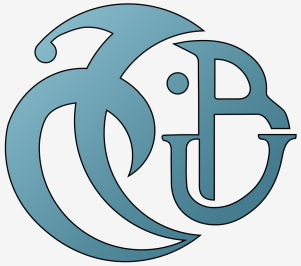


Guaranteeing Information Integrity through Blockchains for Smart Cities

W. Miloud. Dahmane, Dr. S. Ouchani and Pr. Hafida Bouarfa

MEDI 2021 – International Conference on Model and Data Engineering
Tallinn, Estonia
21-23 June 2021.

Contents

1 Introduction

2 Related Work

3 Framework

4 Experiment

5 Conclusion

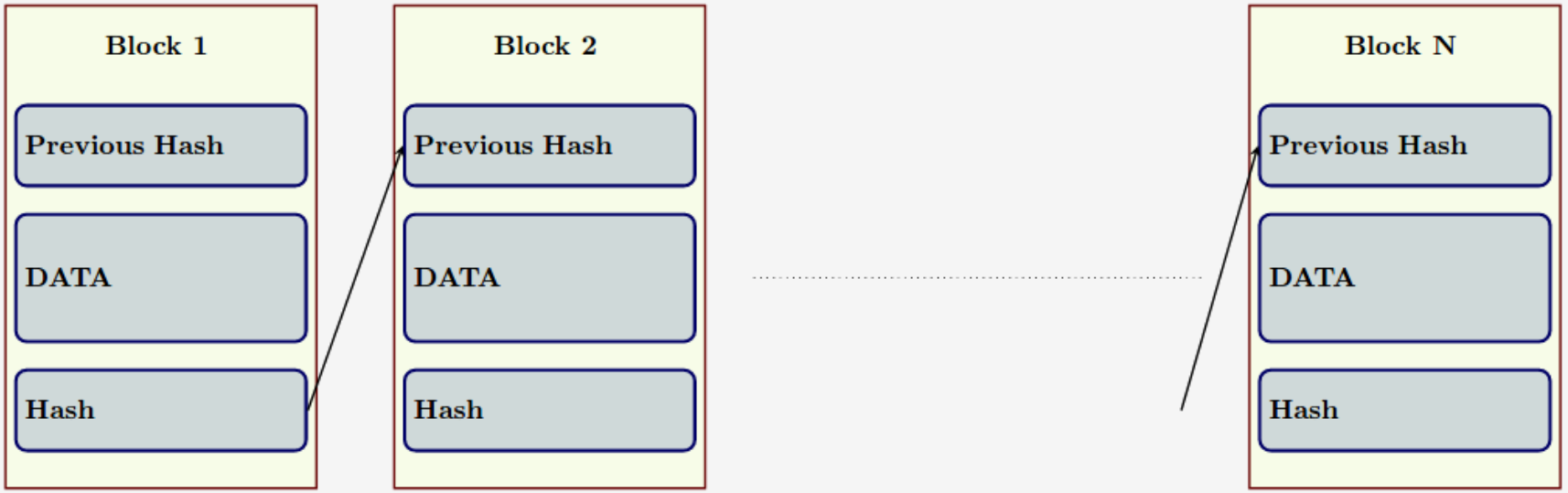
Problematic

Integrity Issue $\longrightarrow$ The world suffers from constant threats

Examples:

- Carry out cyber attacks (In 2006)
- Spread misinformation (2009)
- Penetration the system of iOS (2014)
- Etc,

Blockchain $\longrightarrow$ It is a series of blocks



Challenges

- Adopting this successful technology
- Respecting the characteristics of IoT devices
- Proposing a compatible framework for *Integrity of Information* issue

Goal

Secure Frameworks

- Realising and testing the proposition
- Applying the Blockchain concepts
- Not contradicting reality

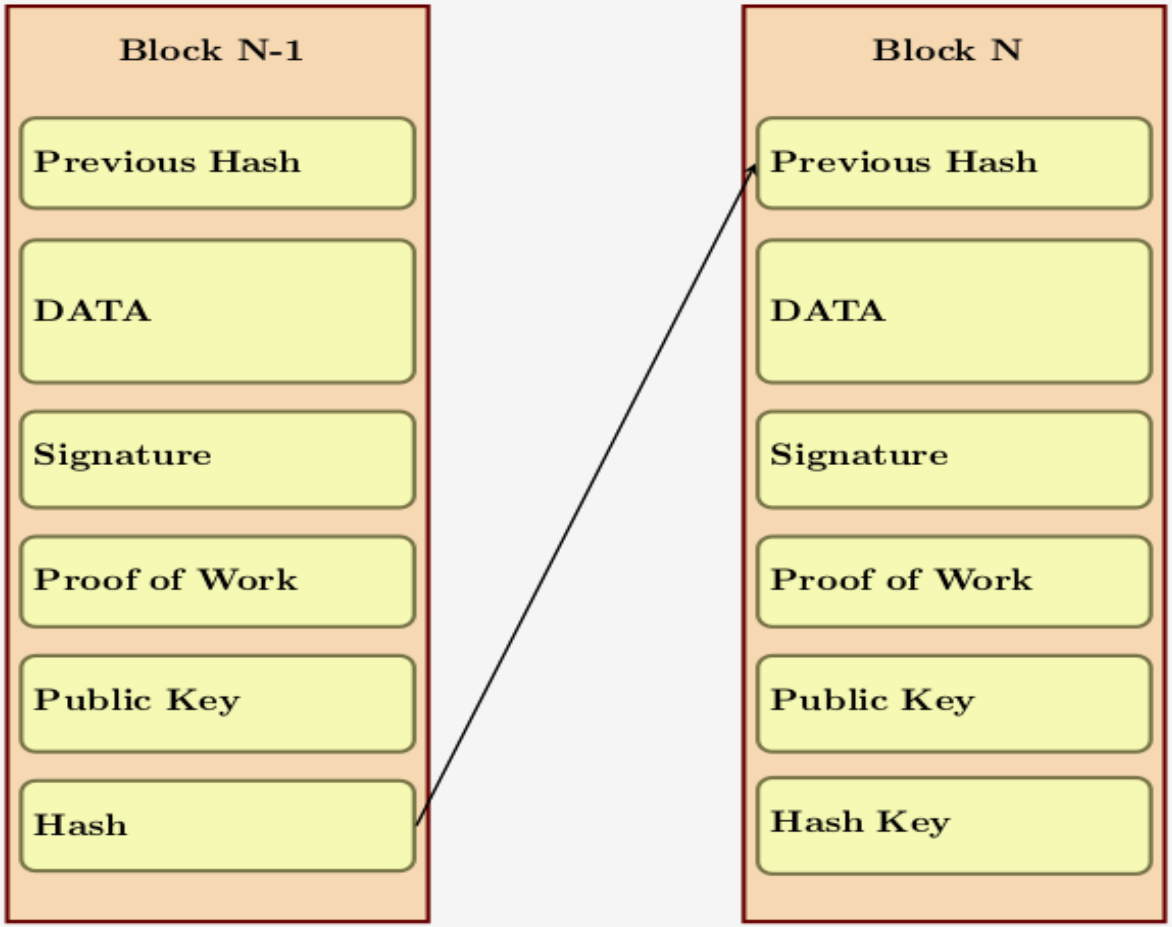
Comparison and perspectives

- Blockchain was used many domains and systems
- Some of them:
  - Did not give examples
  - Hard operations on constrained devices
  - Symmetric cryptography technique

