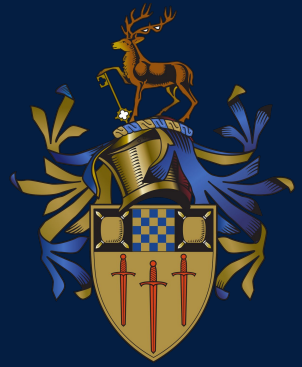


Crypto-agility for Blockchain Protocols

The difference compared to existing crypto-agility concepts, transition mechanisms, and issues specific to blockchain protocols

Shin'ichiro Matsuo (Georgetown University/Virginia Tech), Kigen Fukuda (Virginia Tech), Tadahiko Ito (Secom), Masahi Sato (Secom), Long Meng (University of Surrey), Liqun Chen (University of Surrey)

Presented at NIST Crypto Agility Workshop , Apr 18 2025

Why Blockchain?

Blockchain and Crypto-agility

- Crypto-agility is particularly important for protocols that need longer-term security.
- Blockchain and the resulting crypto-assets are examples of use cases that require long-term security. Since **Blockchain is designed to be persistent and immutable**, the crypto-agility of the underlying algorithms becomes critically important.
- Without proper transition mechanisms, today's crypto assets could lose their value within the next few decades.

What are the underlying cryptographic algorithms of blockchain?

- **Hash Function**:
  PoW, Computing/Verifing Transaction hash, Address generation… e.g. SHA-256, Keccak-256
- **Digital Signature Algorithm**:
  Transaction Signing… e.g. ECDSA, BLS Signature. **Vulnerable to quantum attacks.**

What happens when these primitives are compromised?

Breakage	Effect
Selective forgery	Steal coins from public key
Integrity break	Claim payment not received
Repudiation	-

Table 2: Effects of a break in the signature scheme.

	Signature Property		
Hash Property	**Selective forgery**	**Integrity break**	**Repudiation**
Address Hash (H_A)			
Collision	Repudiate transaction	-	Change existing payment†
Second pre-image	Steal all coins	-	Change existing payment
Pre-image	Steal all coins	-	-
Bounded pre-image	All of the above	-	Change existing payment
Main Hash (H_M)			
Collision	Steal coins	Steal coins†	-
Second pre-image	Steal coins	Double spend†	-
Pre-image	-	-	-
Bounded pre-image	Steal coins	All of the above	-

† Achieving this requires a slight modification of the definitions. See text for details.

Table 3: The effects of a multi-breakage: broken signature scheme in combination with a break in H_A or H_M.

Crypto-agility is essential

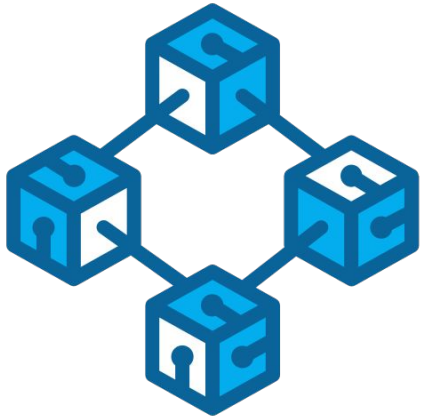
Simply switching cryptographic primitives is not enough. Preserving all past block data validity is also necessary.

[1]Giechaskiel, I., Cremers, C., Rasmussen, K.B. (2016). On Bitcoin Security in the Presence of Broken Cryptographic Primitives. In Proc of ESORICS 2016. LNCS, vol 9879. Springer,

Difficulties in realizing Crypto-agility for blockchain

1. Fundamental Difference Against Crypto-agility for other protocols

- You cannot negotiate cryptographic algorithms (Cipher Suite)
- It is unrealistic to negotiate cryptographic algorithms among an unbounded number of nodes running on distributed consensus.



Security protocol should support one or more algorithm or cipher suite identifiers to achieve crypto agility.

Negotiation for each communication between nodes? Batch update might be required.

Difficulties in realizing Crypto-agility for blockchain

2. Difficulty in updating software at distributed nodes

- Software updates at nodes for distributed consensus are difficult because no one knows which nodes are majority honest nodes.
- Since not all nodes upgrade at the same time, a protocol change may introduce forks.
- **Blockchain becomes more fragile to a 51% attack** during the transition process.

- What happened at CVE-2018-17144? : includes both a DoS component and a critical inflation vulnerability

In order to encourage rapid upgrades, the decision was made to immediately patch and disclose the less serious DoS vulnerability, **concurrently with reaching out to miners, businesses, and other affected systems** while delaying publication of the full issue to give times for systems to upgrade. [2]

[2] https://bitcoincore.org/en/2018/09/20/notice/

Difficulties in realizing Crypto-agility for blockchain

3. Business difficulties

- Blockchain nodes and miners don't have an incentive to adopt crypto-agility strategies.
- The majority of them seek **short-term incentives**. They might not want to discuss PQC migration because such discussions may decrease the price of crypto-assets.
- Detailed cost/benefit analysis and justification are needed.

