

Implementación de una blockchain resistente a ataques criptográficos cuánticos

Trabajo Fin de Grado

Autor

María Victoria Granados Pozo

Directores

Gabriel Maciá Fernández

Francisco Javier Lobillo Borrero

Doble grado de Ingeniería Informática y Matemáticas
Universidad de Granada

26 de Noviembre de 2020



**UNIVERSIDAD
DE GRANADA**

Introducción



BLOCKCHAIN

ALGORITMO CRIPTOGRÁFICO UOV

@mvictoria1997/TFG
@mvictoria1997/core

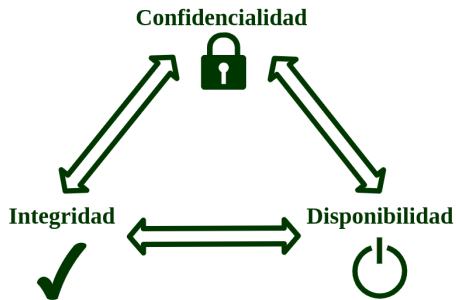


Figure: Pilares de la seguridad informática

Objetivos

Implementación del algoritmo UOV

Aritmética del cuerpo finito de 2^7 elementos y las funciones propias del algoritmo.

Integración del algoritmo UOV

Modificación del algoritmo de firma de la blockchain de ARK por el algoritmo UOV.

Tecnologías utilizadas

OpenProj

L^AT_EX



Contenidos teóricos

Blockchain

Descripción

Una cadena de bloques es un sistema de almacenamiento de información dividido en bloques de datos enlazados mediante el *hash*.

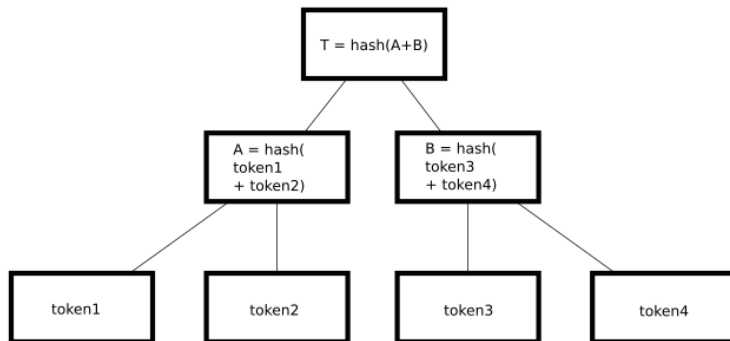


Figure: Estructura árbol de Merkle

Aplicaciones

- ◇ Área financiera o criptomonedas.
- ◇ Centros de salud.
- ◇ Firma de documentos.
- ◇ Cadenas de suministro.



Algoritmo UOV (*Unbalance Oil and Vinegar*)

Algoritmos post-cuánticos

Los algoritmos de clave pública basados en los problemas del logaritmo discreto y factorización de enteros, como Diffie-Hellman, RSA o ECDSA, se podrán romper fácilmente por un ordenador cuántico mediante el algoritmo de Shor.

Los algoritmos simétricos se podrían atacar con el algoritmo de Grover.

Los algoritmos criptográficos post-cuánticos son algoritmos resistentes a ataques cuánticos, un ejemplo es el algoritmo UOV.

Ventajas del algoritmo UOV

- ▲ Problema NP-duro.
- ▲ No se conoce un algoritmo eficiente para la resolución de sistemas multivariados en un ordenador cuántico.
- ▲ Simplicidad de las operaciones.
- ▲ Requiere bajos recursos *hardware*.

