispetto ai sistemi descritti precedentemente è che qualsiasi errore nella fonte di casualità porterebbe al malfunzionamento dell'intero sistema.

Capitolo 5

Applicazioni dei QRNG nei protocolli crittografici tradizionali

Gli algoritmi di cifratura vengono utilizzati su ampia scala, dunque è importante capire cosa li rende sicuri e cosa si potrebbe fare per incrementarne l'affidabilità. Tutti i protocolli crittografici tradizionali sono deterministici e di conseguenza reversibili, pertanto l'unica vera fonte di sicurezza è rappresentata da una parte non deterministica: una chiave o un valore a singolo utilizzo che dovrebbe essere casuale. La qualità e la dimostrabilità della casualità di questo insieme di bit diventano dunque cruciali per la sicurezza dell'intero sistema.

5.1 Sistemi di crittografia

Un algoritmo crittografico ha lo scopo di trasformare testo in testo cifrato per successivamente riconvertire il testo cifrato in quello originale. Questa procedura stabilisce un canale di comunicazione sicuro tra due entità in un dominio pubblico, come ad esempio Internet, dove sono presenti anche utenti non autorizzati o malintenzionati.

Ogni sistema crittografico include tre processi fondamentali: il Key Schedule Algorithm (KSA), l'Encryption Algorithm (EA) e il Decryption Algorithm (DA). Il KSA è un processo di generazione chiavi per la crittografia e decrittografia. Richiede una chiave di partenza come insieme di bit, ad esempio 40, 128, 192, 256, o anche un numero più significativo, per poi espanderli in base alle fasi di elaborazione o al numero di round progettati per l'algoritmo. Lo scopo del KSA è rendere la chiave abbastanza forte da non essere vulnerabile ad attacchi e fare in modo che nessuno possa risalire all'input di partenza. L'EA, algoritmo di cifratura, converte i dati in un formato illeggibile utilizzando la chiave prodotta dal KSA. Dall'altro lato, un algoritmo di decifratura (DA) prevede l'utilizzo della stessa o di un altra chiave per la decodifica, ovvero la riconversione del testo cifrato nei dati originali.

La sicurezza di un cifratore dipende da quanto la chiave utilizzata è vulnerabile ad attacchi di crittoanalisi. In aggiunta al KSA, che divide le chiavi in sottoparti, possono essere utilizzati numeri random per rendere le chiavi più robuste e complesse. Come mostrato in 5.1, i Random Number Generators (RNGs) forniscono i numeri random utilizzati per ottenere causalità nei sistemi di crittografia. L'incorporazione di RNG nei KSA e durante la fase di cifratura migliora diversi

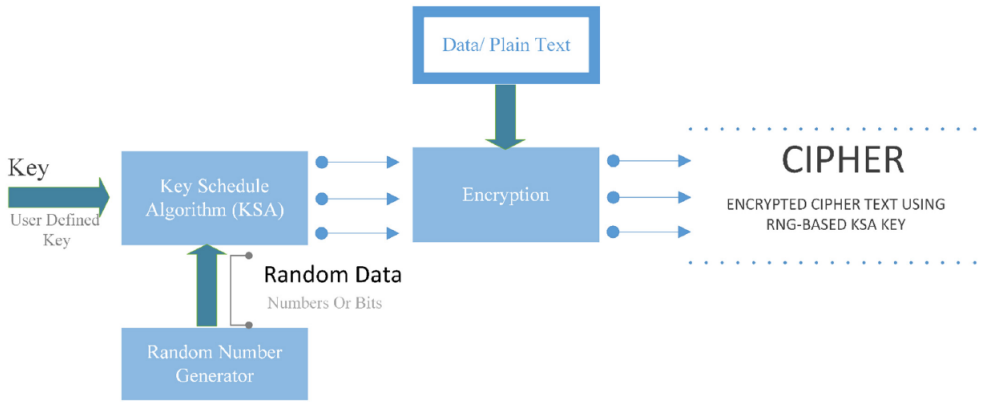

Figura 5.1: Key Schedule Algorithm (KSA) e fase di cifratura di un sistema crittografico basato su RNG [5].

parametri nei crittosistemi, che variano in base al tipo di generatore utilizzato, come vedremo nella sezione 5.2.

La differenza principale che sussiste tra i diversi algoritmi crittografici consiste nella relazione tra la chiave di cifratura e la chiave di decifratura. Gli algoritmi classificati come simmetrici utilizzano la stessa chiave privata per entrambe le fasi. Invece, la crittografia asimmetrica si avvale di una coppia di chiavi, una privata ed una pubblica, utilizzate per cifratura e decifratura.

