EL BLOCKCHAIN

SISTEMAS INFORMÁTICOS (1ªeva)

ERIC MOROS PÉREZ

1º DAW (DUAL)

Contenido

[Bibliografía 2](#_Toc499637247)

[¿Qué es blockchain? 3](#_Toc499637248)

[¿Cómo funciona la validación de blockchain? 3](#_Toc499637249)

[¿Cómo se almacenan los datos? 4](#_Toc499637250)

[Características/Configuraciones 5](#_Toc499637251)

[Acceso a los datos 5](#_Toc499637252)

[ Pública 5](#_Toc499637253)

[ Privada 5](#_Toc499637254)

[Permisos 5](#_Toc499637255)

[ Activados 5](#_Toc499637256)

[ Desactivados 5](#_Toc499637257)

[Conclusiones 5](#_Toc499637258)

# Bibliografía

*bit2me*. (19 de Noviembre de 2017). Obtenido de http://blog.bit2me.com/es/

Brownworth, A. (28 de 11 de 2017). *Blockchain Demo*. Obtenido de https://anders.com/blockchain/

*es.wikipedia.org*. (26 de 11 de 2017). Obtenido de https://es.wikipedia.org/wiki/Problema\_de\_los\_generales\_bizantinos

*slideshare.net* . (28 de 11 de 2017). Obtenido de https://www.slideshare.net/boolberry/boolberry-reduces-blockchain-bloat

# ¿Qué es blockchain?

Como primer punto básico lo que hay que entender es que blockchain es una base de datos, solo que, a diferencia de las habituales, ha sido diseñada para ser lineal en el tiempo e inmodificable.

Una vez sabemos esto, ya se nos quita de la cabeza el que por el hecho de que Bitcoin usara este concepto por primera vez en 2009 esté extralimitado a almacenar las transacciones de algo “tan sencillo como una criptomoneda”.

Y como segundo punto es importante destacar que es una base de datos distribuida, lo que significa que no funciona con un equipo servidor o varios con el cual se accede a los datos, funciona a base de nodos que mantienen una copia de la base de datos la cual se la distribuye con el fin de mejorar su seguridad, su velocidad y su estabilidad contra la perdida de datos (La cual debería ser imposible siempre y cuando quede un nodo en pie).

# ¿Cómo funciona la validación de blockchain?

Existe un ejemplo principal al cual es habitual referirse (Probablemente porque será en el que se basaron para crear el software). El problema matemático de los generales bizantinos: En el cual se presentan varias situaciones, en las que los generales han de comunicarse entre ellos a base de mensajeros, con la intención de coordinarse para atacar o retirarse en las distintas ciudades a las cuales están atacando, pero el problema se encuentra en que entre esos generales hay un traidor y hay que descubrirlo para ponerse de acuerdo (Prefiero proponer el mío propio).

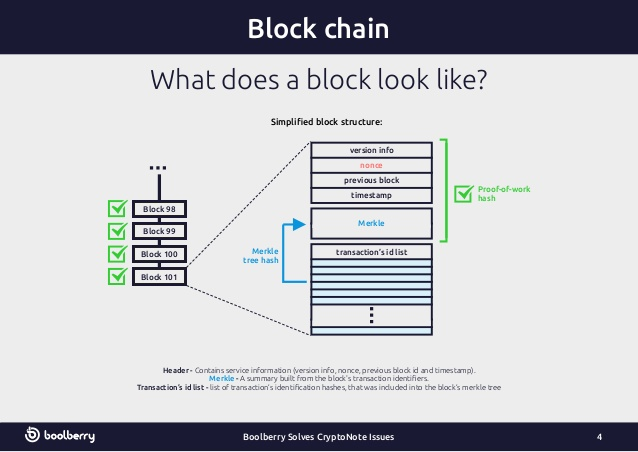
*(Para leer sobre dicho problema encontrarás información siguiendo este enlace:* [*https://es.wikipedia.org/wiki/Problema\_de\_los\_generales\_bizantinos*](https://es.wikipedia.org/wiki/Problema_de_los_generales_bizantinos)*)*

Pongamos que un profesor de sistemas nos pone un trabajo en el cual debemos de hablar sobre el tema que nos de la gana con algunas condiciones, las cuales no especifica, nos dice que nos mandará al e-mail y además que entregaremos a un profesor sustituto porque él se encontrará ausente (Este profesor es muy estricto, y el que no siga al pie de la letra las instrucciones automáticamente tendrá un 0 en la evaluación). Al día siguiente aprovechando que el profesor sustituto tarda en venir porque hay un problema con la red del instituto, los 12 alumnos nos ponemos a comparar trabajos por aburrimiento, pero resulta que 4 de clase han hecho un trabajo de 500 palabras en lugar de 1000 como se especificaba en el e-mail, total que el resto de alumnos les comentamos sobre dicha condición, pero dicen que en su e-mail ponía que era solo de 500. ¿Quién crees que tiene razón?

