OpenTimestamps

Timestamping proof standard basato su Bitcoin

Valerio Vaccaro

Satoshi Spritz Milano

18 Marzo 2025

Valerio Vaccaro 👀

- 💻 Sviluppatore Bitcoin ed Esperto Hardware
- 🔥 Contributore a progetti Bitcoin open source
- ⚠️ Appassionato di hardware fai-da-te (DIY)
- Ingegnere Bitcoin e Liquid presso Blockstream

Social

- 👨‍💼 **LinkedIn** linkedin.com/in/valeriovaccaro
- 🐙 **Github** github.com/valerio-vaccaro
- **Telegram** t.me/valeriovaccaro

Licenza

May this presentation inspire you to become more self-sovereign!

Meme


IT WAS A VERY NICE TIMESTAMP
IT WAS THE PERFECT TIMESTAMP
makeameme.org

Sommario

- Cos'è OpenTimestamps
- Timestamping classico in Italia e data certa
- Creare, aggiornare e verificare proof
- Esempi d'uso
- Flusso e diagrammi
- Calendari e risorse

Cos'è OpenTimestamps?

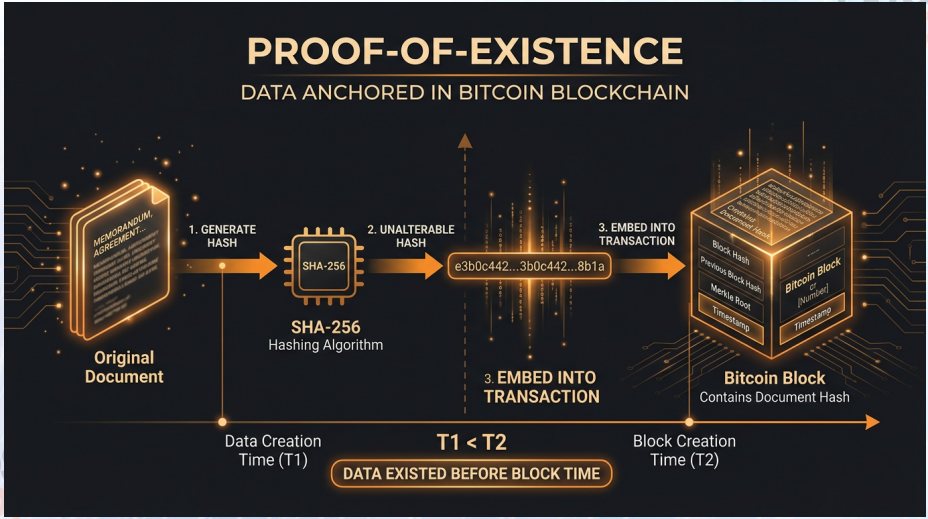
Standard per timestamp su blockchain

- **Proof-of-existence**: prova che dati esistevano prima di un certo tempo
- opentimestamps.org
- Trust-minimized: usa Bitcoin, nessuna autorità centrale
- Scalabile: migliaia di timestamp in una transazione

Vantaggi (Peter Todd, 2016)

- **Trust**: blockchain decentralizzata, auditabile
- **Cost**: aggregazione Merkle, costo trascurabile
- **Convenience**: timestamp in ~1 secondo (calendar server)

Concetto proof-of-existence



Cosa può e non può provare un timestamp

Può provare

- ☑ **Integrità record**: dati esistevano prima di un evento
- ☑ **Firma software/PGP**: data della firma verificabile
- ☑ **Autenticità evidenze**: limita chi può aver alterato

Non può provare

- ● **Proprietà**: non attribuisce direttamente ad una persona sebbene posso mantenere segreta la ricevuta per dimostrare la conoscenza del file ad una certa data
- ● **Unicità**: non risolve doublespend
- ● **Contenuto**: prova esistenza, non significato

Timestamping classico in Italia

Esempi di data certa

- ✉️ **Ufficio postale**: raccomandata con ricevuta di ritorno
- 🚗 **PRA** (Pubblico Registro Automobilistico): registrazione veicoli
- 📜 **Notaio**: atti notarili con data certa
- 📅 **Data certa**: valore legale ex art. 2704 c.c. (registro, bollo, PEC)

