Funzioni Hash Crittografiche: confronto tra Poseidon e SHA-256

Corso di Laurea in Informatica - Macroarea di Scienze Matematiche, Fisiche e Naturali

Laureando: Mihai Alexandru Sandu

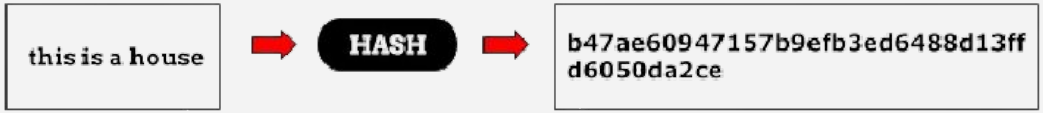
Relatore: Francesco Pasquale

Anno Accademico: 2024-2025

Le Funzioni Hash

Le funzioni hash sono fondamentali in crittografia perché permettono di trasformare input di lunghezza arbitraria in output a lunghezza fissa, il *digest*.

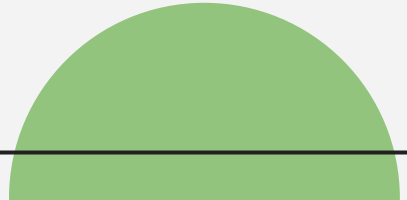
$H: \{0, 1\}^* \rightarrow \{0, 1\}^n$



Proprietà fondamentali

Sia H una funzione hash crittografica:

- **One-wayness**: Dato il digest, è computazionalmente difficile risalire all'input.
- **Second preimage resistance**: Dato un input x e il suo digest H(x), è difficile trovare $x' \neq x$ tale che $H(x) = H(x')$.
- **Collision resistance**: È difficile trovare due input distinti con lo stesso digest.



Applicazioni

Le Funzioni Hash Crittografiche vengono utilizzate in molti ambiti informatici: Checksum, Firme, Blockchain, Memorizzazione Password, ...

Nel tempo sono state ideate diverse funzioni hash:
- MD5: ad oggi deprecato
- SHA-256: ancora sicura utilizzata in protocolli come TLS/SSL
- RIPEMD-160: ancora sicura utilizzata in Bitcoin

Poseidon

Poseidon è una nuova Funzione Hash proposta nel 2019 da Lorenzo Grassi et al., che utilizza un struttura Sponge.

È progettata per essere efficiente su circuiti aritmetici nei campi primi finiti, rendendola ideale per blockchain come Ethereum.

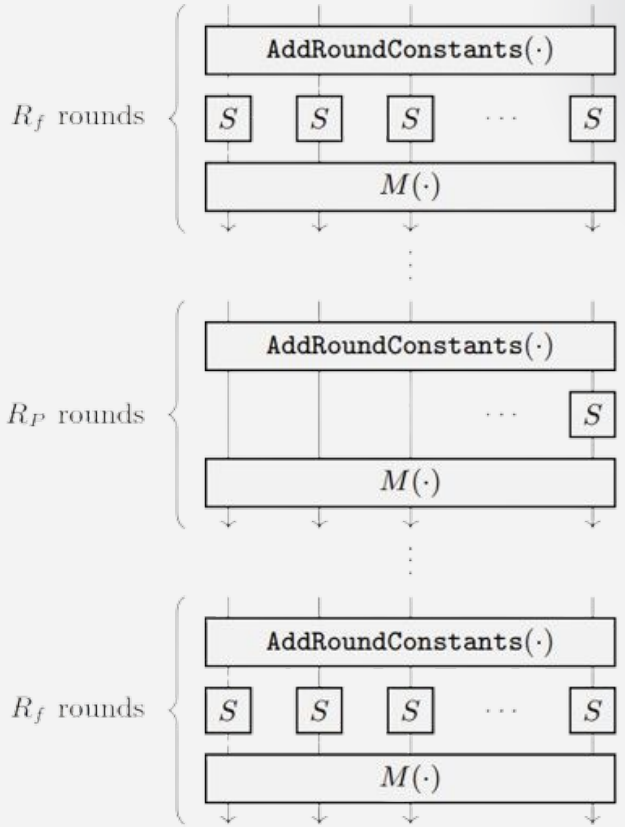
È ottimizzata per le *Zero-Knowledge Proofs*,

cioè tecniche crittografiche che permettono
ad un **prover** di dimostrare a un **verifier** di conoscere
un'informazione segreta, senza rivelarla.