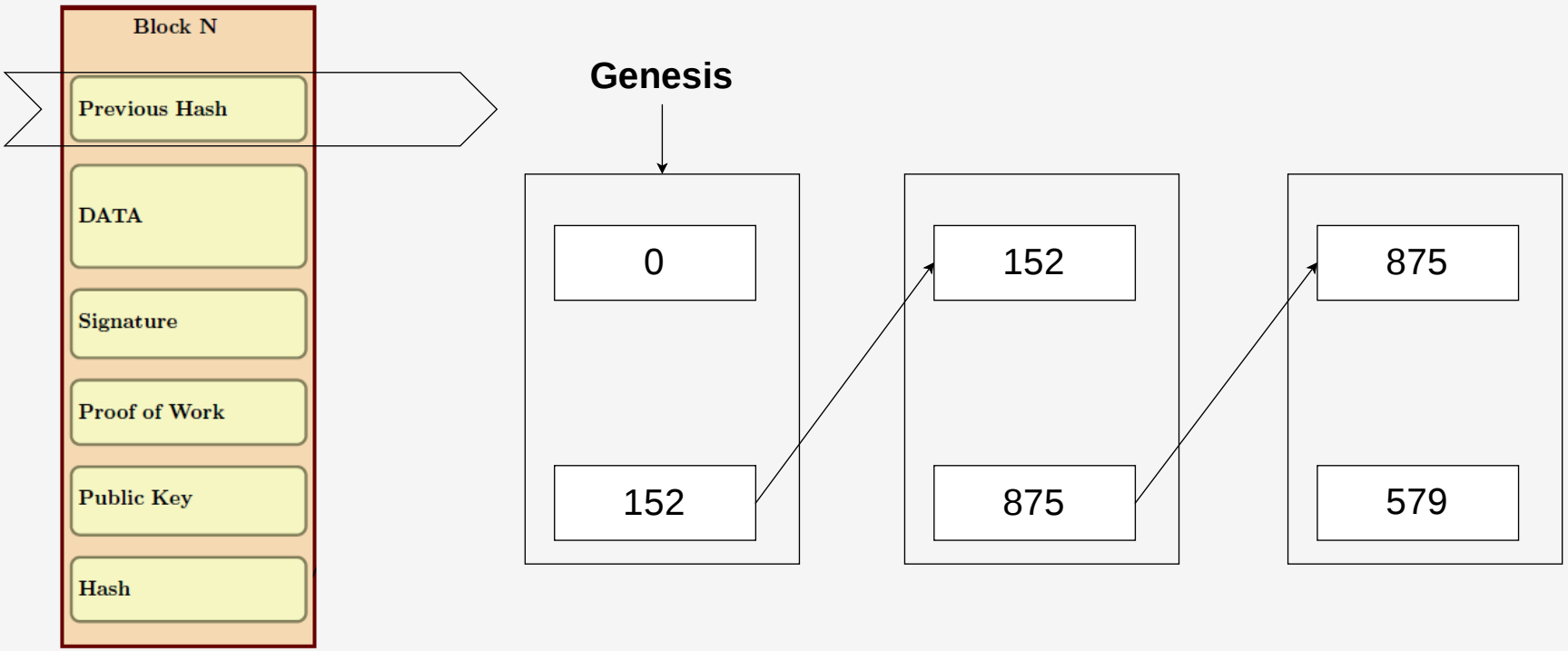
Our contribution

- Focus on the secure protocols
- Unconstrained devices with hard operations
- Asymmetric cryptography technique