Clave privada

- $(\alpha_{i,j,k})_{\substack{1 \leq i \leq v \\ 1 \leq j \leq n}}$
- $(\beta_{i,k})_{1 \leq i \leq v}$

Clave pública

- $\alpha_{pub_k} = \left(\frac{I_v}{T_{v \times m}^T} \right) (\alpha_{i,j,k})_{\substack{1 \leq i \leq v \\ 1 \leq j \leq n}} T$
- $\beta_{pub_k} = (\beta_{j,k})_{1 \leq j \leq n} T$

$$firma = x \cdot (T^T)^{-1} = x \cdot T^T$$

donde $x = (x^v, x^m)$ con x^v variables de vinagre y x^m variables de aceite

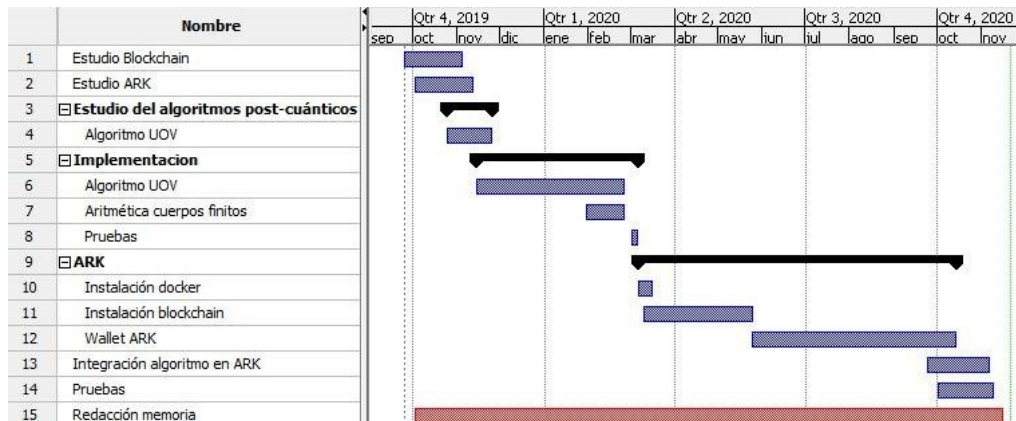
$$(x^m)^T = (A_k^m + \beta_k^m)^{-1} (hash_k - (A_k^v + \beta_k^v)(a^v)^T)$$

$$\text{con } A_k = a^v \left(\alpha_{i,j,k} \right)_{\substack{1 \leq i \leq v \\ 1 \leq j \leq n}} = (A_k^v, A_k^m)$$

$$\text{hash}_k = \alpha_{\text{pub}_k} s^T + \beta_{\text{pub}_k} s^T \quad k \in \{1, \dots, n\}$$

Planificación y presupuesto

Diagrama de Gantt



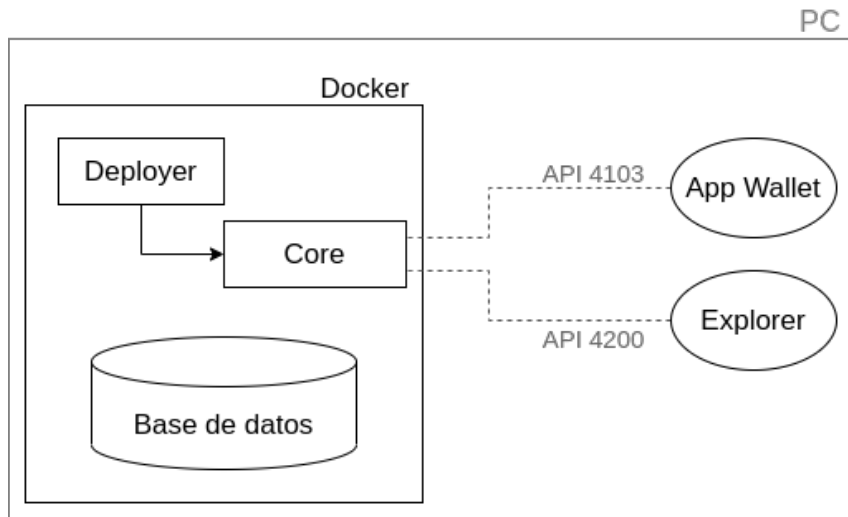
Presupuesto desglosado

Tipo de costes	Cantidad,
Recursos humanos tutores	4.830,00€
Recursos humanos alumna	10.720,00€
Indirectos	1.578,24€
Directos	210,40€
Viajes	22,00€
Gastos imprevistos	868,03€
TOTAL (€)	18.228,67€

