ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

# Quantum-Safe Blockchain

Evaluating the Feasibility of Introducing Quantum-Safe Digital Signatures For Blockchain Using the Example of a Minimal Python-based Blockchain

Kimika Uehara - 1900124818
Silas Pohl - 1900124387

01 **Introduction**

Motivation and Research Question

02 **Method**

Approach to Answer Our Research Question

03 **Implementation**

Practical Showcase of the Project

04 **Results**

Measurements, Metrics and Graphs

05 **Conclusion**

Discussion, Limitations and Further Research

**“It’s time to prepare for quantum threats.”**

- Dr. Lily Chen (mathematician and NIST fellow)

https://www.theverge.com/22523067/nist-challenge-quantum-safe-cryptography-computer-lattice

# How feasible is the integration of quantum-safe signature algorithms into blockchains?

**Select quantum-safe algorithms to evaluate**

Implement Python blockchain (classic and quantum-safe)

Conduct comparison by measuring perf. / attributes

🟢

**Hash-based Cryptography**
rely on secure cryptographic hash functions, which exhibit properties like being difficult to reverse, resistant to finding original inputs,
and robust against collision attacks

🔵

**Lattice-based Cryptography**
sets of points arranged periodically in multi-dimensional spaces. Lattice-based systems are founded on the shortest vector problem (finding the smallest non-zero point within a lattice), which is NP-hard

🟠

**Multivariante Cryptography**
rely on the complexity of multivariate system of equations, which have been demonstrated to be NP-complete or NP-hard

First group of winners from NIST’s six-year competition

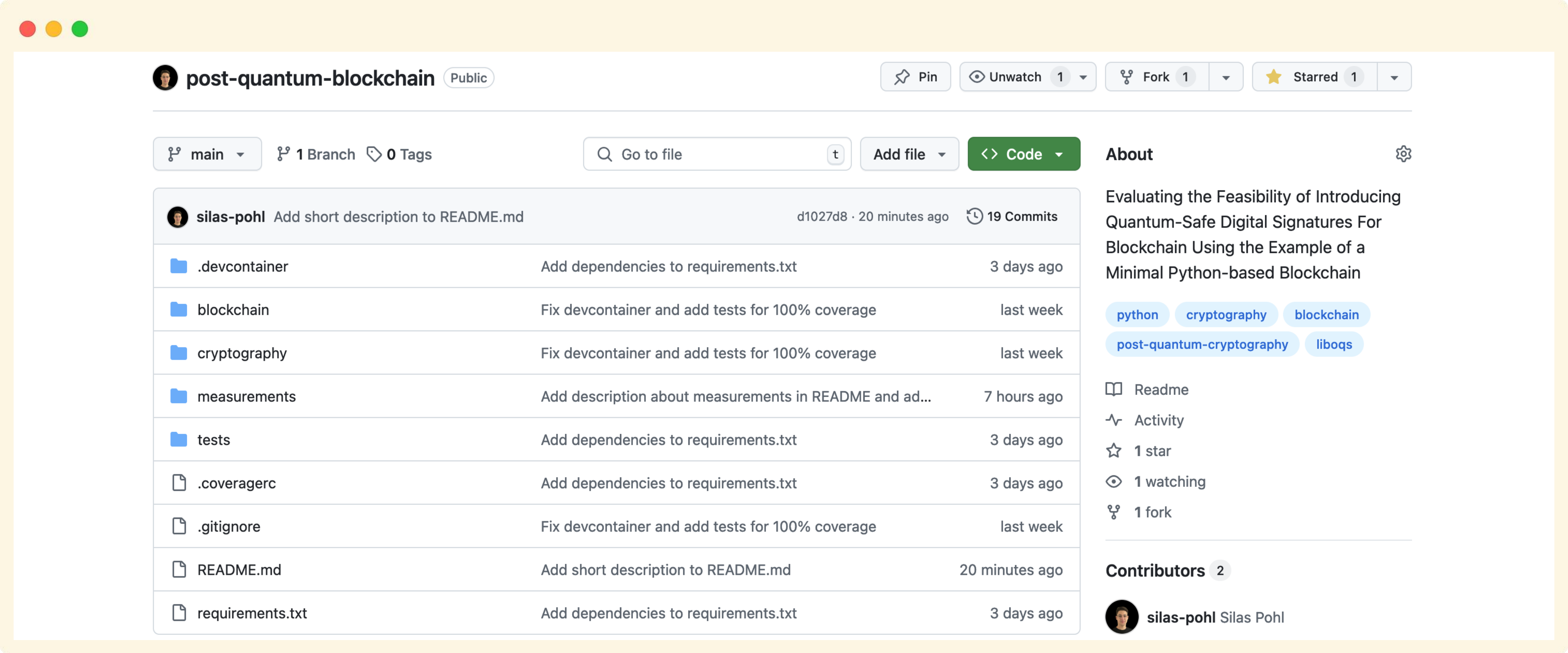
# 🔵 CYSTALS-Dilithium, 🔵 FALCON, 🟢 SPHINCS+