Contesto

- 🔑 Tutti richiedono un **terzo fidata** (Poste, PRA, notaio)
- ⚙️ Processi cartacei o burocratici, costi e tempi

Difficoltà: file digitali e firme

Il problema

- ⚠️ **File digitale**: nessun supporto fisico da imbucare o bollare
- ⚠️ **Firma digitale/PGP**: la firma prova autenticità, ma **non quando** è stata apposta
- 🔴 Senza data certa: impossibile dimostrare che un documento esisteva a una data precisa

Perché serve

- 🔗 Contratti, brevetti, prove legali, commit di codice
- 🛡️ La data è spesso cruciale per validità e priorità

Data certa e furto delle chiavi

Scenario critico

- Chiave PGP/software **rubata** → revoca e rilascio nuova chiave
- **Problema**: come distinguere firme valide (pre-furto) da firme false (post-furto)?

Soluzione: timestamp sulla firma

- **Data certa sulla firma** → prova che la firma esisteva **prima** della revoca
- Firma con timestamp pre-revoca = **valida**; firma senza = sospetta
- Fondamentale per riconoscere firme legittime in caso di furto chiavi

Creare un timestamp (stamp)

Comando

```
$ pip3 install opentimestamps-client
$ ots stamp documento.pdf
```

Output

- Crea `documento.pdf.ots` (file proof)
- Invia hash ai calendar server (Alice, Bob, ...)
- Ricevi proof incompleto in ~1 secondo

Con attesa conferma Bitcoin

```
$ ots stamp --wait documento.pdf
```

Aggiornare un timestamp (upgrade)

Proof incompleto vs completo

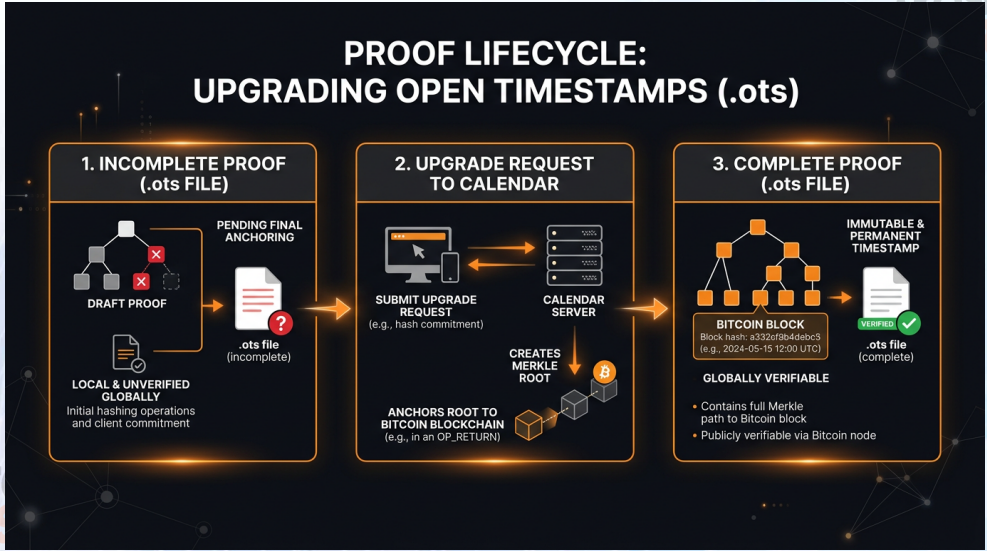
- ⚠️ **Incompleto**: dipende dal calendar server (PendingAttestation)
- ✅ **Completo**: proof Bitcoin nella blockchain, verifica locale

Upgrade

```
$ ots upgrade documento.pdf.ots
```

- 🔗 Scarica attestazione Bitcoin dal calendar
- 🔑 Salva proof completo nel file .ots
- 🛡️ Verifica possibile senza calendar

Ciclo di vita proof: Incompleto a Completo



Verificare un timestamp (verify)