Poseidonπ

Le permutazioni di poseidon sono divise in round parziali e round full. Ogni round è composto da 3 passi principali:

1. Addizione di **Costanti** di round.
2. Applicazione dell'**S-box**: unica componente non lineare,
   $S(x) = x^{\alpha}$
3. Mix Layer: Utilizzo di **Matrici MDS** (Maximum Distance Separable)



Bounty Program

Per incentivare la ricerca di vulnerabilità in Poseidon, e garantire la sua sicurezza, è stato istituito un Bounty Program con un montepremi di ben

$130.000

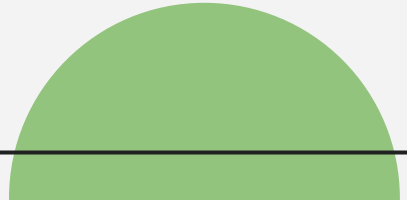
che premia chiunque riesca a rompere l'algoritmo o versioni semplificate di esso.

Implementazione di Poseidon

Il lavoro svolto non ha come obiettivo la valutazione delle prestazioni computazionali bensì testare la robustezza di Poseidon.

È stato scelto Poseidon perché, pur essendo un algoritmo recente, è già usato in contesti molto importanti come Ethereum.

Poiché non esistevano librerie funzionanti utilizzabili, è stata sviluppata un’implementazione da zero, seguendo le specifiche e linee guida del paper originale.



Attacchi Noti

Nonostante la robustezza delle funzioni hash, esistono tecniche crittoanalitiche avanzate (lineare, differenziale e algebrica), per testarne la robustezza.

Poiché queste tecniche sono complesse e richiedono l'individuazione di eventuali debolezze, sono stati svolti diversi test statistici:

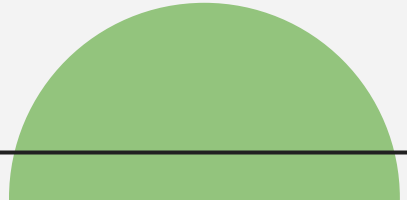
- Avalanche Effect
- Collision Resistance
- Uniformità & Chi-square
- Bit Positional Analysis & Shannon Entropy
- Hash Pattern Analysis & Autocorrelazione

Test Statistici

Per brevità andremo a vedere i test che hanno prodotto i risultati più significativi:

- Collision Resistance
- Uniformità & Chi-square
- Hash Pattern Analysis & Autocorrelazione

Per eseguire questi test sono stati generati casualmente **1.000.000** di file da 1KB e computati dalle funzioni hash Poseidon e SHA-256 per confrontare i risultati.



Collision Resistance

La collision resistance misura la difficoltà di trovare due input distinti con lo stesso digest. Normalmente, questi test vengono effettuati sull'intero digest.

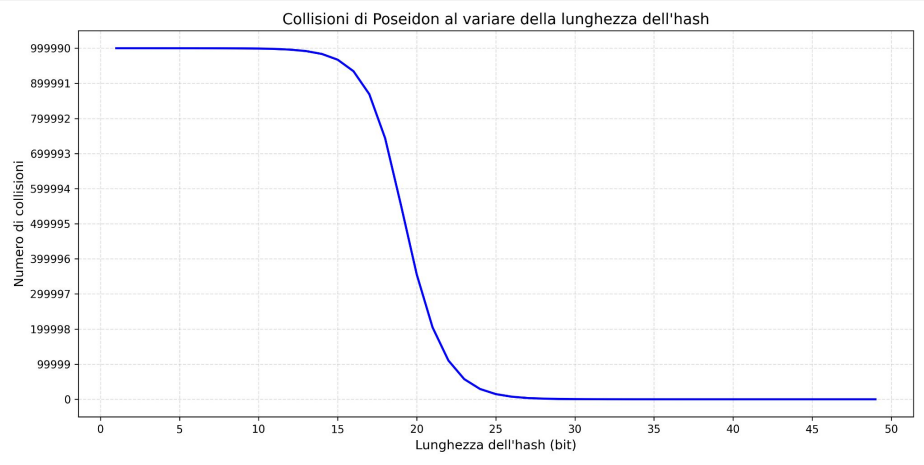
Con un digest di n bit, servono circa **$2^{n/2}$ input distinti** per avere ~50% di probabilità di collisione (Paradosso del Compleanno).

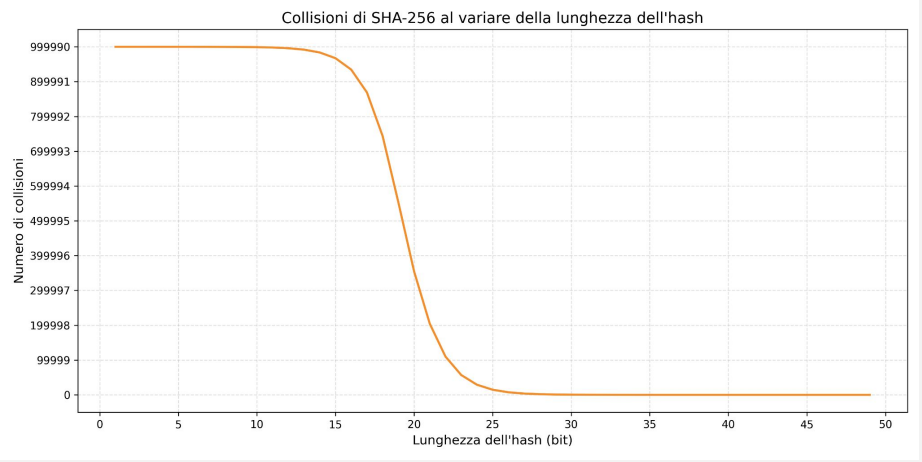
Con digest lunghi 256 bit come Poseidon e SHA-256, è impossibile osservare collisioni su dataset "piccoli".

