A blockchain-based property ownership recording system

by Alex Mizrahi alex.mizrahi@chromaway.com

Overview

For any kind of a high-value property (real estate, cars, art) it is important to have accurate records which identify the current owner and provide a proof that he is indeed the owner. These records can be used to

- protect owners' rights (e.g. in case of theft)
- resolve disputes
- make sure ownership is correctly transferred to a new owner after sale
- prevent sale fraud

Thus it is crucial to maintain correctness and completeness of this information, and prevent unauthorized, fraudulent changes.

From the point of view of a computer security expert, currently people have to rely on a trusted third party. E.g. a government agency might be responsible for keeping track of ownership information. Sometimes, these records are not preserved in a systematic way.

Is it possible to keep track of property ownership through some kind of a distributed system which won't rely on trust? What would it require?

At minimum, we need a consensus about the current owner and ability for that owner to identify himself.

The same problem was solved by Satoshi Nakamoto when he created:

- consensus is established using the blockchain (which keeps records of previous transactions) and proof-of-work which makes changing historic records prohibitively costly
- correctness is guaranteed by protocol rules
- onwer can be identified using public key cryptography

Naturally, we would want to re-use this solution. In principle, any protocol with similar properties can be useful for keeping property ownership records.

But it is also possible to re-use Bitcoin itself. Conventionally, bitcoins are fungible. However, it is possible to create non-fungible tokens by tracking a specific "coin" through the transaction history, which is preserved in the blockchain. This concept is known as colored coins, although it was first described by Mike Hearn in his Smart Property article:

> Smart property is property whose ownership is controlled via the Bitcoin block chain, using contracts. Examples could include physical property such as cars, phones or houses. Smart property also includes non-physical property like shares in a company or access rights to a remote computer. Making property smart allows it to be traded with radically less trust. This reduces fraud, mediation fees and allows trades to take place that otherwise would never have happened. For example, it allows strangers to loan you money over the internet taking your smart property as collateral, which should make lending more competitive and thus credit cheaper.

Let's go through the process in detail:

1. At some point, a property is associated with a certain transaction output, which is called a 'genesis transaction output'. That transaction output (coin) belongs to the initial owner recorded by the system. (In Mike Hearn's example with a car, a genesis transaction output is established on a factory which have produced a car, and that factory is the initial owner.)
2. When the property is sold or transferred, a transaction output which belongs to the previous owner is spent, and a transaction output which belongs to a new owner is created in the same transaction, which needs to be created according to certain rules.
3. When somebody needs to identify an owner, he will go through the transaction history starting from the genesis transaction up to an unspent transaction output. The owner of the unspent transaction output is the current owner of the property. Bitcoin blockchain has his Bitcoin address (in a general case, scriptPubKey), and he is able to prove that he is the owner by signing a message with the private key associated with that address (in general case, by producing scriptSig which satisfies scriptPubKey).

It is important to note that in this case property ownership is associated with a certain private key rather than with a certain person. If we assume that only one person is in possession of that private key, the effect is the same. However, a private key can be lost or stolen. Also, in some cases a legal system (courts) might override ownership which was recorded in the blockchain. We must take this into account if we want to build a robust system.

In the following sections we will outline possible ways to exgtend this basic system and analyze its properties.

Use of the Bitcoin blockchain

As it was mentioned in the overview section, in principle, any public ledge system similar to Bitcoin can be used for keeping ownership records. However, the use of specificatlly the Bitcoin blockchain has certain advantages:

- Reduced implementation cost: we only need a thin layer on top of Bitcoin, there is no need to implement a cryptocurrency from scratch.
- Security: By piggybacking on top of Bitcoin, we inherit its security properties. It is crucial for such a system to be tamper-proof, and Bitcoin is currently the cryptocurrency which is secured by the largest hashrate.
- Persistence: Signiificant total value of all bitcoins in existence create a huge financial incentive to keep the system alive and healthy.
- Integration with Bitcoin as a payment system: An ability to swap bitcoins and colored coins in one atomic transaction makes it possible to do trustless trades. E.g. if a car and a payment for a car are sent through one transaction, any problem with a transaction (e.g. a dounle-spend) will invalidate the trade. Thus it is impossible that a buyer will end up with a car when he haven't paid, just as it is impossible that seller will receive a money without transferring ownership of the car. Bitcoin is currently the largest cryptocurrency, which make integration with it interesting and useful.