Comando

```
$ ots verify documento.pdf.ots
```

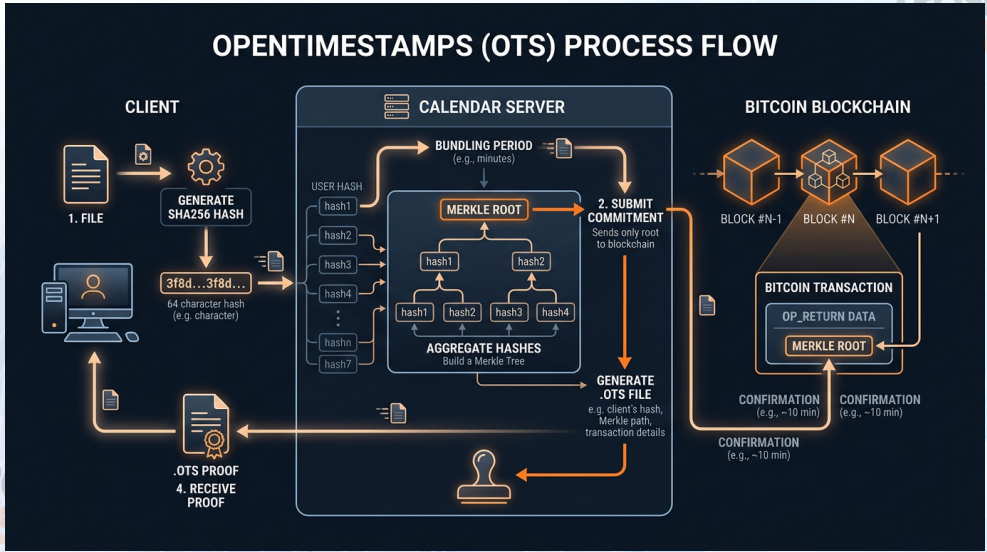
Processo

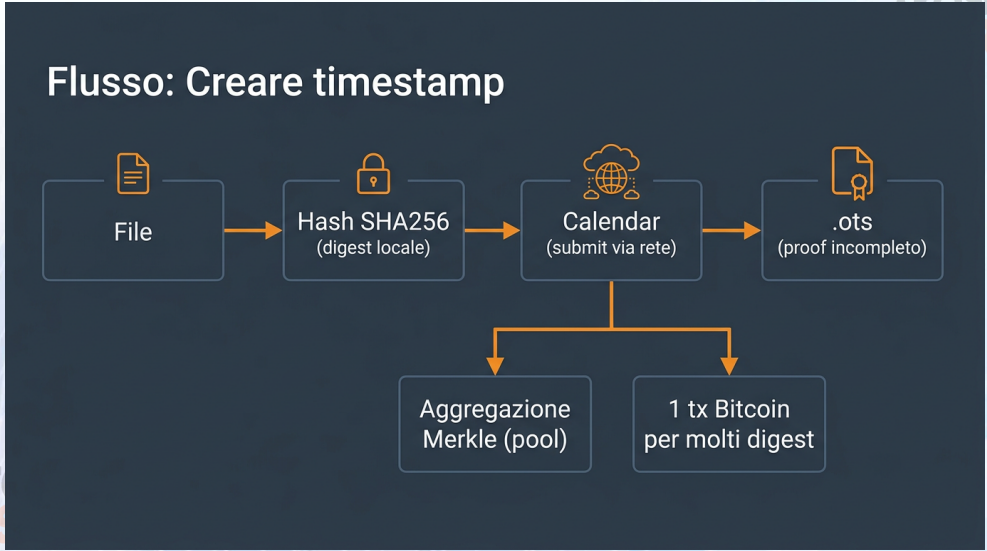
- ⚙ Calcola hash del file originale (se fornito)
- 🔑 Esegue le commitment operations del proof
- 🔗 Se incompleto: chiede al calendar le attestazioni
- ✅ Verifica contro block header Bitcoin