Si è deciso quindi di analizzare non le collisioni sull'intero digest ma su porzioni di esso.

Collision Resistance

Si analizzano le **collisioni** considerando solo **i primi *n* bit** del digest.
Al crescere di n, lo spazio di output aumenta, e la frequenza di collisioni diminuisce rapidamente.
Per n > 32, il numero di collisioni osservate cala drasticamente, e tende a zero quando n ≥ 43. La quantità di collisioni per Poseidon e SHA-256 è molto simile.





Uniformità & Chi-square

Una buona funzione hash deve produrre output con **distribuzione uniforme**, simile a sequenze casuali.

Anche se deterministica, ci si aspetta che:

- La proporzione di bit 0 e 1 sia **bilanciata** (~50% ciascuno).
- Tutti i possibili byte (0-255) appaiano con **frequenze simili**.

Il test **Chi-square** confronta le frequenze osservate con quelle attese e ne se ricava il p-value, che indica la probabilità che le differenze siano significative o casuali.

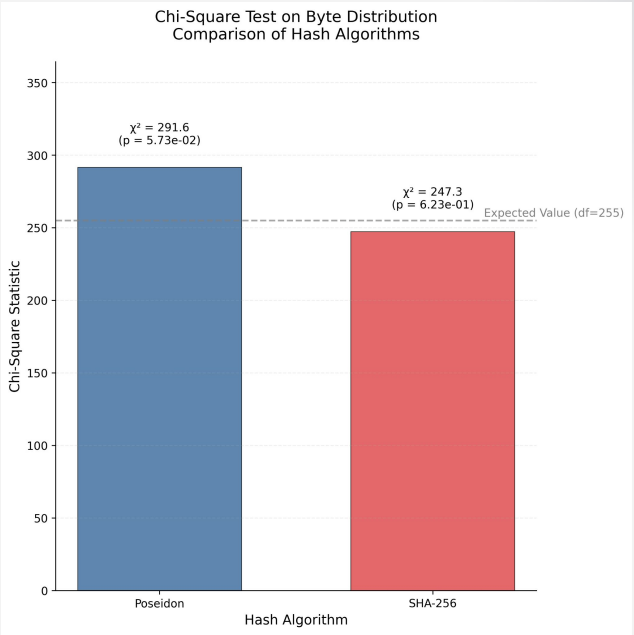
Uniformità & Chi-square

Sia SHA256 che Poseidon hanno un rapporto tra bit 0 e 1 **bilanciato.**

La frequenza media di ogni byte è circa **0.39%**, cioè 1/256.

Test Chi-square (χ^2):
- SHA-256: χ^2 = **247.3**, p-value = **0.6234**
- Poseidon: χ^2 = **291.6**, p-value = **0.0573**

Valori vicini all'ideale di 256



Notare i p-value diversi, questa differenza è dovuta al diverso design degli algoritmi. Tuttavia, entrambi sono sopra lo **0.05** confermando Poseidon come funzione hash affidabile.

Risultati Hash Pattern e Correlazione

Si verifica la presenza di sequenze ripetute all'interno dei digest, contando eventuali collisioni. Se ne calcola anche l'Autocorrelazione (R), cioè la correlazione tra i bit di un hash e sé stesso traslato di un certo lag k (ritardo).

Finestre analizzate:
- 32 bit: tasso **praticamente nullo** di collisioni.
- 24bit: **0.0007**%.
- 16bit: **0.18**%.

Per quanto riguarda l'Autocorrelazione algoritmi hanno mostrato valori di autocorrelazione per lag > 0 vicini a 0.25 (valore ideale), con una media di **0.2531**.

Poseidon mostra un comportamento statistico comparabile a SHA-256, senza evidenza di pattern ripetuti o correlazioni sfruttabili.

Conclusioni e Sviluppi Futuri

Poseidon ha dimostrato performance crittografiche **comparabili** a SHA-256 nei vari test statistici.

Confermandosi come robusto e privo di debolezze strutturali, risultando idoneo non solo per ambienti *ZKP*, ma potenzialmente anche per applicazioni più generali.

I risultati ottenuti aprono quindi la strada a futuri approfondimenti, come ad esempio quelli legati ad analisi crittoanalitiche avanzate.

Grazie per l’attenzione!