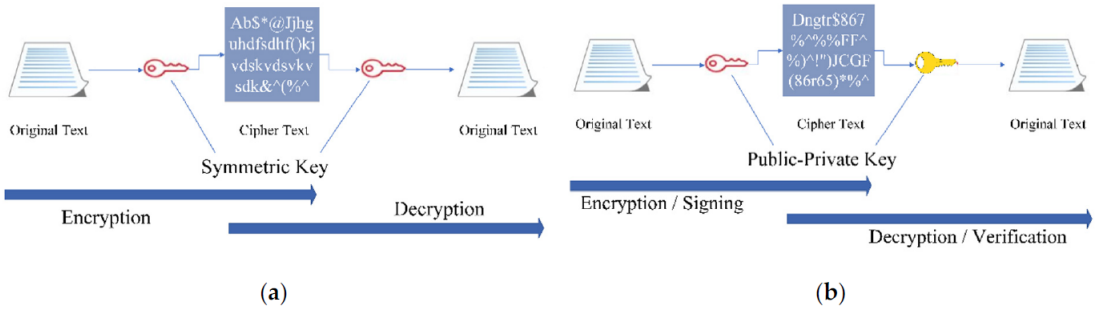

Figura 5.2: (a) Crittografia simmetrica; (b) Crittografia asimmetrica [5].

Un ultimo elemento necessario all'utilizzo degli algoritmi di cifratura, sono i metodi per lo scambio delle chiavi che definiscono dei protocolli sicuri per permettere a due entità di scambiarsi una chiave in totale segretezza. Tali protocolli consistono sostanzialmente nello scambio di una serie di messaggi tra due entità al fine di concordare i valori con cui costruire la chiave. Tali messaggi sono appositamente studiati per fare in modo che chiunque ascolti il canale non abbia modo di riprodurre la chiave finale generata dalle due entità. Esistono diversi metodi per la generazione di chiavi condivise, due tra i più famosi sono RSA e Diffie-Hellman che verranno illustrati più nel dettaglio nella sezione 5.2.2.

5.2 Crittosistemi basati su RNGs

In questo paragrafo verranno presentati diversi algoritmi di cifratura e di scambio chiavi nei quali sono stati integrati generatori di numeri random di vario genere. Lo scopo di questa sezione è quello di sottolineare i miglioramenti apportati grazie all'utilizzo di RNG all'interno di protocolli di sicurezza tradizionali. In ciascuno dei sistemi che verranno citati è possibile implementare generatori quantici di numeri random al fine di ottenere risultati ancora più vantaggiosi.

5.2.1 Algoritmi di cifratura

5.2.1.1 BloStream

Gli stream ciphers sono cifratori a chiave simmetrica ampiamente utilizzati per la cifratura di dati sensibili ad alta velocità, ad esempio quando i dati devono essere trasmessi tramite un canale di comunicazione o tramite un browser web. La struttura di base di uno stream cipher richiede la generazione di una keystream, ovvero una sequenza di cifre pseudocasuali. Per la generazione del testo cifrato, ogni cifra in chiaro viene combinata con una cifra della keystream, generalmente utilizzando "Exclusive or" (XOR). La stessa keystream viene poi posta in XOR con il testo cifrato per poter recuperare l'originale. Gli stream cipher sono stati progettati per avere determinate caratteristiche di sicurezza ed efficienza, tuttavia la loro velocità non è sufficientemente alta per poter evitare attacchi di sicurezza informatica. Apprensioni di questo tipo hanno spinto la comunità informatica a prediligere l'utilizzo di block cipher.

BloStream [8] è uno stream cipher ad alta velocità disegnato per essere più flessibile, veloce, casuale e sicuro rispetto agli stream cipher convenzionali che utilizzano lo XOR nella fase di combinazione. Tali risultati sono stati ottenuti grazie all'utilizzo di un PRNG ad alte prestazioni. Con un generatore di numeri pseudo casuali, utilizzando un algoritmo Rabbit mescolato con un combinatore con funzione di arrotondamento invertibile non lineare, diventa estremamente complesso calcolare la keystream o recuperare il testo in chiaro. Uno studio approfondito [8] ha dimostrato come l'introduzione di un PRNG in BloStream abbia perfezionato la velocità e i requisiti di memoria dell'algoritmo rispetto ad una serie di altri cifrari attualmente in uso tra cui RC4, Chameleon e Serpent. Inoltre, si è dimostrato considerevolmente resistente a numerosi attacchi informatici tra cui attacchi di forza bruta, attacchi statistici, attacchi differenziali, che sono stati applicati al cifratore.