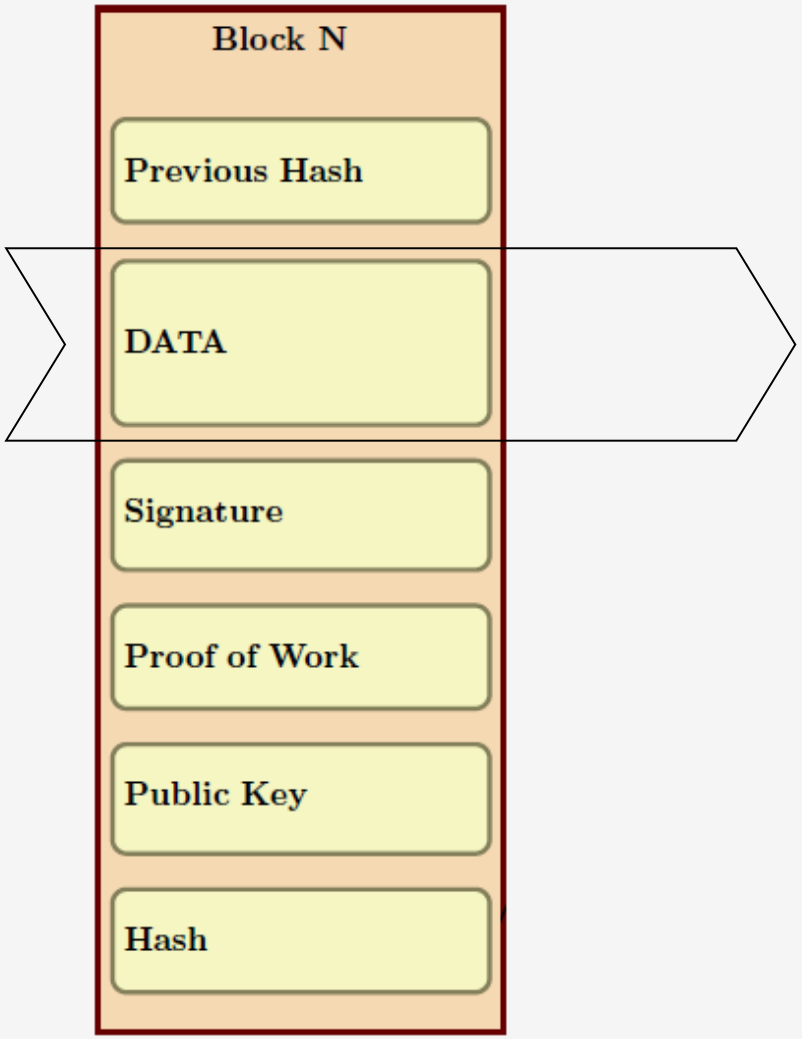
Blockchain Structure



Previous Hash



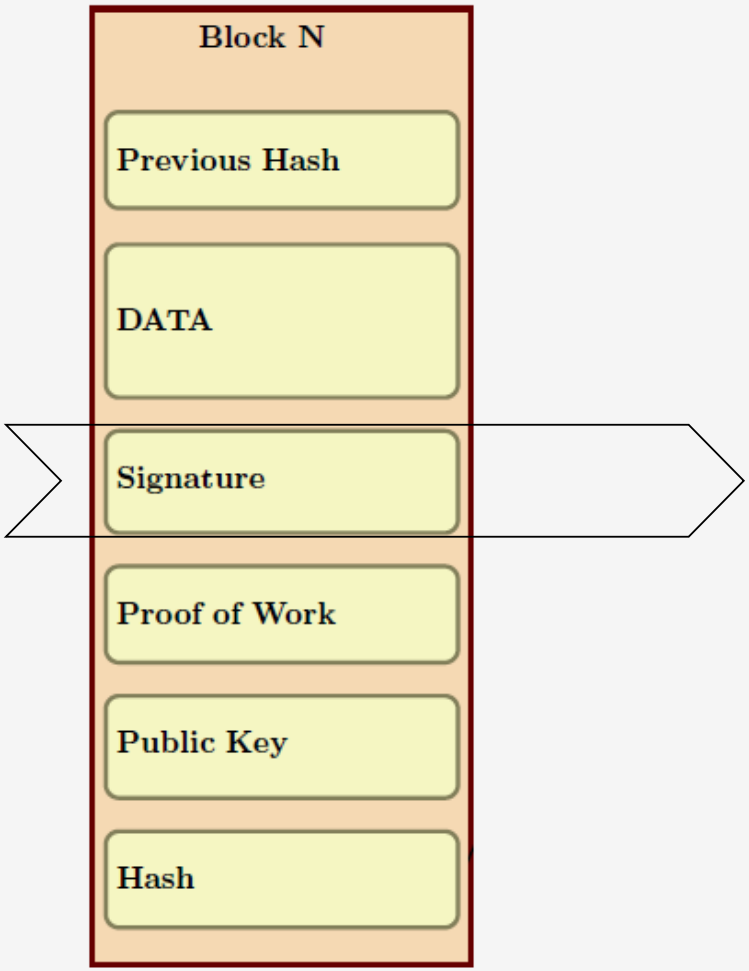
Data

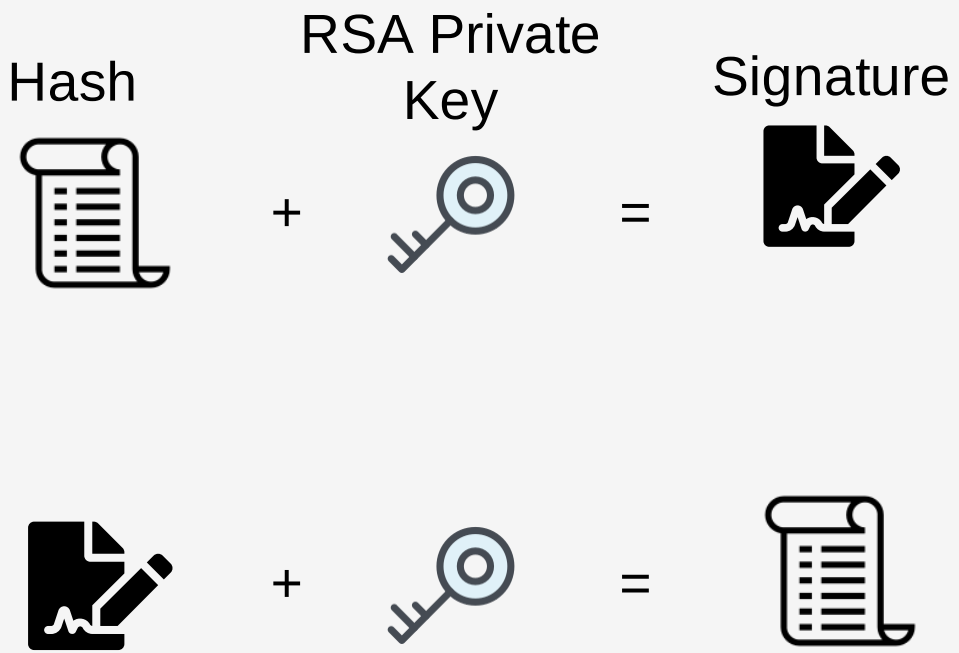


Example:

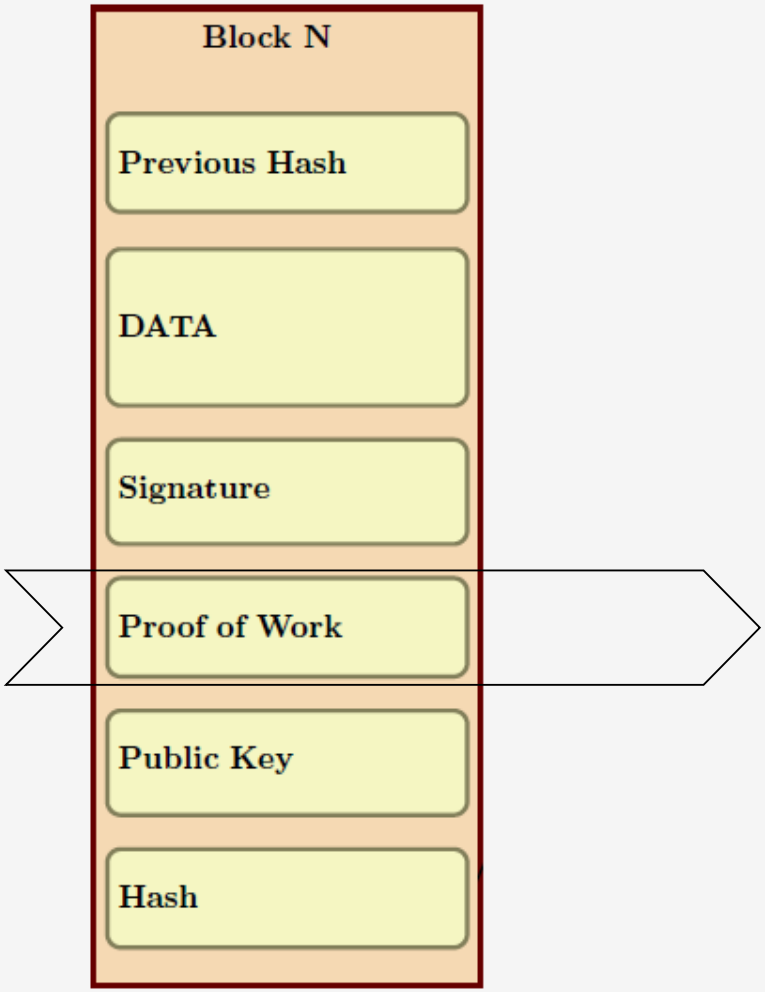
- WSN information
- worker's personal information
- etc

Signature


Block N
Previous Hash
DATA
Signature
Proof of Work
Public Key
Hash


Hash
RSA Private Key
Signature
+
=
+
=

Proof of Work



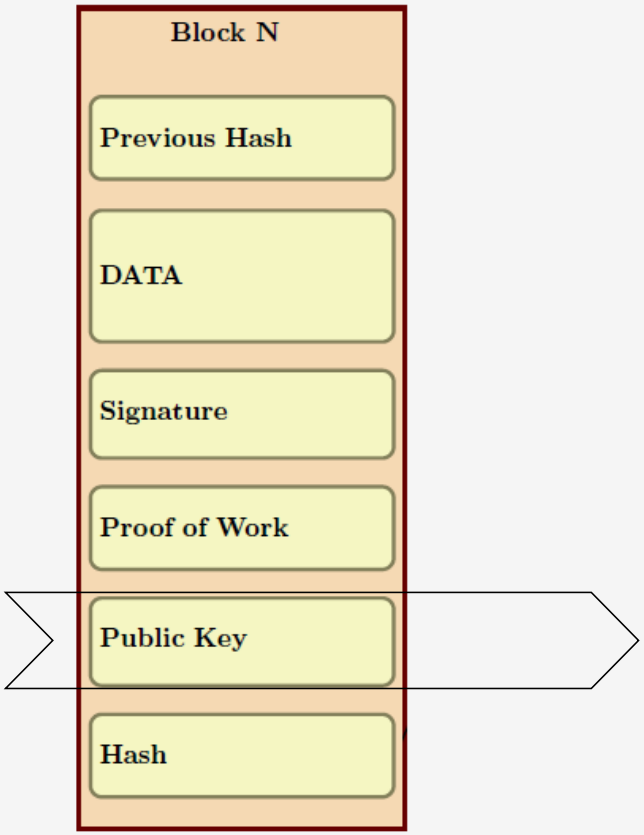
- Defining **Condition**

Do {

PoW + DATA == **Hash**

} *Until* (**Hash** respect **condition**)

