THALES

Consensus protocols for Blockchain

Ambre Toulemonde - Université de Versailles Saint-Quentin-en-Yvelines and Thales DIS

EJCIM 2020 – June 2020

Table of contents

THALES

Consensus protocols

The Byzantine Generals problem, 1982

- How reaching an agreement on a value in a distributed manner ?

Basic security properties of consensus protocols

- Satisfying Agreement, Validity and Termination conditions
- Security properties: Safety and Liveness

Main practical consensus protocol before blockchain

- Practical Byzantine Fault Tolerance (PBFT)

Nakamoto blockchain 2008

- New needs for consensus protocols: proof-of-work

THALES

Consensus protocol for blockchain

Blockchain

- A distributed ledger or chain of blocks designed to be immutable
- New blocks are appended into the ledger in distributed manner after reaching a consensus

Bitcoin Proof-of-Work consensus protocol

- Being the first who solves the hash puzzle to be rewarded
- Issues : energy waste problem, centralization to big pool, fork problem and selfish strategy

Many new consensus protocols proposed in the literature

- Avoid the issues of the Bitcoin Proof-of-Work consensus protocol
- Examples of security properties: unpredictability, uniqueness, fairness

THALES

Algorand consensus protocol

Jing Chen and Silvio Micali, 2016

First consensus protocol based on a Verifiable Random Function (VRF)

VRF

- Micali, Rabin and Vadhan, 1999
- VRF provides a random value R and a proof π that R is correctly computed

VRF Properties

- Uniqueness : the output R of the VRF is unique
- Provability : R is the output of VRF and π allows to be convinced of this
- Pseudorandomness : R is not indistinguishable from a truly random

THALES

Algorand consensus protocol

Security properties

- Secret election to prevent DoS attack against the leader: the leader identity remains hidden until she chooses to reveal her leader eligibility proof
- Unique leader eligibility proof computed only by the leader with VRF and her secret

Known attacks

- Fork, bribe attack, DDoS against node

Ongoing work

- Selfish strategy

THALES

Algorand consensus protocol

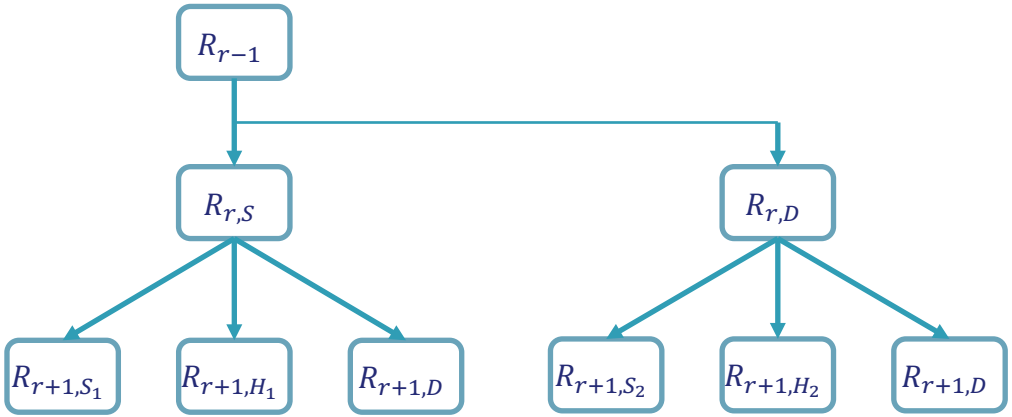
R_0 first random value agreed by the initial nodes

1. Node i is a potential leader for the election r if $HASH(SIGN[sk_i](R_{r-1})) \leq target_r$
2. Leader is the lowest $HASH(SIGN[sk_i](R_{r-1}))$ and $R_r = HASH(SIGN[sk_i](R_{r-1}), r)$
3. If no leader $R_r = HASH(R_{r-1}, r)$

- $target_r$ computed in order to have n potential leaders for each election r

Evaluate selfish strategy on Algorand

- Algorand election with one potential leader
- Goal: keep control as long as possible
- Idea: predict for some rounds the leader eligibility of the selfish pool in order to decide if the selfish leader reveal or not her random value



THALES

Take away

- Consensus protocols allow to reach an agreement on a value in distributed manner
- New requirements for consensus protocols due to blockchain while avoiding Bitcoin PoW issues
- Verifiable Random Function may be a suitable approach to construct a leader election for consensus protocols

THALES