Pues siguiendo el algoritmo que usa blockchain, esos 4 alumnos se equivocan, porque el porcentaje de alumnos que están de acuerdo en que el trabajo era de 1000 palabras y no de 500 es superior al que dice lo contrario. Y por ello en nuestro caso ellos son los traidores que pretendían hacer un trabajo más pequeño y van a tener un 0 en la evaluación.

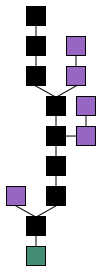
También dejo este enlace a otro ejemplo sobre un suricato gigante volador que vi en una plaza el otro día: <http://blog.bit2me.com/es/que-es-cadena-de-bloques-blockchain/>

# ¿Cómo se almacenan los datos?

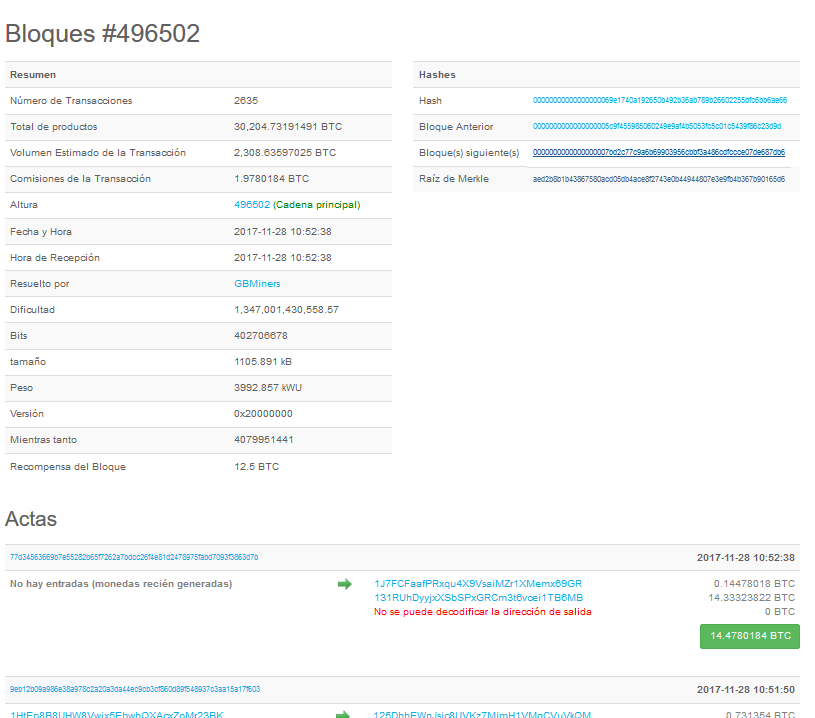
Los datos se almacenan e indexan en un árbol de Merkel.

Luego **[**se incluyen con el resto de atributos: Versión, fecha y hora, el bloque anterior y el nonce (Que servirá para sustituir temporalmente al bloque siguiente).

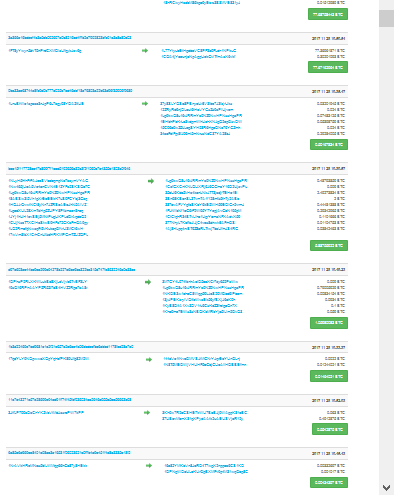
Y finalmente con todo ello se genera un hash, que identificará al propio bloque**]** ✔.

En caso de que se pudiera modificar los datos de **un bloque** todos los bloques hijos que se hubiesen generado quedarían invalidados y se convertirían en **bloques huérfanos** que no servirían para nada. Pero no significa que todos los datos se vallan a perder, puesto que los mineros pueden validar los datos de los bloques huérfanos y generar con ellos unos nuevos.

En este gráfico además se está representando un bloque verde queriendo destacar el **bloque matriz**: Que es el primer bloque del que parten todos los demás.

Este es un ejemplo del resultado de un bloque de la blockchin de bitcoin.

Presentado en la web de <https://blockchain.info/es/>



# Características/Configuraciones

Se pueden hacer múltiples combinaciones de las capacidades de las siguientes clasificaciones en todos los niveles.

## Acceso a los datos

### Pública

Todos los datos son explorables y todas las entidades pueden incluir información para la cual se halla preparado la cadena de bloques.

### Privada

Tanto la exploración como la inclusión de información está limitada a unas entidades definidas.

## Permisos

### Activados

Las entidades que crearán nuevos bloques estarán definidas además de los validadores. Con lo cual no es necesario aplicar incentivos

### Desactivados

Cualquier entidad puede participar en la creación de la cadena de bloques y en el proceso de validación los datos. Para incentivar la participación, se aplican recompensas por dicha creación y validación. Puesto que requiere altos costes computacionales.

# Conclusiones