Public key

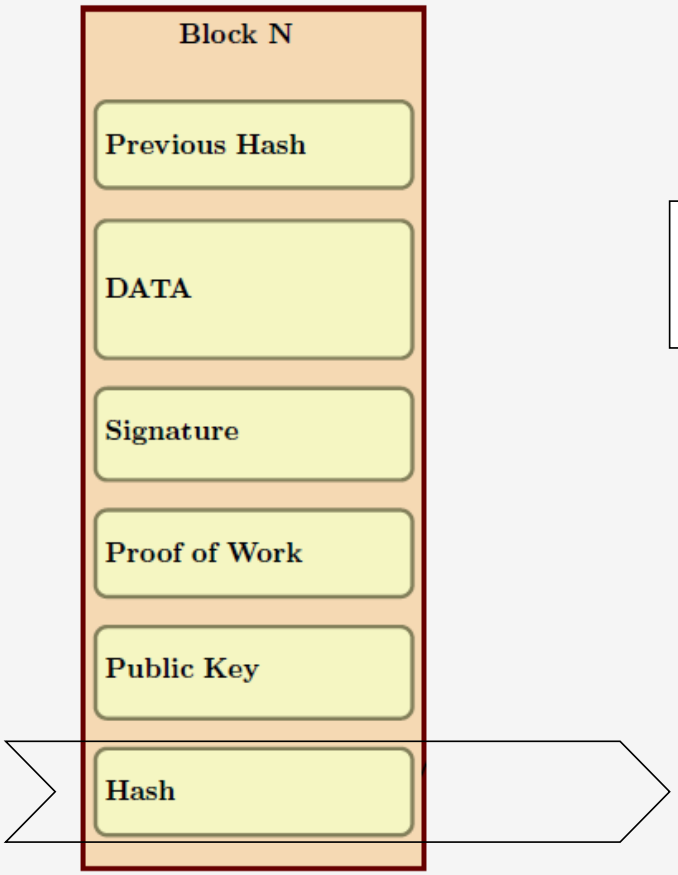


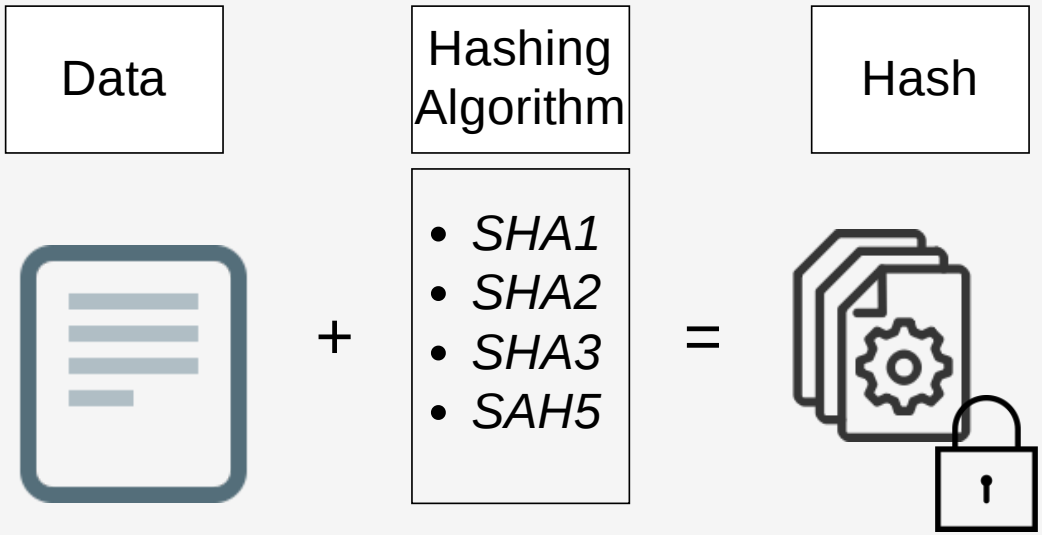
RSA Public Key

2048 bits
<High-strength>

4096 bits
<Very high-strength>

Hash


Block N
Previous Hash
DATA
Signature
Proof of Work
Public Key
Hash


Data
Hashing Algorithm
Hash
+
=
SHA1
SHA2
SHA3
SAH5

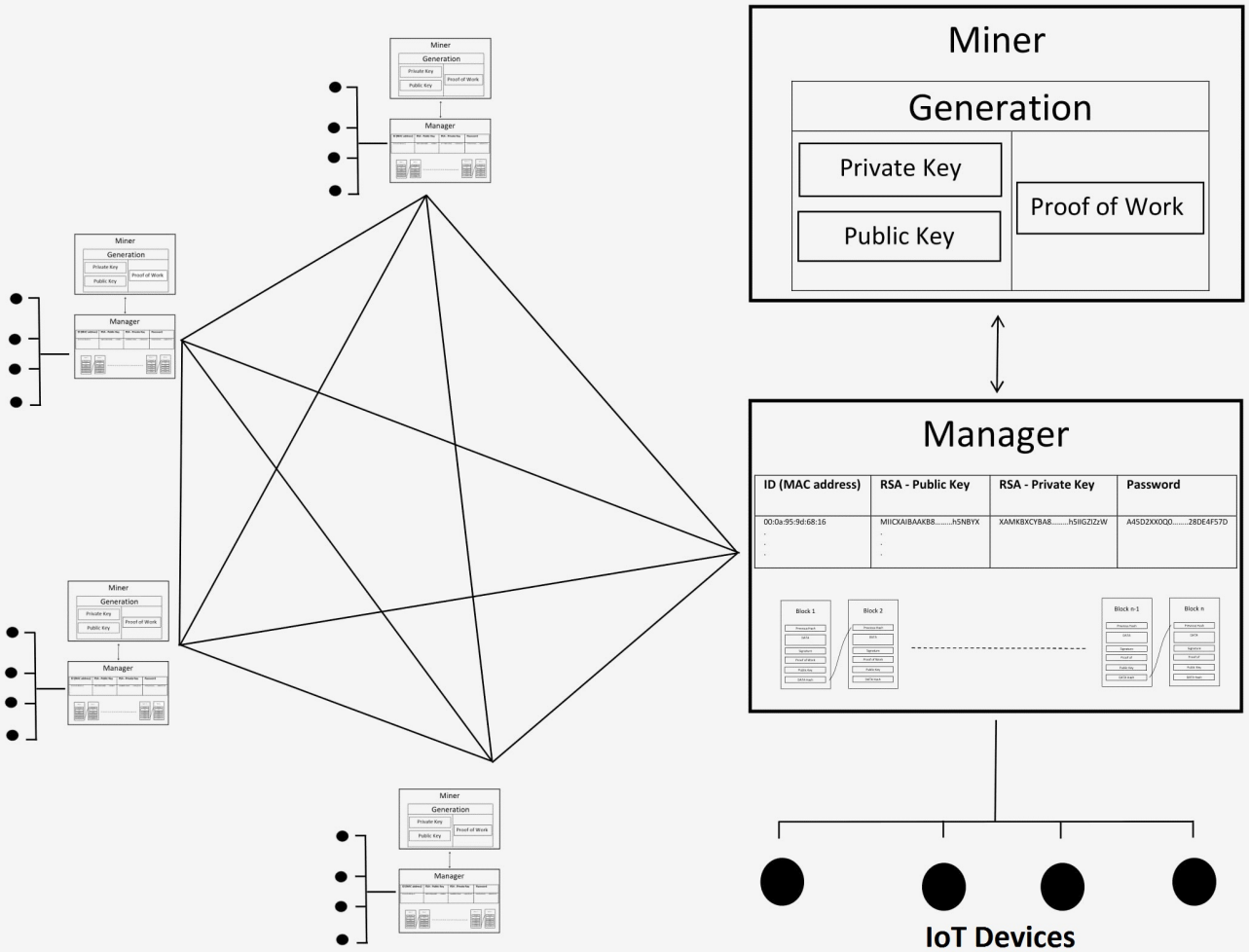
Hash Algorithm

Algorithm 1 Hash Blockchain Algorithm.

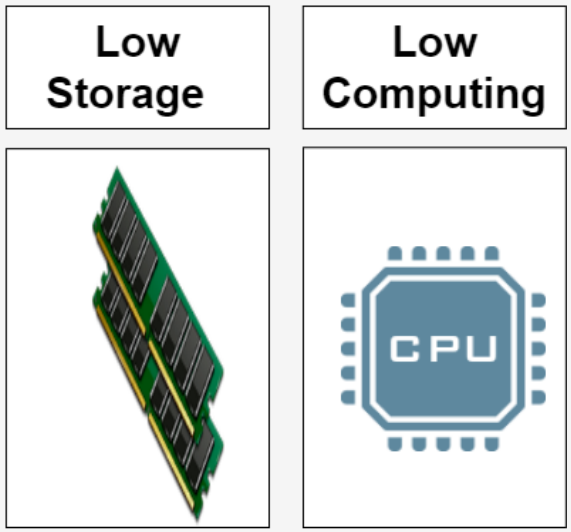
1: $Hash \leftarrow SHA5(SHA3(SHA2(SHA1(New\_DATA))))$
2: **if** $NB >= 10$ **then**
3: $\quad n \leftarrow 9$ ▷ NB is the size of the Blockchain.
4: **else**
5: $\quad n \leftarrow NB - 1$
6: **end if**
7: **for** $i \leftarrow NB, NB - n, i - -$ **do**
8: $\quad Hash \leftarrow SHA5(SHA3(SHA2(SHA1(Hash + Hash\_Block[i]))))$
9: **end for**
10: **return** $Hash$

- **Goal** $\longrightarrow$ Difficulty of finding the data

Blockchain Network

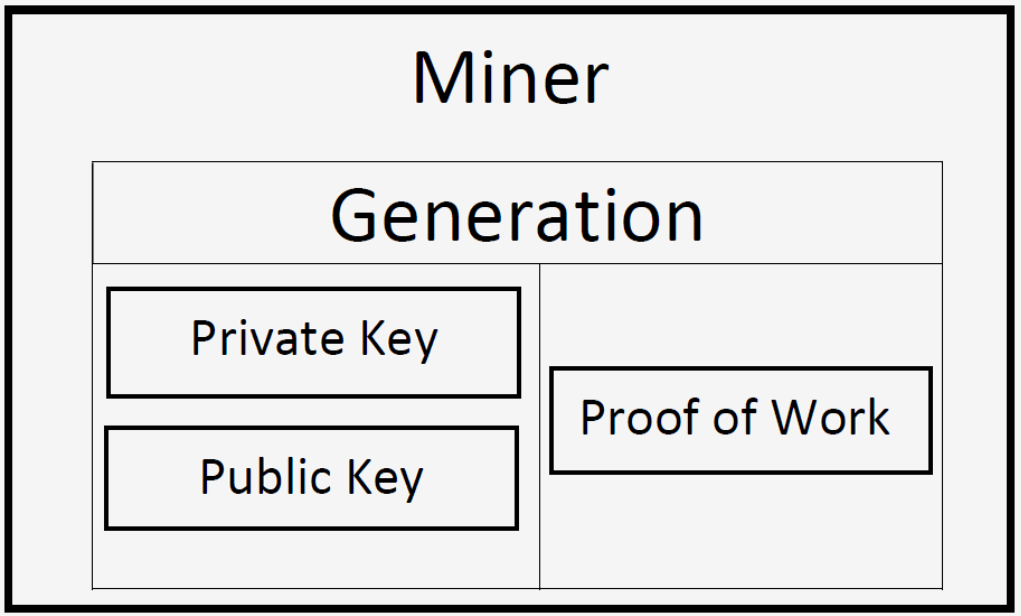


IoT Devices



Manager

ID (MAC address)	RSA - Public Key	RSA - Private Key	Password
00:0a:95:9d:68:16 . . .	MIICXAIBAAKB8.....h5NBYX . . .	XAMKBXCYBA8.....h5IIGZIZzW	A45D2XX0Q0.....28DE4F57D