5.2.1.2 Present

La Lightweight Cryptography (LWC) gioca un ruolo fondamentale nell'ottenimento di elevati livelli di sicurezza quando si hanno a disposizione bassi livelli di energia, memoria e capacità computazionale. Nell'ambito della crittografia simmetrica, troviamo adatti a questo scopo i block cipher. Un block cipher processa l'input suddividendolo in blocchi costituiti da un numero prefissato di bit; a differenza degli stream cipher che cifrano un elemento alla volta, questo algoritmo cifra gli elementi presenti in un blocco contemporaneamente.

L'Advanced Encryption Standard (AES) è il block cipher più diffuso e rappresenta una parte fondamentale di numerosi sistemi di sicurezza. Tuttavia, AES è utilizzato nei processori ad alte prestazioni e non risulta adatto alle piattaforme che presentano vincoli in termini di risorse. Di conseguenza, sono stati sviluppati block cipher più leggeri al fine di ottenere il miglior compromesso tra sicurezza e potenza. Tra questi troviamo PRESENT, un cifrario a blocchi che spicca

tra gli altri grazie alla sua efficienza hardware ed è anche standardizzato da ISO/IEC 29192-2. L'algoritmo si avvale dell'utilizzo di un modulo per la generazione di chiavi casuali da 80 bit che combina due generatori di numeri random: TRNG e PRNG. Tale modulo rende il valore chiave utilizzato nell'algoritmo impredicibile da parte di attaccanti malintenzionati, inoltre l'architettura PRESENT-TRNG-PRNG si è dimostrata [9] più performante rispetto ad altre configurazioni possibili dell'algoritmo. Dunque, l'implementazione di generatori di numeri random all'interno del cifratore PRESENT ha portato ad un miglioramento sia in termini di prestazioni che di sicurezza.

5.2.1.3 Chaos-based AES (CCAES)

Con l'avanzamento delle tecnologie di comunicazione, la trasmissione di immagini digitali è diventata un problema comune in quanto esse rappresentano il 70% dei dati scambiati via Internet. Gli algoritmi di cifratura svolgono un ruolo essenziale per assicurare uno scambio sicuro delle immagini, tuttavia per essere considerati affidabili devono non solo offrire elevati livelli di sicurezza ma anche un basso tempo di esecuzione. Per questi motivi, la maggior parte degli algoritmi di cifratura di immagini presenti sul mercato non sono adatti ad applicazioni nelle quali la sicurezza delle comunicazioni è cruciale.

CCAES [11] è un algoritmo di cifratura delle immagini che combina la tecnica di chaos sequence per la generazione di numeri casuali con l'algoritmo AES modificato. In questo metodo la chiave di cifratura viene generata dalla sequenza del caos di Arnold. Quindi, l'immagine di partenza viene crittografata utilizzando l'algoritmo AES modificato ed implementando le round key prodotte dal sistema caos. La teoria del caos è una branca della matematica che studia i sistemi estremamente complicati. In questi sistemi, l'applicazione di piccoli cambiamenti (apparentemente ignorabili) nell'input determina cambiamenti significativi nell'output. Se una persona non autorizzata non conosce i parametri di controllo e i valori iniziali corretti, non può indovinare la sequenza del caos. L'algoritmo AES è un block cipher simmetrico ed esiste in tre varianti: AES-128, AES-192, AES-256. Ogni variante, cifra e decifra i dati in blocchi da 128 bit utilizzando chiavi crittografiche rispettivamente di 128, 192 e 256 bit. L'algoritmo è basato sull'esecuzione di diversi round, ognuno dei quali consiste in diverse fasi di elaborazione che includono la sostituzione, la trasposizione e la combinazione del testo in chiaro al fine di trasformarlo in testo cifrato. L'AES modificato sviluppato per CCAES consiste in 10 round di cifratura e le operazioni di sostituzione ed integrazione delle colonne sono state rimpiazzate dalla conversione lineare e dalla somma dei valori dei pixel. Queste operazioni non solo riducono la complessità temporale dell'algoritmo ma aggiungono anche capacità di diffusione all'algoritmo CCAES, rendendo le imamgini crittografate più difficilmente predicibili. Tali fattori, ai quali va aggiunta la grande sensibilità ai valori di input di questo approccio, consentono all'algoritmo di resistere agli attacchi differenziali. Inoltre, il key space previsto dall'algoritmo è abbastanza grande da resistere ad attacchi di forza bruta. In conclusione, l'algoritmo CCAES si è dimostrato [11] molto più veloce rispetto all'algoritmo AES tradizionale e più sicuro e resistente agli attacchi informatici rispetto agli algoritmi di cifratura immagini comunemente utilizzati.