https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms

Select quantum-safe algorithms to evaluate

**Implement Python blockchain (classic and quantum-safe)**

Conduct comparison by measuring perf. / attributes



silas-pohl Add short description to README.md d1027d8 · 20 minutes ago 19 Commits

| | | |
|---|---|---|
| .devcontainer | Add dependencies to requirements.txt | 3 days ago |
| blockchain | Fix devcontainer and add tests for 100% coverage | last week |
| cryptography | Fix devcontainer and add tests for 100% coverage | last week |
| measurements | Add description about measurements in README and ad... | 7 hours ago |
| tests | Add dependencies to requirements.txt | 3 days ago |
| .coveragerc | Add dependencies to requirements.txt | 3 days ago |
| .gitignore | Fix devcontainer and add tests for 100% coverage | last week |
| README.md | Add short description to README.md | 20 minutes ago |
| requirements.txt | Add dependencies to requirements.txt | 3 days ago |

About

Evaluating the Feasibility of Introducing Quantum-Safe Digital Signatures For Blockchain Using the Example of a Minimal Python-based Blockchain

python cryptography blockchain post-quantum-cryptography liboqs

Readme
Activity
1 star
1 watching
1 fork

Contributors 2

silas-pohl Silas Pohl

https://github.com/silas-pohl/post-quantum-blockchain

Select quantum-safe algorithms to evaluate

Implement Python blockchain (classic and quantum-safe)