BGIN's Takeaway[3]:

As a result of multi-stakeholder discussions, the participants concluded that

- they should design and implement crypto-agility mechanisms, including incentive mechanisms.

[3] https://drive.google.com/drive/folders/1iV8GR0mgGNN-rbT_uXmz-hqCjT4eZGAw

Any Studies?

From academia perspective, the Blockchain transition schemes are already proposed, yet these proposal remains on protocol level. Governance issues are not covered.

- **SM17[4]**
  Sato and Matsuo proposed a protocol to extend the validity of blockchain data for past transactions without a hard fork. This protocol applies the long-term signature scheme standardized by ETSI.

- **MC22[5]**
  The Meng and Chen work further develops the scheme proposed by Sato and Matsuo.

  - Support multiple transitions
  - Present a formal security model for the transition scheme.
  - Ensure compatibility of block hash calculation with existing blockchains.

[4] M. Sato and S. Matsuo, "Long-Term Public Blockchain: Resilience against Compromise of Underlying Cryptography," 2017 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 2017, pp. 1-8

[5] Meng, L., Chen, L. (2022). A Blockchain-Based Long-Term Time-Stamping Scheme. In Proc. of ESORICS 2022. LNCS, vol 13554. Springer

Any Studies?

MC22[5]

Generation and Renewal. In the hash renewal phase, the blockchain system recalculates the hash values of previously generated blocks by grouping them. In the digital signature renewal phase, users sign a transaction to transfer their coins to a new address derived from the new signature scheme.

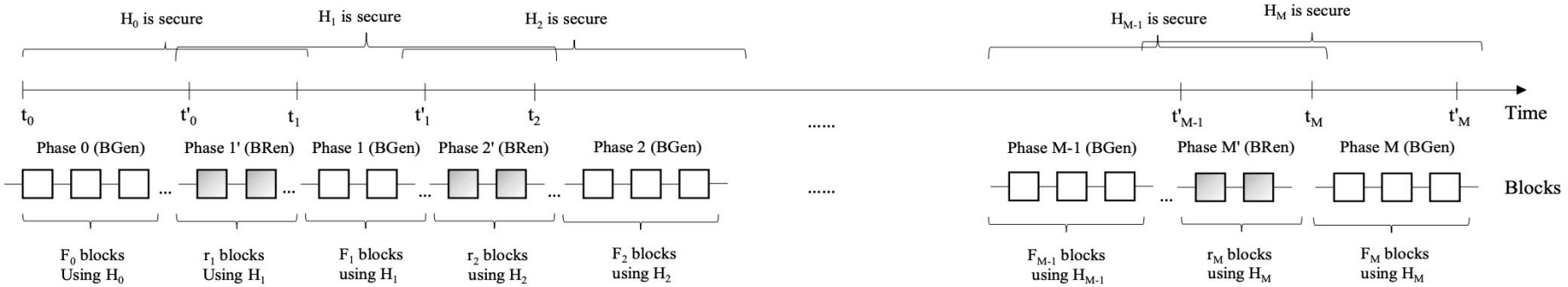


Figure 3: The timeline of hash transition scheme

[5] Meng, L., Chen, L. (2022). A Blockchain-Based Long-Term Time-Stamping Scheme. In Proc. of ESORICS 2022. LNCS, vol 13554. Springer

Blockchain Crypto-agility Requirement

Meng and Chen also defined three requirements of the scheme: correctness, long-term integrity, and long-term unforgeability. [5]

The definitions of each requirement are as follows:

- **Correctness:** if the block and signed transactions are generated following the scheme, they always pass the hash and signature verification, maintaining the blockchain validity.

- **Long-term integrity:** any polynomial time adversary is unable to compromise the scheme's integrity regardless of the lifetime of the underlying cryptographic algorithm. No one can claim non-existed data or tamper data in any of blocks.

- **Long-term unforgeability:** any polynomial time adversary is unable to forge signatures regardless of the lifetimes of the underlying cryptographic algorithm. No one can forge a signature only from the knowledge of public key.

Are there the other requirements?

[5] Meng, L., Chen, L. (2022). A Blockchain-Based Long-Term Time-Stamping Scheme. In Proc. of ESORICS 2022. LNCS, vol 13554. Springer

Required Steps

What is needed?

BGIN's Takeaway[3]

- Develop PQC Blockchain requirements
  - Which primitives should be quantum secure?

- Design and Implement crypto-agility (transition) mechanisms as the first step of the PQC migration.
  - Not all client will update it even quantum threat becomes reality. Set any obligations?
  - Client updates (Digital Signature Algorithm transition) are costly. Any strategies?
  - How to handle the business / technical / governance difficulties?

Blockchain Transition Difficulties and Importance

- Data on-chain requires **extremely** long-term validity.
- The economic impact of broken primitives is **extremely** large.
- Migration is **extremely** difficult and will differ from existing transition schemes.
- Given that the SHA-1 migration took over 20 years, **further study is required without delay.**

[3] https://drive.google.com/drive/folders/1iV8GR0mgGNN-rbT_uXmz-hqCjT4eZGAw

Thank you for listening