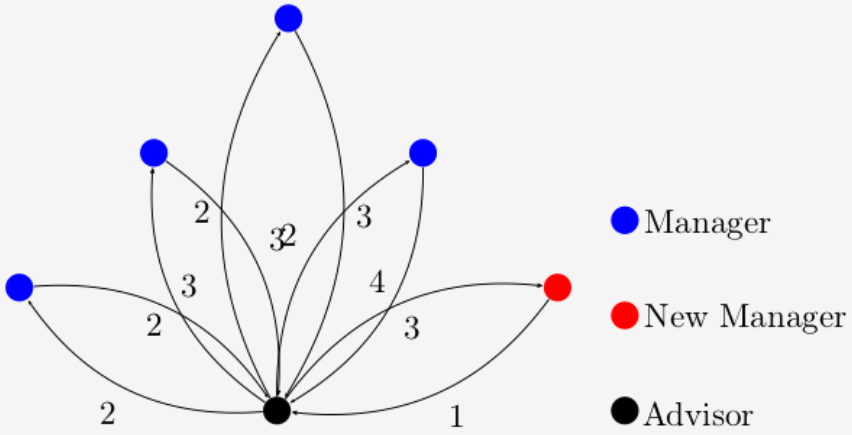


What is Proof of Work (PoW) ??

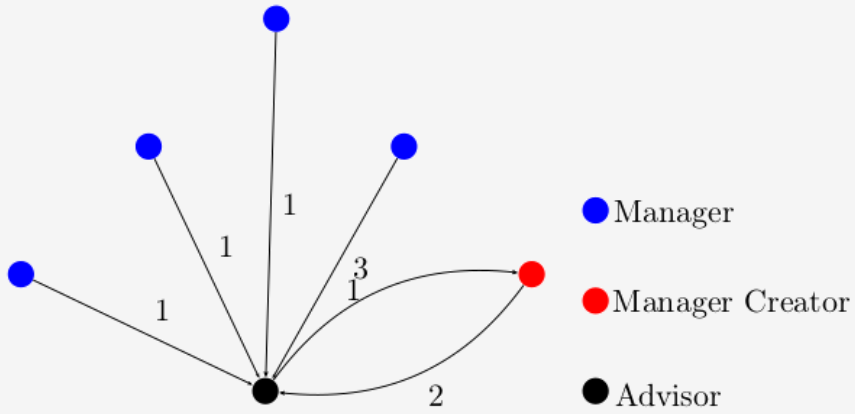
Exp: Condition = The hash must start by 10 zeros.

PoW + Data = Hash
>> Does "hash" respect the condition?