**Conduct comparison by measuring performance**

**Public & Secret Key Sizes**

**Signature Size**

**Blockchain Storage**

**Transaction Time**

**Verification Time**

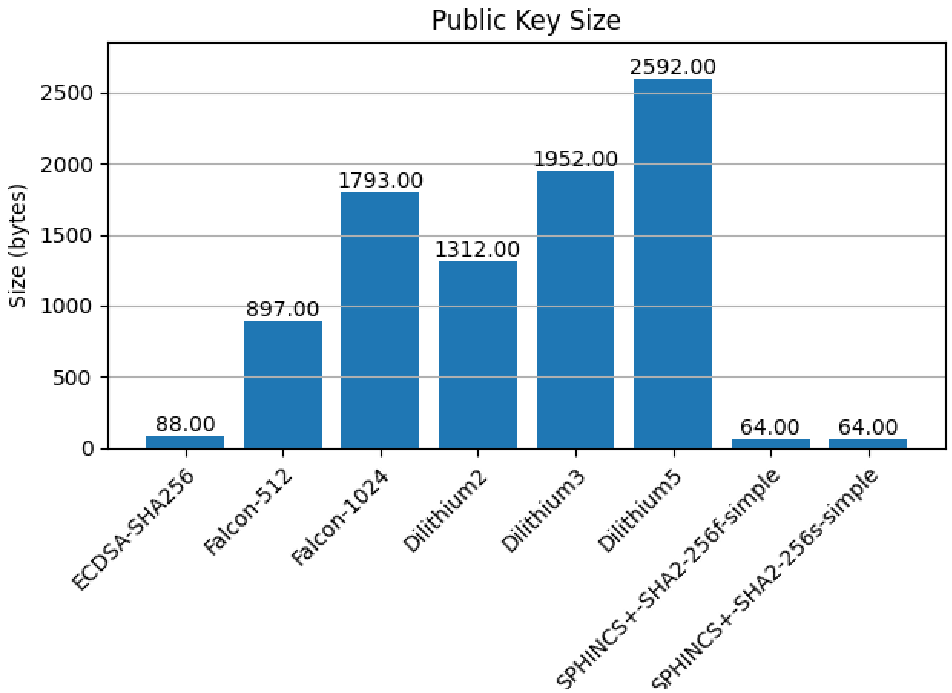
**Mining Time**

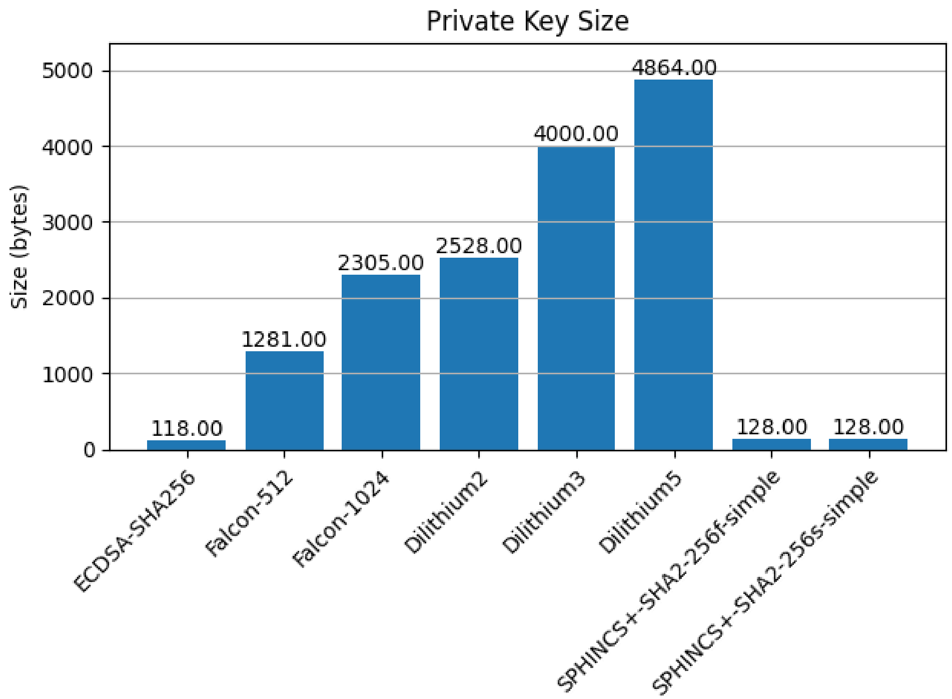