5.2.2 Algoritmi di scambio chiavi

5.2.2.1 Chaos-based hybrid RSA (CRSA)

RSA è un algoritmo di crittografia asimmetrico comunemente utilizzato per cifratura e scambio di chiavi. L'algoritmo prevede quattro step: la generazione delle chiavi, lo scambio delle stesse, una fase di cifratura e la conseguente decifratura. Un principio alla base di RSA è l'osservazione che è

pratico trovare tre interi positivi molto grandi e, d e n, tali che con esponenziazione modulare per tutti gli interi m (con $0 \leq m < n$):

$$(m^e)^d \equiv m \pmod{n} \tag{5.1}$$

e che conoscendo e e n, o anche m, può essere estremamente difficile trovare d. Inoltre, per alcune operazioni è conveniente che si possa cambiare l'ordine delle due esponenziazioni e che questa relazione implichi anche:

$$(m^d)^e \equiv m \pmod{n}. \tag{5.2}$$

RSA prevede una chiave pubblica e una chiave privata, la chiave pubblica può essere conosciuta da tutti e viene utilizzata per crittografare i messaggi. L'intenzione è che i messaggi crittografati con la chiave pubblica possano essere decifrati solo in un ragionevole lasso di tempo utilizzando la chiave privata. La chiave pubblica è rappresentata dagli interi n ed e, e la chiave privata dall'intero d.

Uno dei metodi più significativi utilizzati negli studi per il mantenimento della sicurezza dei dati è l'uso di sistemi caotici nella crittografia, come appena visto anche nel caso di CCAES. Infatti, a causa della sensibile dipendenza dalle condizioni iniziali e dai parametri di controllo dei sistemi caotici, è stato affermato che la scienza del caos e la crittografia sono strettamente accoppiate. CRSA [12] è un algoritmo di crittografia ibrido che utilizza algoritmi RNG e RSA chaos-based. Per lo sviluppo di questo algoritmo è stato sviluppato un nuovo sistema caotico con una complessa struttura dinamica, al fine di evitare problemi di sicurezza legati all'utilizzo di sistemi con una struttura dinamica già ben conosciuta da tutti. Su tale sistema è stato disegnato il generatore di numeri random, sottoposto a test NIST e FIPS [12] che rappresentano gli standard più elevati per l'affidabilità dei numeri casuali generati. L'algoritmo CRSA nel suo complesso è stato utilizzato in applicazioni crittografiche ed i risultati raccolti sono stati confrontati con quelli prodotti dall'algoritmo RSA tradizionali privo di RNG chaos-based. L'analisi [12] dimostra che CRSA dispone di un più elevato livello di sicurezza, in particolare sono stati migliorati i seguenti parametri: istogramma, correlazione, sensibilità delle chiavi, key space ed entropia delle informazioni.

5.2.2.2 Diffie-Hellman con QRNG

Il protocollo di scambio chiavi Diffie-Hellman offre una soluzione al seguente problema: due entità vogliono condividere una chiave segreta, ma il loro canale di comunicazione non è sicuro e dunque tutte le informazioni potrebbero essere intercettate da un utente malintenzionato. Il primo passo dell'algoritmo prevede che le due entità, Alice e Bob, si accordino per un numero intero p arbitrariamente grande, ed un intero $root = g \; modulo \; p$. I due numeri sono resi pubblici. Nel secondo passo, Alice sceglie un intero a che mantiene segreto e Bob fa la medesima cosa selezionando un intero b. Alice e Bob usano il numero segreto per calcolare:

$$A \equiv g^a \bmod p, \; B \equiv g^b \bmod p. \tag{5.3}$$

Nell'ultimo step, le due parti si scambiano i valori calcolati ed alla fine saranno entrambi in possesso della medesima chiave di cifratura, ottenuta come di seguito:

$$A : B^a \bmod p, \; B : A^b \bmod p. \tag{5.4}$$

Nell'esperimento riportato dal documento [4], è stato utilizzano un Quantum Random Number Generator (QRNG) per il calcolo dei numeri segreti a e b. Il QRNG utilizzato è Quantis, prodotto da