Advisor Tasks

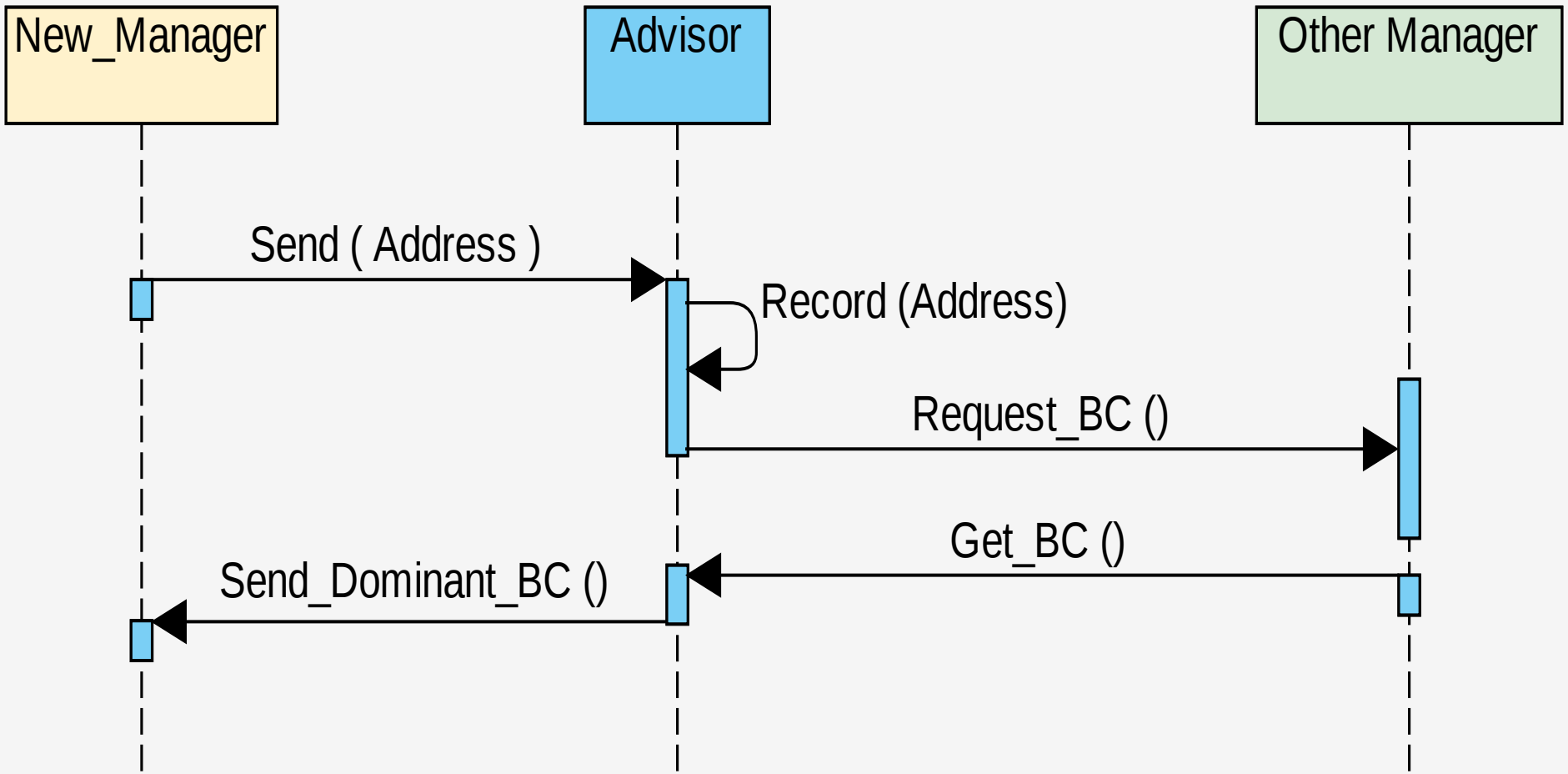


(a) Record New Manager

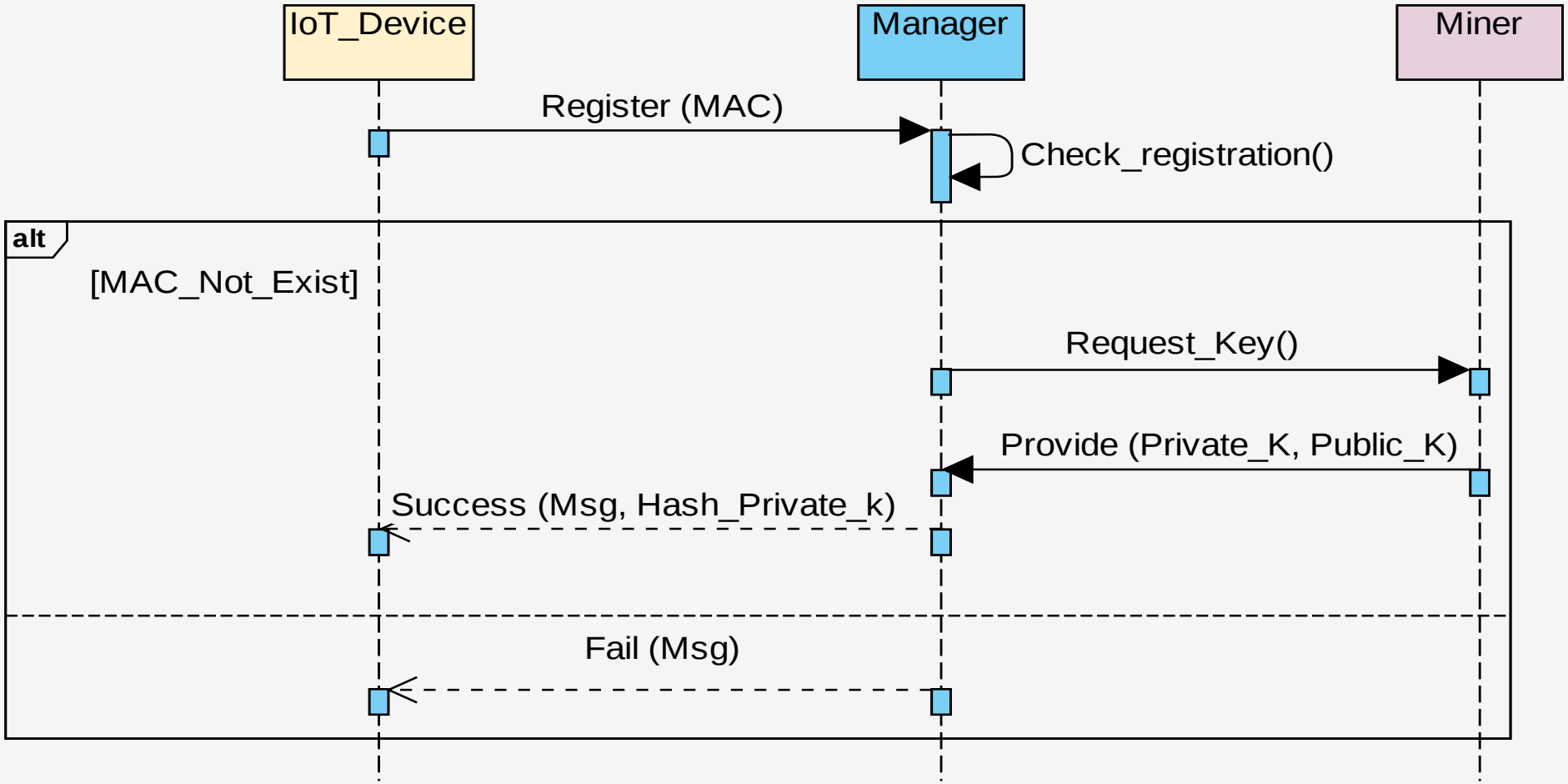


(b) Grant the Mangers Addresses.