# SHOWCASE

# RESULTS

# Public & Private Key Size

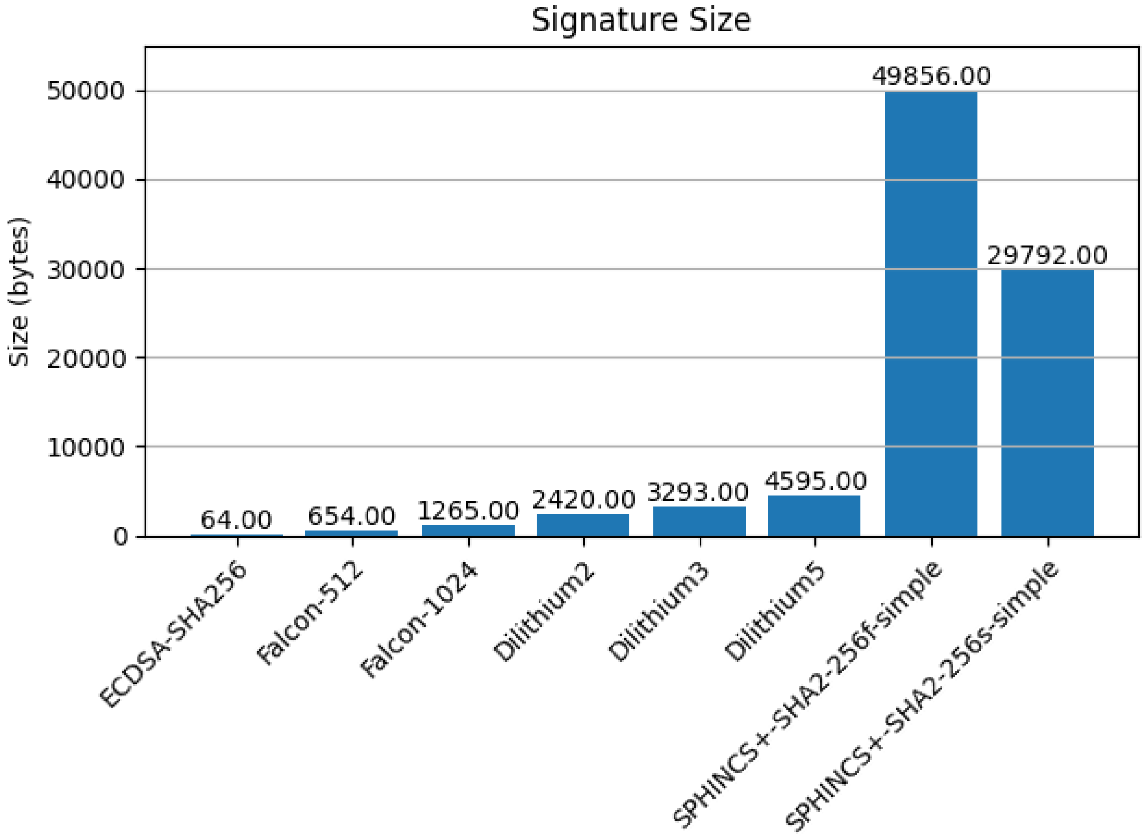
- ▶ Larger key sizes are designed to resist attacks, but result in larger transaction data size
- ▶ Smaller key sizes help improve performance in high-throughput environments