IdQuantique Company, Svizzera. Quantis è un generatore di numeri casuali quantistici che sfrutta un processo quantistico ottico come fonte di casualità, secondo il processo illustrato nella sezione 4.1.3. I generatori di numeri casuali quantistici hanno il vantaggio rispetto alle fonti di casualità convenzionali di essere invulnerabili alle perturbazioni ambientali e di consentire la verifica dello stato in tempo reale. Inoltre, i QRNG, come sottolineato più volte in questa trattazione, sono fonti di numeri casuali reali utilizzabili in applicazioni con gli standard di sicurezza più rigorosi. Questi fattori rendono Diffie-Hellman potenziato con un QRNG un sistema crittografico praticamente non vulnerabile.

5.3 Analisi costi/benefici dell'utilizzo di QRNG nei sistemi crittografici

5.3.1 Vantaggi

Come appena illustrato, i numeri casuali vengono utilizzati nei sistemi crittografici come semi per la generazione di chiavi, sia nel caso degli algoritmi di cifratura sia per quelli di scambio chiavi. L'affidabilità delle chiavi generate dipende dalla casualità del seme di input, portando l'intera sicurezza del sistema a dipendere anche dai RNG. Nella sezione precedente abbiamo sottolineato i vantaggi apportati dall'utilizzo di un qualsiasi tipo di generatore di numeri random all'interno dei crittosistemi classici. In Tabella 5.1 presenteremo invece i vantaggi dell'utilizzo di un QRNG rispetto ai RNG tradizionali.

Ciò che separa i QRNG dai generatori classici di numeri random è la capacità di generare numeri puramente casuali. I PRNG e i TRNG classici risultano vulnerabili agli attacchi informatici per la loro predicibilità. Infatti, i numeri casuali prodotti dai tradizionali RNG si basano su un input che risulta essere predicibile, così come l'output generato. Di conseguenza, tutti i numeri generati possono essere riprodotti successivamente anche grazie a tecniche innovative di machine learning ed attacchi operati da computer quantici. Questa problematica viene superata dai QRNG che sono basati su fonti intrinsecamente random.
I vantaggi dell'integrazione di un QRNG in un sistema crittografico possono essere riassunti dai seguenti punti:

- si affidano alle caratteristiche quantistiche per generare una casualità genuina;
- la fonte di casualità è imprevedibile e controllata dal processo quantistico. Ciò è in netto contrasto con i tradizionali RNG che si basano su sconosciuti, ma, in principio, conoscibili dati impliciti preesistenti in un dispositivo fisico, informazioni che potrebbero anche essere parzialmente impiantate;
- la certificazione e la convalida sono aiutate dai processi fisici di natura relativamente semplice alla base dei QRNG;
- è possibile ed estremamente efficace il monitoraggio in tempo reale della sorgente di entropia;
- tutti gli attacchi alla sorgente di entropia sono rilevabili;
- sono inespugnabili da computer quantistici.

Tutti i punti appena elencati sottolineano che i generatori quantici di numeri random sono essenzialmente sicuri.

Proprietà	**Tradizionali**	**Quantum**
Sorgente di entropia	Casualità basata sulla complessità del processo e ignoranza parziale.	Casualità genuina.
Facilità di certificazione	Capacità limitata di certificare il processo fisico sottostante, essendo questo intrinsecamente complesso. Certificazione della qualità della produzione basata su prove standard.	Può convalidare i processi fisici sottostanti. Certificazione della qualità della produzione basata su prove standard.
Resistenza alla manomissione	Una certa capacità di eseguire un controllo sullo stato della fonte di entropia.	Controllo integrato basato sulla semplicità del processo e più sensibile alla manomissione. Le versioni Device-Independent offrono la massima resistenza contro la manomissione della sorgente di entropia, anche da parte dei fornitori stessi.
Qualità dell'entropia	Vari gradi. Il processo sottostante utilizzato come fonte di entropia può funzionare in un regime fisico in cui vi sono grandi bias e correlazioni relativamente elevate, ovvero bassa entropia.	Elevata entropia sin dall'inizio basata sulla natura quantica della sorgente.
Velocità	Può essere molto alta e possono essere combinate diverse fonti per ottenere tassi ancora più elevati.	Alta, anche a causa della qualità dell'entropia iniziale, ma le implementazioni Device-Independent possono essere lente.
Dimensioni	Può essere molto piccolo e integrato in un chip, ad esempio sfruttando una fonte di casualità come il rumore termico.	Varia in modo sostanziale, passando da incorporabile negli smartphone a dimensioni comparabili con una stanza nel caso di QRNG Device-Independent basati sulla non località.