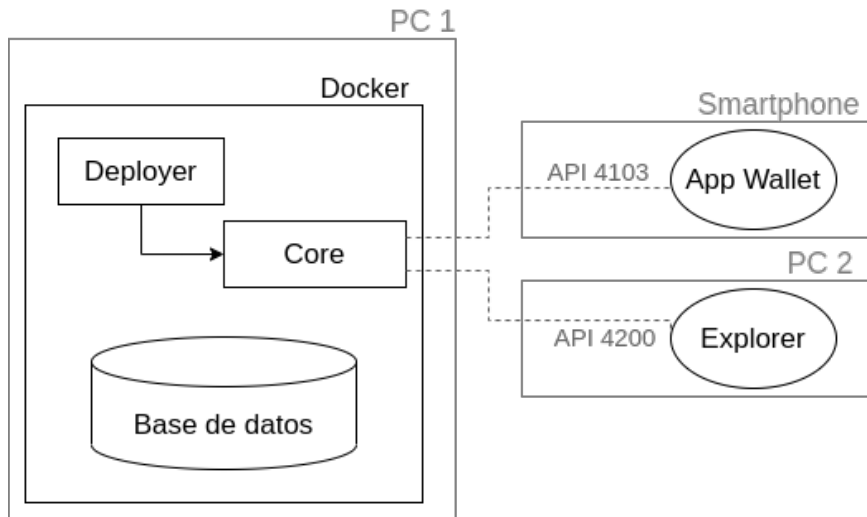
Table: Presupuesto total desglosado

Diseño

Configuración de los bloques



Otra posible configuración de los bloques

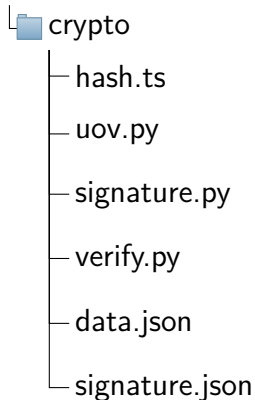


Implementación

Estructura directorio

core-bridgechain/packages/crypto/src/crypto

core-bridgechain/packages/crypto/src

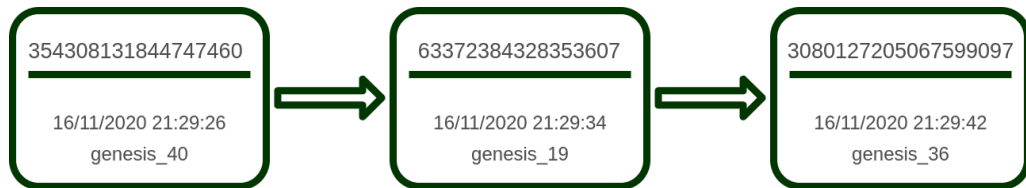


Problemas encontrados

- ▼ Necesidad de implementar la aritmética del cuerpo finito de 128.
- ▼ Adaptar las claves de la *blockchain* a las del algoritmo UOV.
- ▼ La función de firma devuelve el identificador de la firma.