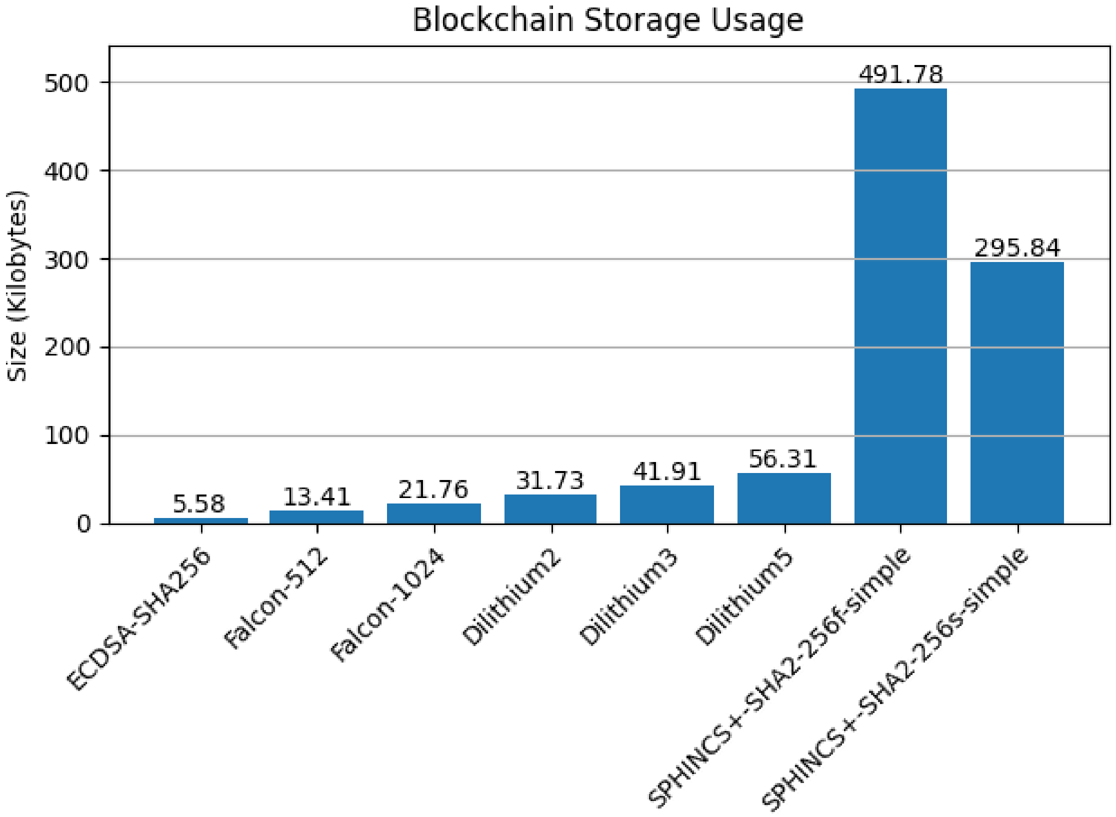
# Signature Size

- Signature size increases blockchain’s **storage requirements**
- Effects on transaction throughput



# Blockchain Storage

- After 10 transactions
- Bloated blockchain sizes due to key/signature sizes for PQC
- Large storage requirements may lead to issues with **scalability** and **nodes' ability to store and synchronize** blockchain