Tabella 5.1: Confronto tra generatori di numeri casuali tradizionali e quantici [7]

5.3.2 Costi

Attualmente, il costo delle tecnologie QRNG è relativamente alto rispetto ad alternative come i tradizionali TRNG e PRNG, per i quali le implementazioni sono spesso incluse gratuitamente nei processori o come parte dei sistemi operativi. Si crea dunque una barriera alla creazione di un mercato di QRNG a causa della pletora di offerte alternative a basso costo che forniscono numeri casuali con vari gradi di sicurezza. I QRNG si presentano come la scelta di sicurezza più elevata; tuttavia, agli occhi dei designer di prodotti che cercano di soddisfare un particolare standard di sicurezza, se l'utilizzo di un PRNG consente loro di accedere al rispettivo mercato non risulta conveniente spendere denaro aggiuntivo per soddisfare il livello di sicurezza più elevato.

Un altro fattore di costo da tenere in considerazione prima di utilizzare generatori quantici di numeri casuali riguarda la velocità e le dimensioni. Sino ad ora abbiamo parlato della proprietà unica dei QRNG di certificare che i numeri generati siano validi e privati. Tuttavia, questa peculiarità è propria solo dei generatori quantici self-tested. Questo tipo di dispositivi, sebbene dispongano delle caratteristiche che si cercano in un generatore quantico, presentano anche degli svantaggi. Infatti, come discusso nella sezione 4.2, i self-tested QRNG possono essere molto ingombranti e raggiungere velocità non sufficienti per soddisfare i criteri di molti sistemi. Dall'altro lato, le altre tipologie di QRNG riescono a sorpassare tali problematiche, ma possiedono una limitazione fondamentale: è impossibile per l'utente verificare che i numeri generati siano genuinamente random.

Infine, un inibitore all'integrazione di QRNG nei sistemi crittografici riguarda le certificazioni. Per le certificazioni dei prodotti crittografici ai governi statunitense e canadese viene utilizzato lo standard di valutazione della sicurezza FIPS140. Tale standard è anche ufficiosamente utilizzato dal settore finanziario del Nord America per prendere decisioni in merito all'acquisto di moduli di sicurezza hardware (HSM). FIPS140 stabilisce un livello di sicurezza minimo per le implementazioni crittografiche e si concentra sul test dei prodotti rispetto a un set standard di input e sulla valutazione degli output e del comportamento del prodotto. I fornitori tendono a concentrarsi solo sull'insieme minimo definito di funzionalità di sicurezza per ottenere il livello di certificazione che stanno perseguendo, poiché l'aggiunta di funzionalità aggiuntive oltre a ciò comporta un rischio negativo che il prodotto possa non superare la valutazione. A meno che i QRNG non siano specificamente aggiunti e richiesti dallo standard FIPS140, i fornitori tenderanno a scegliere il percorso meno costoso per la certificazione. Ciò potrebbe inibire l'adozione di QRNG per i settori governativi nel mercato nordamericano.

5.3.3 Considerazioni finali

Tenendo in considerazione i vantaggi ed i costi appena discussi, ne risulta che i QRNG potrebbero essere la soluzione ideale in molte circostanze grazie alle loro proprietà fondamentali, ma non in tutte. L'integrazione di QRNG è particolarmente adatta nei casi in cui i guadagni ottenuti in termini di sicurezza sono sufficienti a giustificare i maggiori costi e la velocità di generazione inferiore che li caratterizza. In particolare, l'integrazione di generatori quantici di numeri casuali nei sistemi crittografici tradizionali risulta vantaggiosa nel caso di applicazioni che proteggono asset di alto valore e sistemi critici: queste potrebbero includere informazioni riservate (mediche e finanziarie) da condividere in forma crittografata o autenticazione per l'accesso ad applicazioni e database strategici presso le forze armate, a livello governativo o aziendale. La casualità fornita dai QRNG Device-Independent può assumere un'importanza particolare per tali risorse o sistemi. Dall'altro lato, le proprietà uniche dei QRNG non sono sempre richieste. Per le applicazioni in cui la segretezza non è un problema, come l'esecuzione di simulazioni, il premio pagato per i QRNG potrebbe non

essere un valore giustificabile. Inoltre, nel caso in cui i protocolli di sicurezza prevedano un algoritmo di scambio chiavi nel quale entrambe le parti giocano un ruolo fondamentale nella creazione della chiave condivisa, per assicurare i benefici dell'utilizzo della tecnologia quantica per la generazione di numeri casuali è fondamentale che entrambe le entità in questione dispongano di un QRNG. Ad esempio, nel caso dell'algoritmo di key exchange Diffie-Hellman, se un attaccante dispone di un certo numero di informazioni e riesce a risalire ad uno dei due numeri casuali generati dalle rispettive entità, questo gli basterà per compromettere il sistema, anche se l'altro numero rimane sconosciuto.