Ejemplo práctico

Cadena de bloques



Logs terminal

Firma del bloque


```
2|bridgechain-forgery | signature UOV
2|bridgechain-forgery | python /home/deployer/core-bridgechain/packages/crypto/dist/crypto/../../src/crypto/signature.py 182,85,243,222,157,82,40,158,70,45,235,73,201,34,
239,216,202,182,62,76,120,58,108,176,141,155,20,225,79,192,7,184 02fbefc34e84ca97fa3f1393a7309a4e6964296ae0c91049de57d7365372c89295 96b6d0ee7372521079de79ef94a62f4c4143e
21e5c9e2b59ec2a6b0e696999a6
2|bridgechain-forgery | [[1, 1, 1, 0, 1, 0, 1], [0, 1, 0, 1, 1, 1, 1], [1, 1, 0, 1, 1, 1, 1], [1, 0, 0, 0, 0, 0, 1], [0, 0, 1, 1, 1, 0, 0], [1, 0, 1, 0, 1, 1, 0]]
2|bridgechain-forgery | 5b5b312c20312c20312c20302c20312c20302c20315d2c205b302c20312c20302c20312c20312c20315d2c205b312c20312c20302c20312c20312c20315d2c205b312c
20302c20302c20302c20302c20315d2c205b302c20302c20312c20312c20302c20305d2c205b312c20302c20312c20302c20312c20305d5d0a
2|bridgechain-forgery | verifica UOV
2|bridgechain-forgery | [[1,1,1,0,1,0,1],[0,1,0,1,1,1,1],[1,1,0,1,1,1,1],[1,0,0,0,0,0,1],[0,0,1,1,1,0,0],[1,0,1,0,1,1,0]]
2|bridgechain-forgery | python /home/deployer/core-bridgechain/packages/crypto/dist/crypto/../../src/crypto/verify.py 182,85,243,222,157,82,40,158,70,45,235,73,201,34,239
,216,202,182,62,76,120,58,108,176,141,155,20,225,79,192,7,184 [[1,1,1,0,1,0,1],[0,1,0,1,1,1,1],[1,1,0,1,1,1,1],[1,0,0,0,0,0,1],[0,0,1,1,1,0,0],[1,0,1,0,1,1,0]] 02fbefc34
e84ca97fa3f1393a7309a4e6964296ae0c91049de57d7365372c89295
2|bridgechain-forgery | True
2|bridgechain-forgery | [2020-11-16 20:29:36.442] INFO - Forged new block 63372384328353607 by delegate genesis_19 (02fbefc34e84ca97fa3f1393a7309a4e6964296ae0c91049de57d7
365372c89295)
2|bridgechain-forgery | [2020-11-16 20:29:36.444] DEBUG - Broadcasting block 6 (63372384328353607) with 0 transactions to 127.0.0.1
1|bridgechain-relay | [2020-11-16 20:29:36.456] INFO - Received new block at height 6 with 0 transactions from 127.0.0.1
1|bridgechain-relay | [2020-11-16 20:29:36.456] INFO - Previous block 5 pinged blockchain 0 times
1|bridgechain-relay | [2020-11-16 20:29:36.460] DEBUG - event 'NEWBLOCK': "idle" -> "newBlock"
```

Verificación del bloque

Broadcasting block 6 (63372384328353607)

Forged new block 63372384328353607 by delegate genesis_19

Visualización ARK Explorer



Menu

Find a block, transaction, address or delegate

Q

🔗

Latest transactions and blocks

Height: 7 Network: Testnet Local Supply: 21,000.014 M

Latest transactions

Latest blocks

ID	Height #	Timestamp	Transactions	Generated by	Total forged	Fees
15242...96468	8	16/11/2020 21:29:50	0	genesis_42	2 M	0 M
30801...99097	7	16/11/2020 21:29:42	0	genesis_36	2 M	0 M
63372...53607	6	16/11/2020 21:29:34	0	genesis_19	2 M	0 M
35430...47460	5	16/11/2020 21:29:26	0	genesis_40	2 M	0 M
93449...63789	4	16/11/2020 21:29:18	0	genesis_41	2 M	0 M
38137...13698	3	16/11/2020 21:29:10	0	genesis_21	2 M	0 M
63710...41921	2	16/11/2020 21:29:02	0	genesis_12	2 M	0 M
98561...96001	1	16/11/2020 21:11:26	52	TVUTR...QI4m3	0 M	0 M

Visualización ARK Explorer ampliada

30801...99097

7

16/11/2020 21:29:42

63372...53607

6

16/11/2020 21:29:34

35430...47460



5

16/11/2020 21:29:26

Visualización ARK Explorer del bloque con ID 63372384328353607

Block

Height: 10 Network: Testnet Local Supply: 21.000.020 M

 Block ID
63372384328353607 

[< Previous block](#) [Next block >](#)

Transactions	0
Confirmations	4
Height	6
Reward	2 M
Fees	0 M
Total forged	2 M
Processed amount	0 M
Timestamp	16/11/2020 21:29:34
Generated by	genesis_19