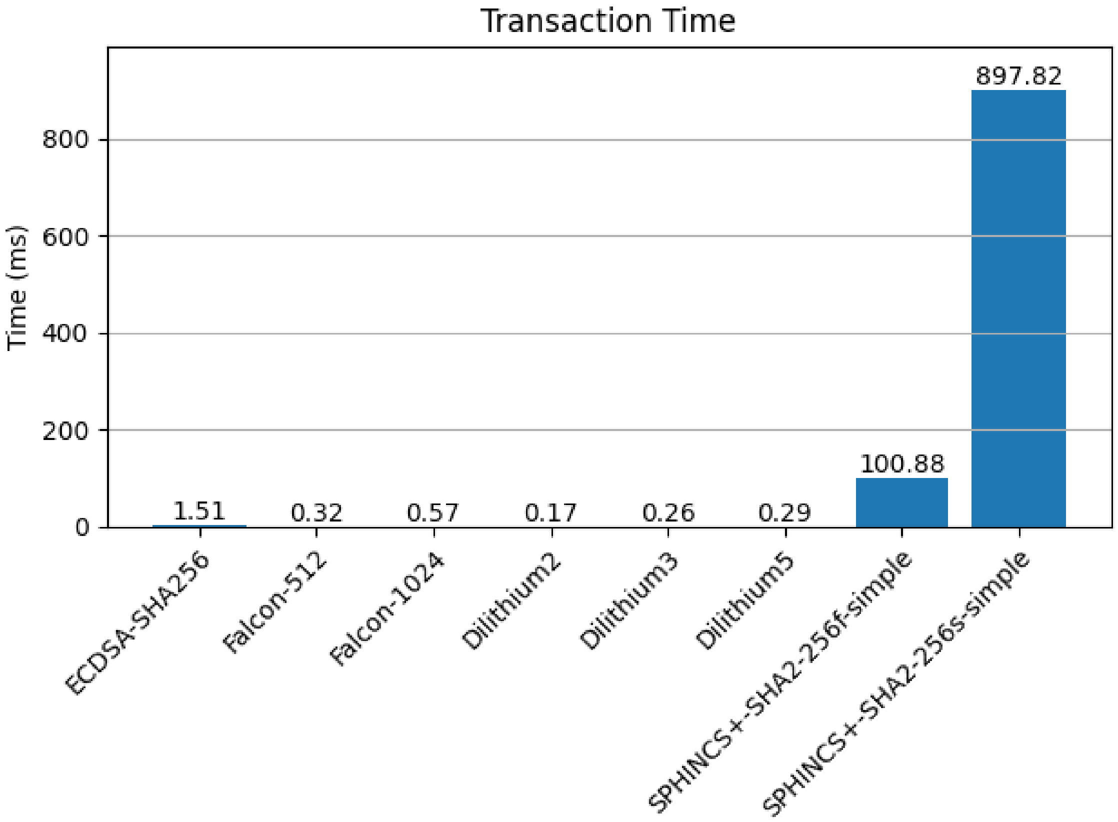


# Transaction Time

- How long it takes to create and sign a transaction

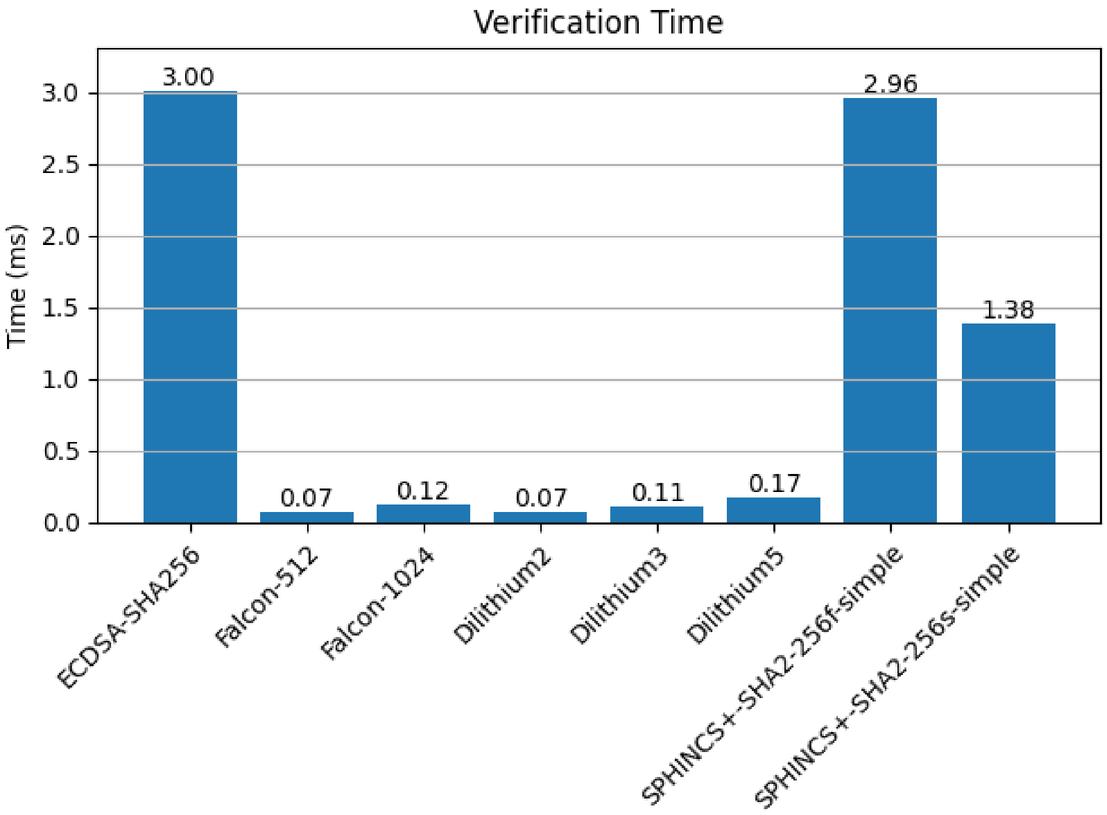
- Longer transaction times could lead to slower confirmations and lower transaction throughput

- For blockchain systems that prioritize **speed and cost-effectiveness** (e.g., microtransactions, DeFi platforms), small transaction times are critical