Capitolo 6

Conclusione

In questa trattazione abbiamo presentato una panoramica sulle diverse tipologie di RNG ed il loro ruolo nei sistemi crittografici. Inoltre, sono state descritte le diverse tipologie di QRNG ed è stata effettuata un'analisi sui vantaggi che questi potrebbero apportare se integrati negli algoritmi di cifratura e scambio chiavi attualmente in uso.

Lo studio ha dimostrato che i generatori di numeri random forniscono numeri casuali fondamentali per la sicurezza dei crittosistemi e ne migliorano la sicurezza rendendo i diversi cifratori robusti e resistenti ai vari attacchi informatici. Inoltre, i Quantum Random Number Generator (QRNG) forniscono una casualità genuina, al contrario dei PRNG e TRNG classici. Esistono diverse categorie di generatori quantici, distinte in base alla dipendenza o meno del dispositivo da specifiche implementazioni fisiche. Tali categorie sono: trusted device QRNG, self-testing QRNG e semi self-testing QRNG.

I QRNG sono uno strumento relativamente emergente al quale va data la giusta considerazione dati i valori aggiunti che le sue proprietà uniche possono apportare al mondo della crittografia. La proprietà fondamentale che caratterizza i QRNG è la capacità di produrre numeri puramente random e, in alcune varianti, di poter certificare l'affidabilità dei valori prodotti. Tali caratteristiche rendono la considerazione dei QRNG particolarmente interessante per quelle applicazioni dove vi è la necessità di proteggere asset di alto valore o accedere a sistemi critici. Sebbene quindi per il momento i QRNG sviluppati non possano sostituire i RNG in ogni settore a causa della maggiore complessità e delle prestazioni solitamente inferiori, non va dimenticato che la tecnologia quantica è in continuo sviluppo e in un futuro non troppo lontano nuove implementazioni potrebbero trovare soluzione ai problemi descritti in questa trattazione.

Bibliografia

[1] V. Mannalath, S. Mishra, and A. Pathak, "A Comprehensive Review of Quantum Random Number Generators: Concepts, Classification and the Origin of Randomness," Mar. 2022. Number: arXiv:2203.00261 arXiv:2203.00261 [quant-ph].

[2] X. Ma, X. Yuan, Z. Cao, B. Qi, and Z. Zhang, "Quantum random number generation," *npj Quantum Information*, vol. 2, p. 16021, Nov. 2016. arXiv:1510.08957 [quant-ph].

[3] M. Stipčević, "Quantum random number generators and their use in cryptography," in *2011 Proceedings of the 34th International Convention MIPRO*, pp. 1474–1479, May 2011.

[4] G. Mogos, "Use quantum random number generator in Diffie-Hellman key exchange protocol," in *2016 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pp. 1–6, May 2016.

[5] A. Saini, A. Tsokanos, and R. Kirner, "Quantum Randomness in Cryptography—A Survey of Cryptosystems, RNG-Based Ciphers, and QRNGs," *Information*, vol. 13, p. 358, Aug. 2022. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.

[6] "QKD Technology."

[7] D. M. Piani, D. M. Mosca, and B. Neill, "Quantum Random-Number Generators: Practical Considerations and Use Cases," p. 38.

[8] D. Kashmar and E. S. Ismail, "Blostream: A high speed stream cipher," *Journal of Engineering Science and Technology*, vol. 12, pp. 1111–1128, Apr. 2017.

[9] T. Kowsalya, R. Babu, B. Parameshachari, A. Nayyar, and R. Mehmood, "Low Area PRESENT Cryptography in FPGA Using TRNG-PRNG Key Generation," *Computers, Materials & Continua*, vol. 68, no. 2, pp. 1447–1465, 2021. Number: 2 Publisher: Tech Science Press.

[10] M. Mishra and V. Mankar, "Text Encryption Algorithms based on Pseudo Random Number Generator," *International Journal of Computer Applications*, vol. 111, pp. 1–6, Feb. 2015.