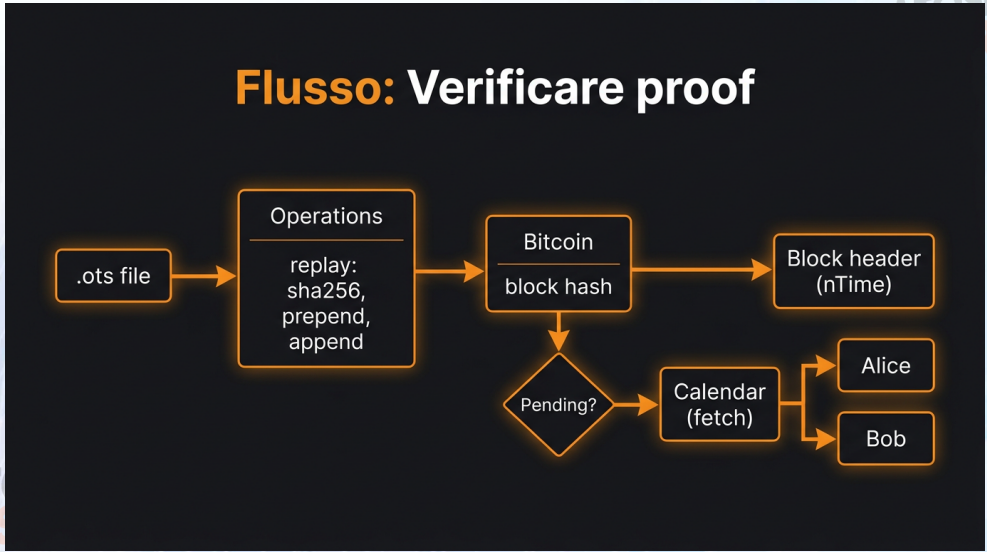
Output

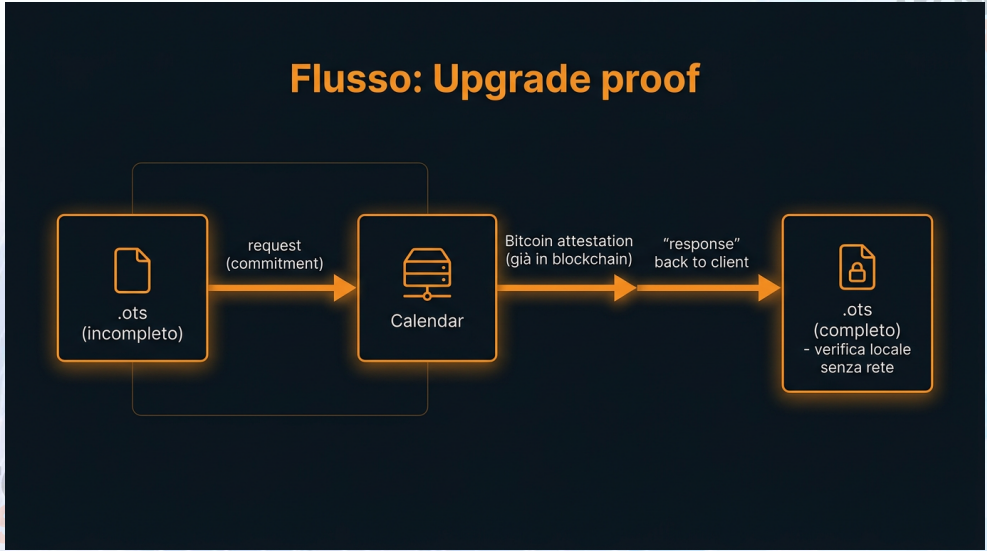
```
Success! Bitcoin attests data existed as of Thu May 28 15:41:18 2015 UTC
```

Diagramma: Flusso OpenTimestamps




Flusso: Creare timestamp
File
Hash SHA256
(digest locale)
Calendar
(submit via rete)
.ots
(proof incompleto)
Aggregazione
Merkle (pool)
1 tx Bitcoin
per molti digest


Flusso: Verificare proof
.ots file
Operations
replay:
sha256,
prepend,
append
Bitcoin
block hash
Block header
(nTime)
Pending?
Calendar
(fetch)
Alice
Bob


Flusso: Upgrade proof
.ots
(incompleto)
request
(commitment)
Calendar
Bitcoin attestation
(già in blockchain)
"response"
back to client
.ots
(completo)
- verifica locale
senza rete

Commitment operations

Albero di operazioni

- **sha256**, **ripemd160**: hash
- **prepend**, **append**: dati fissi
- **verify**: attestazione (Bitcoin, Pending, . . .)