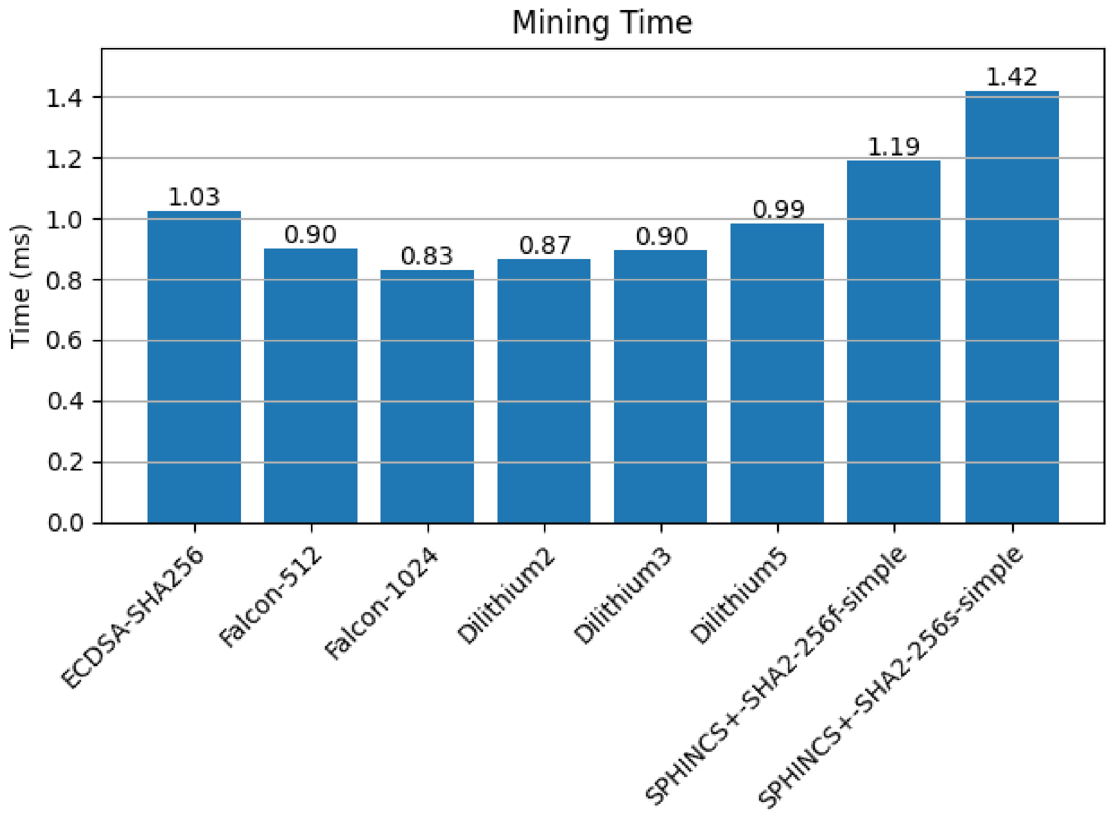
# Verification Time

- How long it takes to verify a transaction
- Verification time affects **transaction throughput, block propagation and scalability**
- Faster transaction confirmation times enhances **usability**



# Mining Time

- How long it takes to mine a transaction and add a new block
- Hashing algorithm, block size and difficulty are kept constant
- Possible variation due to  signature size



# Discussion

| | Falcon | CRYSTALS-Dilithium | SPHINCS+ |
|---|---|---|---|
| + | • Compact signature sizes<br>• Signing and verification efficiency | • Signing and verification efficiency<br>• Moderate signature sizes | • Small public and private key sizes<br>• Comparable verification time to ECDSA |
| - | • Large public and private key sizes | • Large public and private key sizes<br>• Larger signature and storage than Falcon | • Large signature sizes<br>• Long transaction time<br>• Large storage use |

# Discussion

### Limitations

- Results are affected by Python implementation
- Lack of networking capabilities
- Small scale experiment

### Future Directions

- Compare other quantum-safe algorithms
- Compare hashing algorithms
- Implement networking capabilities
- Experiment with block sizes and blockchain difficulties

# Conclusion

- Post-quantum signatures face trade-off: **security vs key/signature size + storage**

- **Lattice-based algorithms** (Falcon, CRYSTALS-Dilithium) display great **signing and verification** efficiency

- SPHINCS+ (**hash-based**) suffers from **inefficiency + storage** despite small key size

- Future post-quantum integration is dependent on system requirements, long-term scalability, and evolution of computational resources

- Public and permissionless blockchains (e.g. Bitcoin, Ethereum) face greater challenges due to their open and decentralized nature