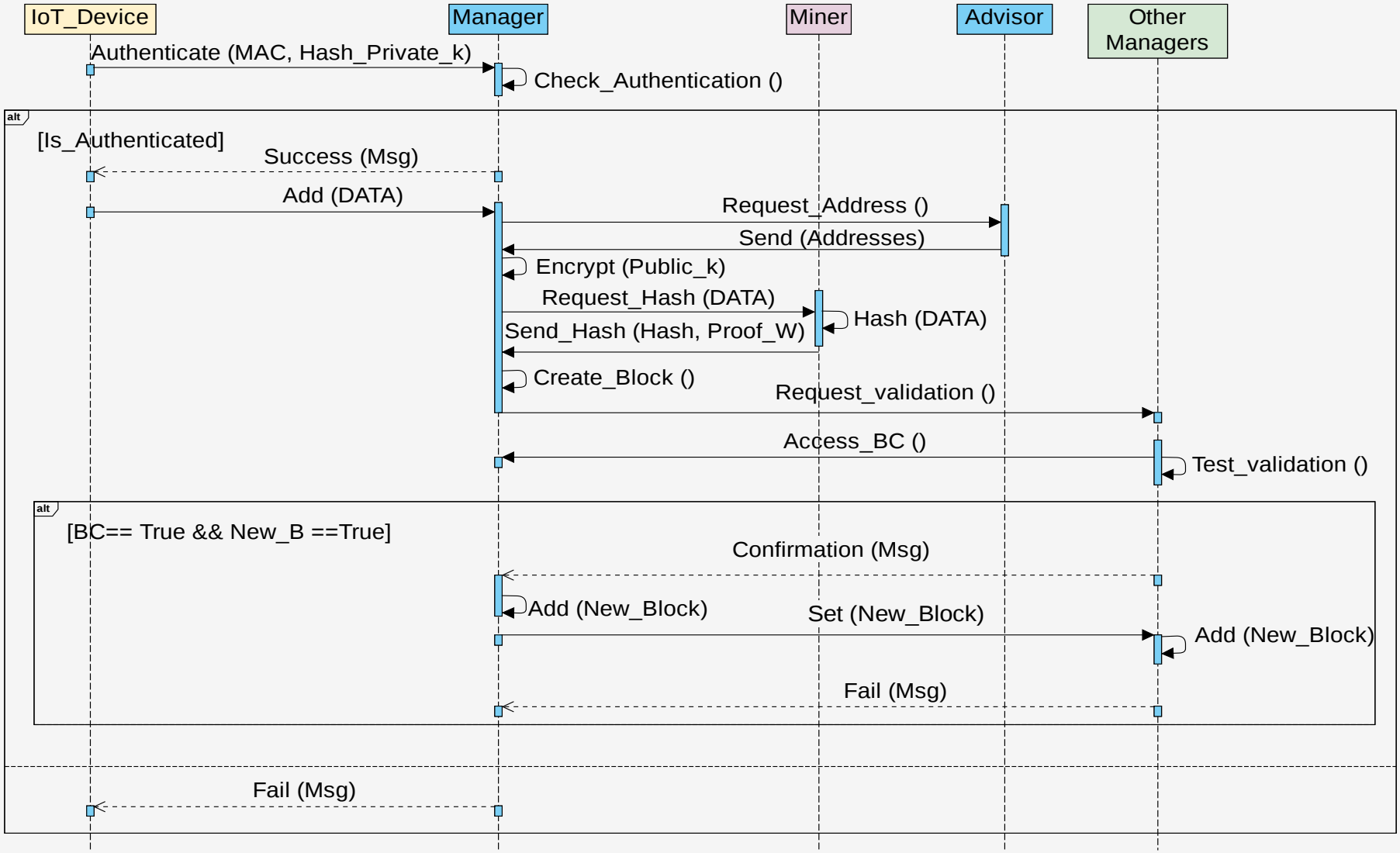
Add New Manager



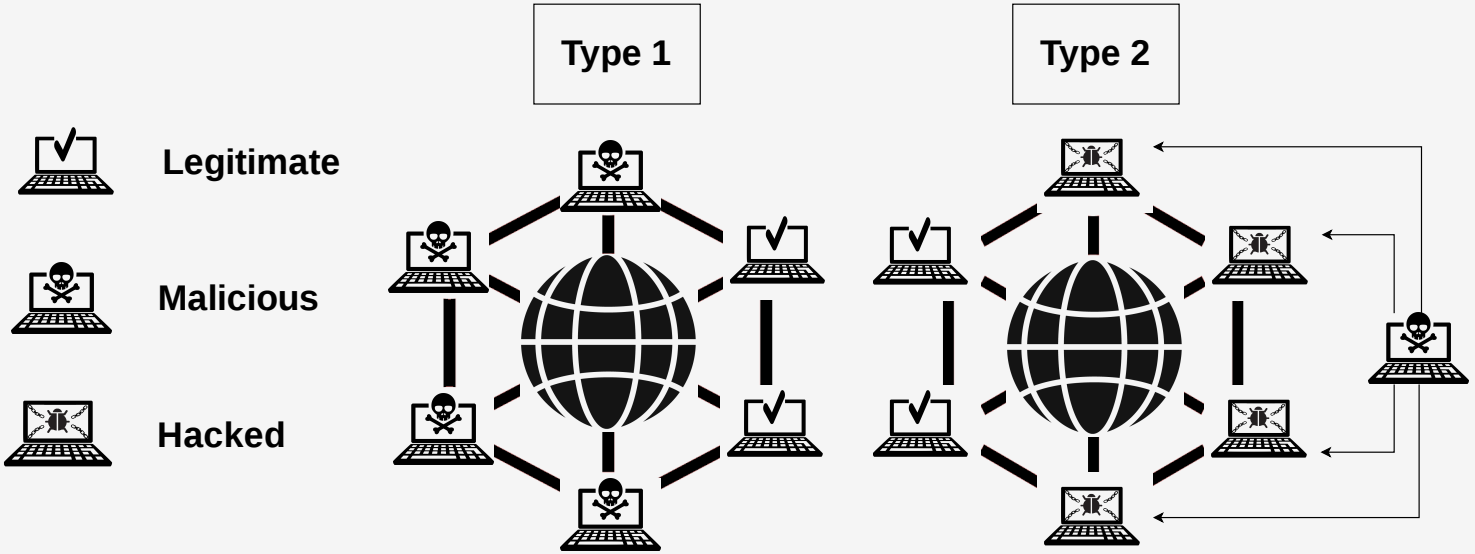
Registration Device in Manager



Add new block with a secure data



The dominance of fraud

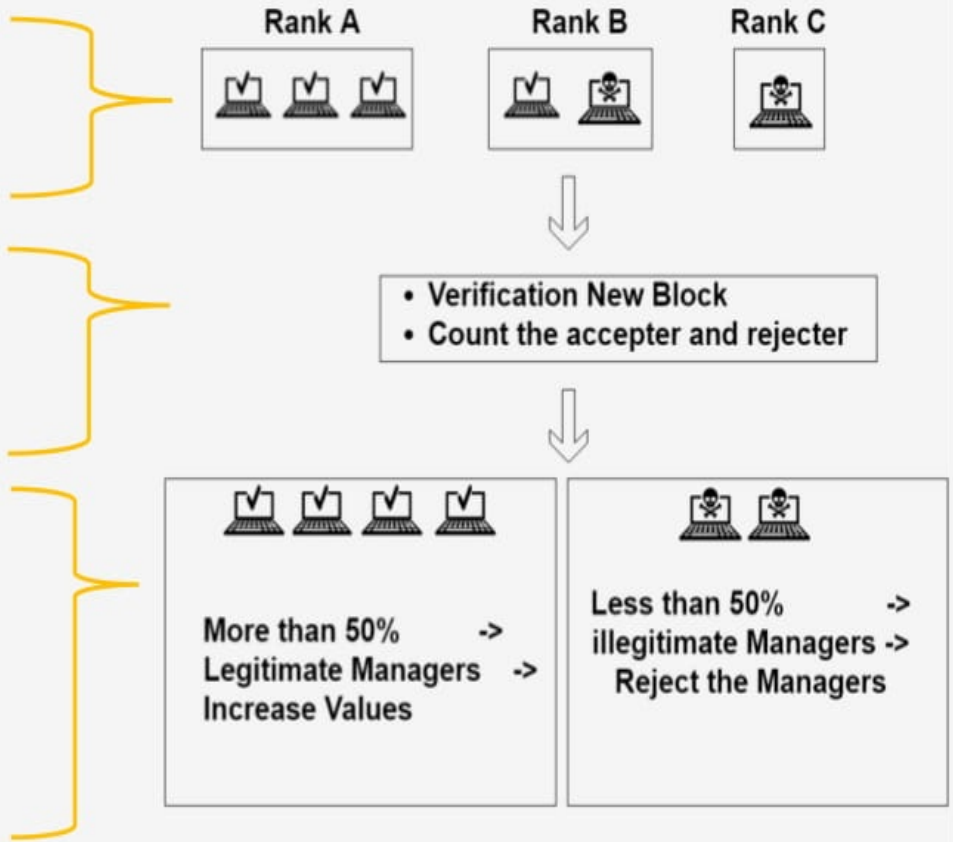


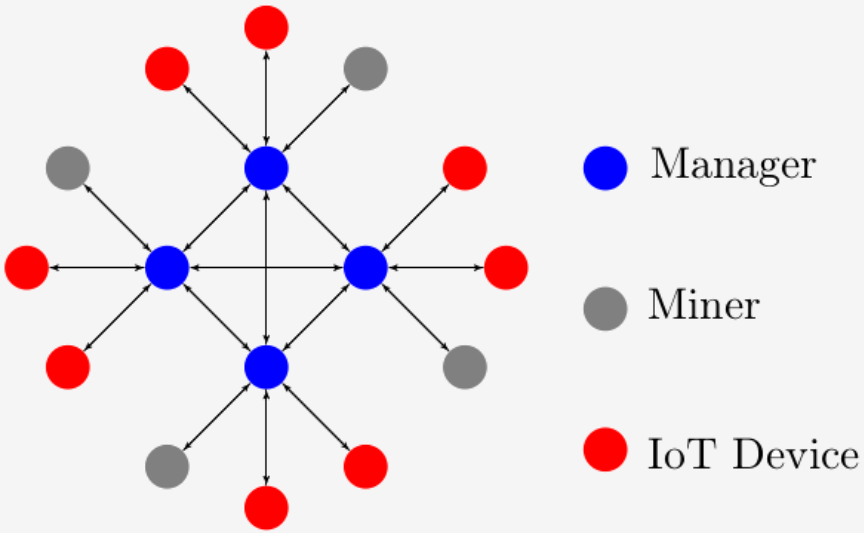
Dominance of fraud = Damaged BC is more than 50%

Solution of "Dominance of fraud"

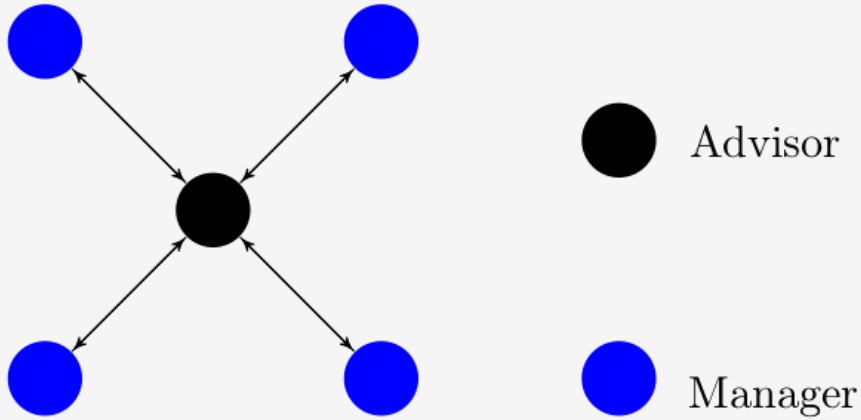
Algorithm Validation through the Confidence Algorithm (VCA)

```
1: procedure VCA(NewBlock)
2:     Define the rank and its ratio          ▷ e.g: Rank_A ← 60% ; Rank_B ← 20%.
3:     do
4:         for i ← 1, N do                    ▷ N: The number of the nodes chosen.
5:             Confidence [i] ← Random(Rank)  ▷ Fill the confidence table by random
   nodes.
6:         end for
7:         for i ← 1, N do
8:             Res ← Block_accept(Confidence [i], NewBlock)
9:             if Res == True then
10:                decision_Accept ++         ▷ Count the nodes which accept the new
   block.
11:            end if
12:        end for
13:    while decision_Accept_ratio == 50
14:    if decision_Accept_ratio > 50 then
15:        Add_Block(NewBlock)
16:        Increase_Node()    ▷ Increase the confidence criteria of the nodes which
   accept the new block.
17:        Delete_Node()                      ▷ Delete nodes which reject the new block.
18:    else
19:        Increase_Node()    ▷ Increase the confidence criteria of the nodes which
   reject the new block.
20:        Delete_Node()                      ▷ Delete nodes which accept the new block.
21:    end if
22: end procedure
```