[11] A. Arab, M. J. Rostami, and B. Ghavami, "An image encryption method based on chaos system and AES algorithm," *The Journal of Supercomputing*, vol. 75, pp. 6663–6682, Oct. 2019.

[12] Çavuşoğlu, A. Akgül, A. Zengin, and I. Pehlivan, "The design and implementation of hybrid RSA algorithm using a novel chaos based RNG," *Chaos, Solitons & Fractals*, vol. 104, pp. 655–667, Nov. 2017.

[13] A. Kr.Banthia and N. Tiwari, "Image Encryption using Pseudo Random Number Generators," *International Journal of Computer Applications*, vol. 67, pp. 1–8, Apr. 2013.

[14] I. Q. Technology, "New IQT Research Report: Quantum Random Number Generators will become a $7.2 Billion Market by 2026," Jan. 2021.

[15] R. Uppal, "Quantum (QRNG) or True Random Number Generator (TRNG) technology for post quantum cryptography, Mobile and IoT security and impenetrable encryption of military communications."

[16] "EU H2020 Project "QUPIC (Ultra-fast and Cost-effective Quantum Random Number Generator Photonic..)": description, partecipants, costs and EC-fundings."

[17] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger, "A fast and compact quantum random number generator," *Review of Scientific Instruments*, vol. 71, pp. 1675–1680, Apr. 2000. Publisher: American Institute of Physics.

[18] Y.-Q. Nie, H.-F. Zhang, Z. Zhang, J. Wang, X. Ma, J. Zhang, and J.-W. Pan, "Practical and fast quantum random number generation based on photon arrival time relative to external reference," *Applied Physics Letters*, vol. 104, p. 051110, Feb. 2014. Publisher: American Institute of Physics.

[19] Q. Yan, B. Zhao, Q. Liao, and N. Zhou, "Multi-bit quantum random number generation by measuring positions of arrival photons," *Review of Scientific Instruments*, vol. 85, p. 103116, Oct. 2014. Publisher: American Institute of Physics.

[20] Y. Shen, L. Tian, and H. Zou, "Practical quantum random number generator based on measuring the shot noise of vacuum states," *Phys. Rev. A*, vol. 81, June 2010.

[21] P. J. Bustard, D. Moffatt, R. Lausten, G. Wu, I. A. Walmsley, and B. J. Sussman, "Quantum random bit generation using stimulated Raman scattering," *Optics Express*, vol. 19, pp. 25173–25180, Dec. 2011. Publisher: Optica Publishing Group.

[22] P. J. Bustard, D. G. England, J. Nunn, D. Moffatt, M. Spanner, R. Lausten, and B. J. Sussman, "Quantum random bit generation using energy fluctuations in stimulated Raman scattering," *Optics Express*, vol. 21, pp. 29350–29357, Dec. 2013. Publisher: Optica Publishing Group.

[23] "What is the Advanced Encryption Standard (AES)? Definition from SearchSecurity."

[24] R. Colbeck and A. Kent, "Private randomness expansion with untrusted devices," *Journal of Physics A: Mathematical and Theoretical*, vol. 44, p. 095305, Feb. 2011. Publisher: IOP Publishing.

[25] "Free randomness can be amplified | Nature Physics."

[26] R. Gallego, L. Masanes, G. De La Torre, C. Dhara, L. Aolita, and A. Acín, "Full randomness from arbitrarily deterministic events," *Nature Communications*, vol. 4, p. 2654, Oct. 2013. Number: 1 Publisher: Nature Publishing Group.

[27] G. Vallone, D. Marangon, M. Tomasin, and P. Villoresi, "Quantum Randomness Certified by the Uncertainty Principle," *Physical Review A*, vol. 90, p. 052327, Nov. 2014. arXiv:1401.7917 [quant-ph].

[28] D. G. Marangon, G. Vallone, and P. Villoresi, "Source-Device-Independent Ultrafast Quantum Random Number Generation," *Physical Review Letters*, vol. 118, p. 060503, Feb. 2017. Publisher: American Physical Society.

[29] M. Avesani, D. G. Marangon, G. Vallone, and P. Villoresi, "Source-device-independent heterodyne-based quantum random number generator at 17 Gbps," *Nature Communications*, vol. 9, p. 5365, Dec. 2018. Number: 1 Publisher: Nature Publishing Group.

[30] T. Symul, S. M. Assad, and P. K. Lam, "Real time demonstration of high bitrate quantum random number generation with coherent laser light," *Applied Physics Letters*, vol. 98, p. 231103, June 2011. Publisher: American Institute of Physics.