Visualización ARK API del bloque con ID 63372384328353607

```
{
  "id": "63372384328353607",
  "version": 0,
  "height": 6,
  "previous": "3543081318344747460",
  "forged": true,
  "reward": "200000000",
  "fee": "0",
  "total": "200000000",
  "amount": "0",
  "payload": "0",
  "hash": "e3b0c44298fc1c149afb4c8996fb92427ae41e4649b34ca495991b7852b855",
  "length": 0,
  "generator": {
    "username": "genesis_19",
    "address": "TFy6ASk5k5cohnrv32LAZR0BNiDqPVTXXe",
    "publicKey": "02fbefc34e84ca97fa3f1393a7309a4e6964296ae0c91049de57d7365372c89295",
    "signature": "5b5b312c20312c20312c20302c20312c20315d2c205b302c20312c20302c20312c20312c20315d2c205b312c20312c20302c20312c20312c20315d2c205b312c20302c20302c20302c20315d2c20",
    "confirmations": 6,
    "transactions": 0,
    "timestamp": {
      "epoch": 1088,
      "unix": 1605558574,
      "human": "2020-11-16T20:29:34.125Z"
    }
  }
}
```

63372384328353607

Previous: 3543081318344747460

Hash: e3b0c44298fc1c149afb4c8996fb92427ae41e4649b34ca495991b7852b855

Firma Vector: [[1, 1, 1, 0, 1, 0, 1], [0, 1, 0, 1, 1, 1, 1], [1, 0, 0, 0, 0, 0, 1], [0, 0, 1, 1, 1, ...

Firma Hex: 5b5b312c20312c20312c20302c20312c20302c20312c205d2c205b302c2....

Generator: {username: genesis_19,
address: TFy6ASk5k5cohnrv32LAZR0BNiDqPVTXXe,
publicKey: 02fbefc34e84ca97fa3f1393a7309a4e6964296ae0c91049de....}

Confirmations: 6

Timestamp: {human: 2020-11-16T20:29:34.12}

Conclusiones y desarrollos futuros

Conclusiones

- ✓ Implementación aritmética del cuerpo finito de 128 elemento y algoritmo UOV.
- ✓ Comparación de los tiempos de ejecución en python y SageMath.
- ✓ Integración del algoritmo en la *blockchain* de ARK.
- ✓ Ejecución de la *blockchain* de ARK modificada.
- ✓ Comprobación de los bloques firmados en el *explorer* de ARK y en la API.
- ✓ Cadena de bloques resistente a ataques cuánticos.

Desarrollos futuros

- Trabajar con la base de datos en lugar de tener archivos json independientes.
- Integrar la *blockchain* ARK modificada en otra cadena de bloques.

Ejemplo UOV

Generación de claves

$$\alpha = [[[1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1] \dots$$

$$\beta = [[1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0] \dots$$

$$\alpha_{pub} = [[[1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0] \dots$$

$$\beta_{pub} = [[[1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1] \dots$$

El mensaje a firmar es "Ejemplo del algoritmo UOV para la presentación. Este mensaje es un mensaje de prueba. Quiero se sea un poco largo para que se aprecie el efecto de la función hash", cuyo *hash* es "bafd830504".

$firma = [[0, 1, 1, 1, 0, 0, 1], [1, 0, 1, 1, 0, 0, 0], [1, 1, 1, 1, 0, 1, 1], [1, 1, 0, 0, 0, 0, 1], [0, 0, 1, 0, 1, 1, 1]]...$

$$\text{hash_binario} = [1011, 1010, 1111, 1101, 1000, 11, 0, 101, 0, 100]$$

$$[0, 0, 0, 1, 0, 1, 1] = [1, 0, 0, 0, 0, 0, 0] + [1, 0, 0, 1, 0, 1, 1]$$

$$[0, 0, 0, 1, 0, 1, 0] = [0, 0, 0, 0, 1, 1, 0] + [0, 0, 0, 1, 1, 0, 0]$$

$$[0, 0, 0, 1, 1, 1, 1] = [0, 1, 1, 1, 0, 0, 1] + [0, 1, 1, 0, 1, 1, 0]$$

$$[0, 0, 0, 1, 1, 0, 1] = [1, 0, 0, 1, 1, 1, 0] + [1, 0, 0, 0, 0, 1, 1]$$

$$[0, 0, 0, 1, 0, 0, 0] = [1, 0, 1, 0, 0, 1, 0] + [1, 0, 1, 1, 0, 1, 0]$$

¡Gracias por su atención!