Esempio

```
message -> sha256 -> prepend X -> append Y -> sha256 -> verify BitcoinBlock(35
```

Calendari pubblici

Server gratuiti

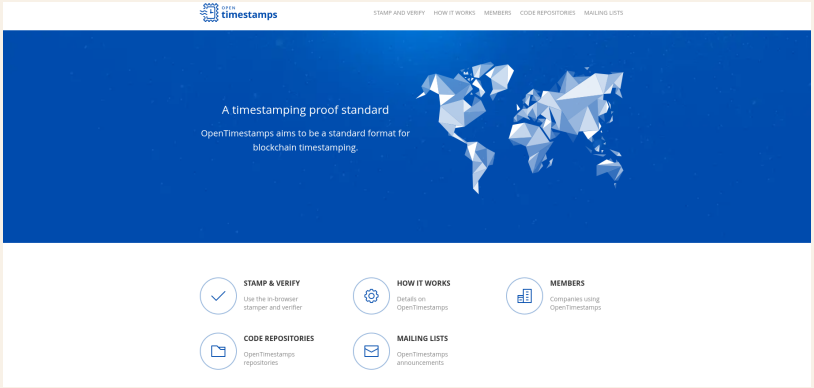
- **Alice**: alice.btc.calendar.opentimestamps.org
- **Bob**: bob.btc.calendar.opentimestamps.org
- **Finney**: finney.calendar.eternitywall.com

Ridondanza

- Di default: 2+ calendari per timestamp
- Calendari opzionali: `ots stamp --wait` (no calendar)

Esempi d'uso

Interfaccia web (opentimestamps.org)



Esempi d'uso

Documenti

```
$ ots stamp contratto.pdf
$ ots verify contratto.pdf.ots
```

Git commit

```
$ git commit -m "Release v1.0"
$ ots stamp .git/COMMIT_EDITMSG
```

PGP / firma software

- Timestamp sulla firma: data revoca chiave vs data firma

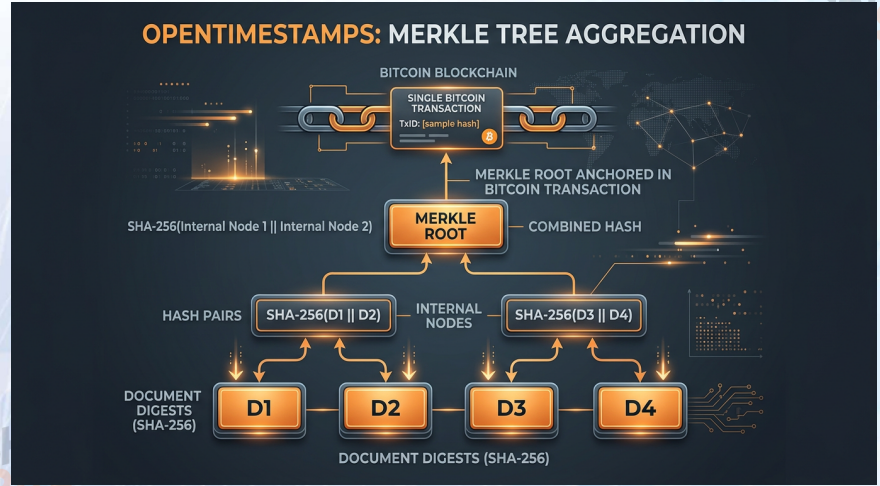
Esempi: codice

```
$ echo "Hello World!" > hello.txt
$ ots stamp hello.txt
$ ots info hello.txt.ots
$ ots verify hello.txt.ots
```

Info

```
$ ots info file.ots
File sha256 hash: 03ba204e50d126e4674c005e04d82e84c21366780af1f43bd54a37816b6a
Timestamp:
  ripemd160
  prepend ...
  verify BitcoinBlockHeaderAttestation(358391)
```