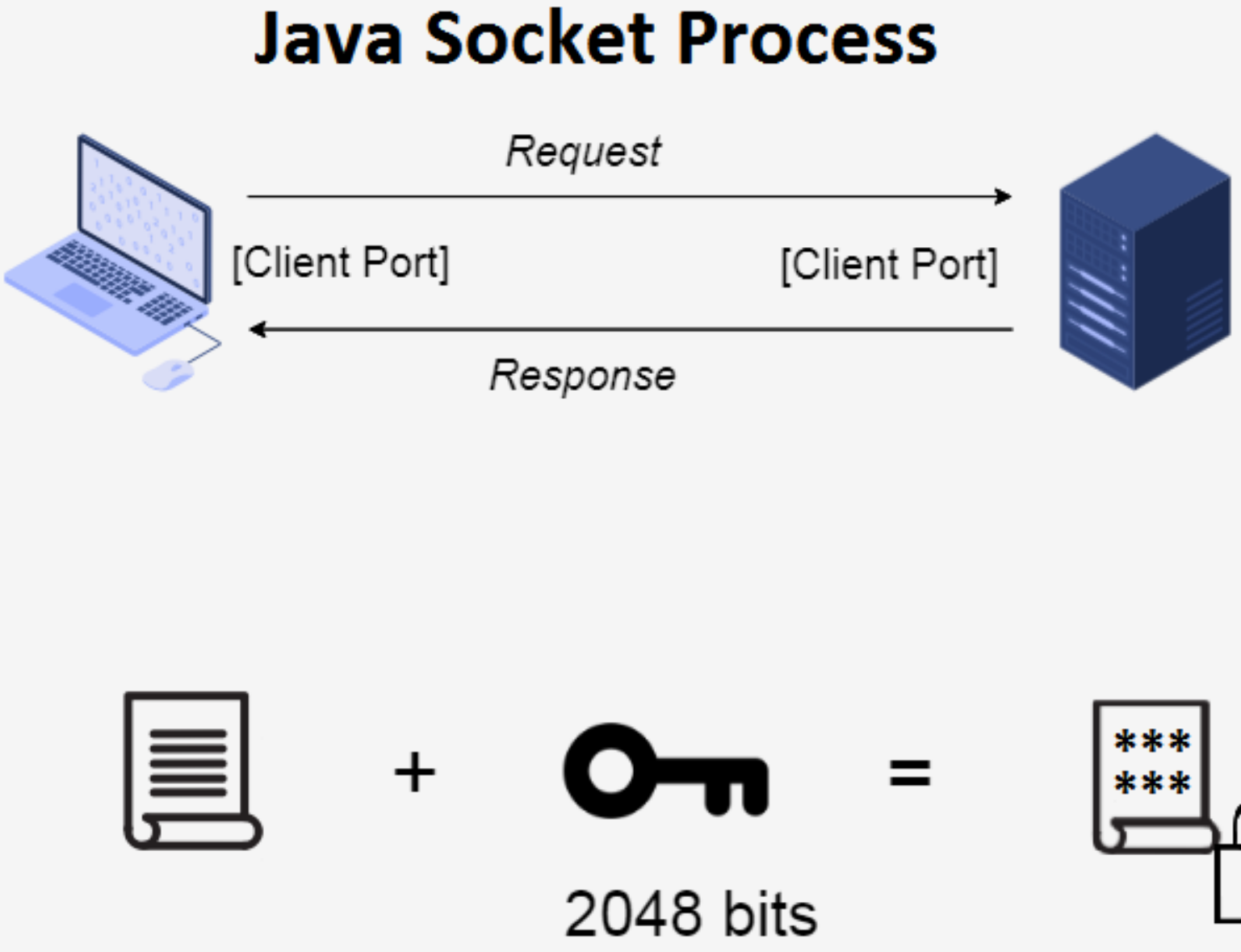




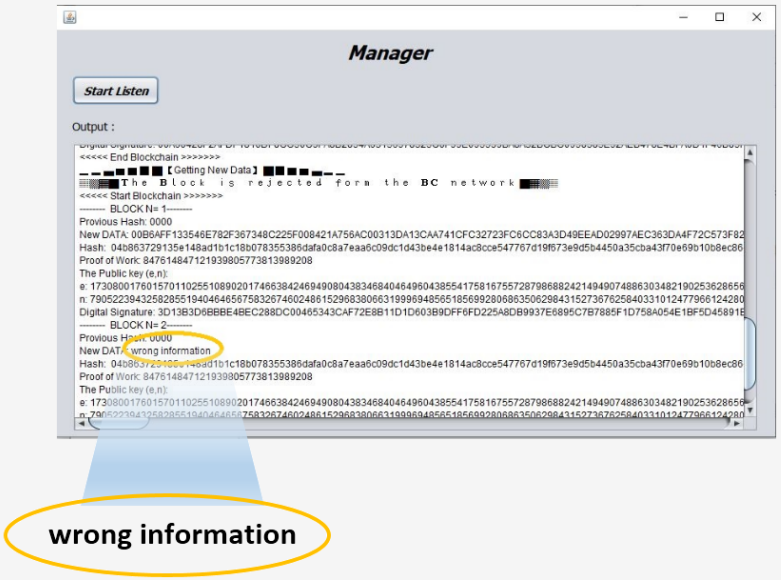
(a) The blockchain topology (IoT Devices, Miners and Managers).



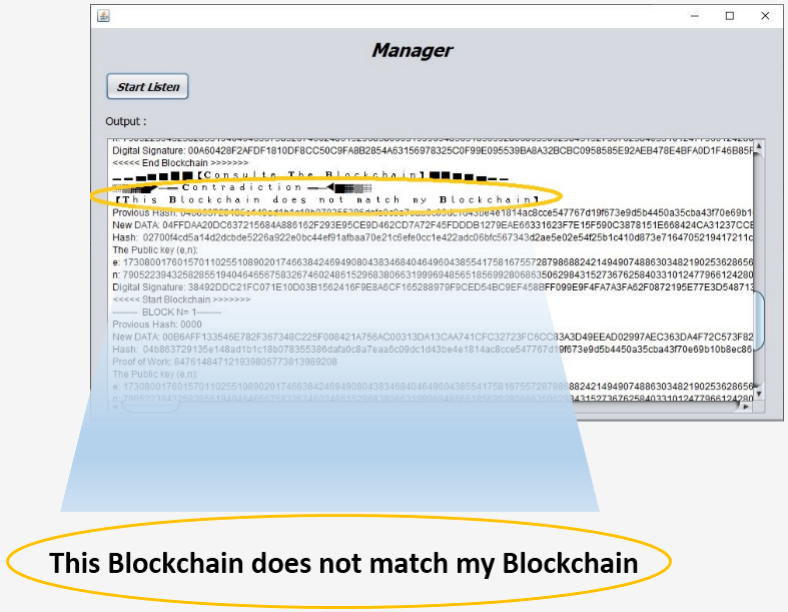
(b) The blockchain topology (Manager & Advisor).


Java Socket Process
Request
[Client Port]
[Client Port]
Response
+
=
***
***
2048 bits

Integrity Attack



(a) Damaged Manager



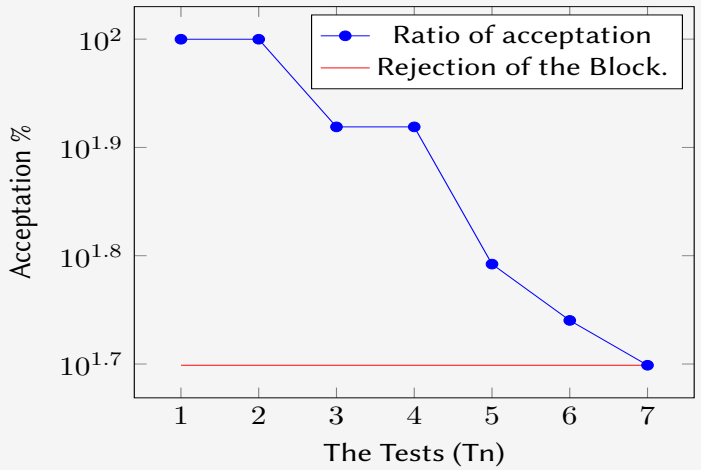
(b) Received Manager

The dominance of Attackers

VCA values

R_A= 80%, R_B= 60% and R_C= 10

Test	T1	T2	T3	T4	T5	T6	T7
Number of legitimates	10	10	10	10	10	10	10
Number of All managers	10	15	20	25	40	50	60
percentage of Attackers	0%	30%	50%	60%	75%	40%	83.3%
percentage of Acceptation	100%	100%	83%	83%	62%	55%	50%



We have proposed a framework, that:

- Serves the smart city digital word
- Consists on blockchain
- Protects through Asymmetric cryptography
- Prevent the domains of fraud attack
- Applies secure processes
- Was tested

We intend to:

- **Cover other protection mechanism**
- **Study device behavior**

Thank you for your attention!

Questions?