Merkle tree: aggregazione



Integrazione OpenTimestamps in GitHub

Perché timestampare i commit

- **Protezione IP**: i commit Git non sono immutabili — le date si possono modificare
- **Prova temporale**: dimostra che il lavoro esisteva a una data precisa (utile in contesti legali)
- **Record permanente**: timestamp su blockchain = prova indelebile

GitHub Action

- **open-timestamps-github-action** (yzernik/open-timestamps-github-action)
- Timestampa i tag Git sulla blockchain Bitcoin
- Disponibile su GitHub Marketplace, si integra nei workflow

Integrazione GitHub: workflow

Esempio di utilizzo

```yaml
# .github/workflows/timestamp.yml
on:
  push:
    tags: ['*']
jobs:
  timestamp:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: yzernik/open-timestamps-github-action@v1
      - run: git push origin --tags
```

Integrazione GitHub: workflow

Alternativa: manuale

```
$ ots stamp .git/COMMIT_EDITMSG
$ ots stamp .git/refs/heads/main
```

Integrazione GitHub: risorse

Documentazione e link

- git-integration.md — guida ufficiale
- open-timestamps-github-action — workflow automatico
- Altri: **tstamp** (Go), client Python/JS/Rust

Casi d'uso

- Release e tag di versione
- Prove di priorità per brevetti
- Audit trail per commit sensibili

Riferimenti

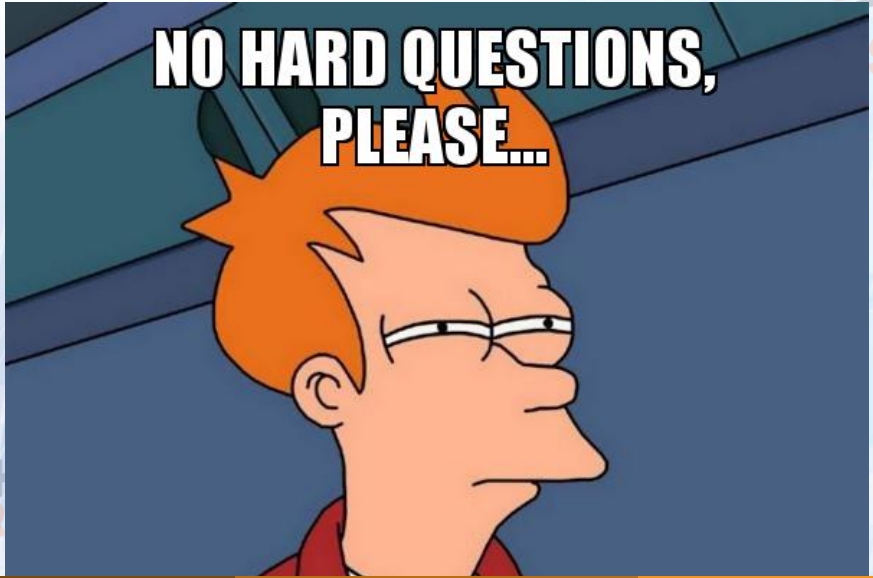
Blockstream / OpenTimestamps

- **opentimestamps.org**: opentimestamps.org
- **Client**: github.com/opentimestamps/opentimestamps-client
- **Announcement**: petertodd.org/2016/opentimestamps-announcement

Implementazioni

- python-opentimestamps
- javascript-opentimestamps
- rust-opentimestamps

? Domande



Satoshi Spritz

- Federazione di gruppi locali di Bitcoiner
- Eventi gratuiti e privacy oriented
- BITCOIN ONLY
- Satoshi Spritz Connect online settimanale

Links

- satoshispritz.it
- t.me/SatoshiSpritzConnect

Officine Bitcoin

- 🤝 Comunità Italiana di Bitcoiners, totalmente gratuita
- 🤖 BITCOIN ONLY
- 🎓 Focus su educazione e sviluppo di progetti

Links

- officinebitcoin.it

Bitcoinissimo

- Podcast Bitcoin e statistiche
- Episodi in italiano, inglese, ungherese, cinese, russo, spagnolo, francese
- Statistiche rete in tempo reale, dati di mercato, block explorer

Links

- bitcoinissimo.it