However, the following sections of this document are not specific to the Bitcoin blockchain (and in some cases, not specific to colored coins) and can be relevant if a different cryptocurrency substrate is being used.

Property registry and catalog

When property transfers are secured by the blockchain, we no longer need to rely on a trusted party to verify them. However, an associated between a particular property and a genesis transaction output becomes the weakest link.

E.g. suppose somebody claims that a certain coin represents ownership of the house. He can demonstate that he is the owner of an unspent coin by signing a message using his private key, and he can demonstrate the transaction history involving that coin. But how can we check that a particular coin represents a particular house? How do we check that there are no other coins which represent it?

In the example with the car, the factory which manufactured the car was responsible for assotiating a colored coin with a car. A tag or a chip attached to a physical object might be used to refer to a genesis output, and thus establish an association. But this is reliable only as long as information contained in that tag or chip cannot be altered, and the cannot be detaching. (Or, rather, detaching them is impractical or prohibitively expensive.)

But we can't rely on tags in the case with a real estate, for example, thus we need some kind of a registry which will be responsible for association between objects and corresponding colored coins. Let's assume that for a kind of objects we are interested in, we can generate property identifiers which unambiguously point to an object (e.g. coordinates, street address, device identifier etc.). Then a registry will map genesis transaction outputs to property identifiers, and property identifiers to genesis transaction outputs.

Is it possible to make this registry distributed and trustless? It might work for some problem domains, e.g. in Namecoin, the first person who tries to register a name gets it. But this doesn't work in a general case.

Thus a registry needs to be a trusted third party. We can't completely escape from that model, however, we can try to minimize reliance on trust and impose rules which would make cheating hard, evident and provable.

Particulalry, trust is much less of a concern when the registry is forced to operate in a transparent way and cryptographic protocols are used to authenticate information supplied by the registry.

This can be accomplished by making registry's complete catalog openly accessible to everyone.

I.e. anyone can request a complete catalog from registry, which will reply with a list of (property identifier, genesis transaction output) pairs, with whole message being signed with registry's public key. This alone is enough to detect basic problems (e.g. duplicate or ambiguous identifiers) and attacks (if you have two messages with different association, you can detect that this registry is faulty and prove this to others).

But it doesn't prevent more sophisticated attacks, e.g. a registry might sent modified catalog only to a specific user, which won't be able to detect wrongdoing without an external point of refernce.

This is another problem which can be solved using the blockchain: when a complete catalog is published in the blockchain (and can be obtained, e.g. by scanning the whole blockchain for messages which are signed by a certain public key which is associated with a particular registry), everybody has the same view, and thus targetted attacks become impossible. It is also impossible to modify historic registation records, so there is only a brief time interval where wrongdoing is even possible.

(Note: Catalog can be seen as an append-only log of registration entries.)

Note that it is usually undersirable to put significant amount of information into the blockchain, in that case it's possble to publish a reference to catalog instead of a complete catalog, for the same effect. We will cover this in appendix.

Now let's go through a complete example. Suppose a certain registry is responsible for real estate registration in a certain geographic area. A person who wishes to register his property will come to this registry with all required documents which prove that he is the current owner. If the registry determines that provided information is correct, it will create a genesis transaction which will:

- contain unambiguous property identifier
- will be signed with the registry's public key
- genesis transaction output will point to current owner's Bitcoin address

Once this information is in the blockchain, the owner can transfer property without any further interaction with registry.

The only possible problem here is that a registry can send a colored coin to an address which doesn't belong to the owner. The owner can detect this by watching the blockchain for his property identifier. Dispute needs to be solved outside of the system (e.g. through litigation).

Transfer security

As it was noted above, cryptographic approach requires us to use public/private keys to identify and authenticate the current owner. However, in practice it is desirable to associate ownership with a specific person, as a private key can be stolen or lost.

This is a trade-off: we either need to rely on a trusted third party to authenticate owners and record transactions, or we need to rely on cryptography. Both approaches have advantages and disadvantages.

Note that it's possible to use the blockchain for record-keeping even if owner is authenticated using his name and documents: in that case registry's private & public keys will be used, and owner's name (as well as other relevant information) can be added to a transaction as meta-data. Use of the blockchain has the same benefits as described in the previous section: transfer history will be securely preserved in the blockchain. However, owners will have to rely on registry to do authentication properly.

A hybrid approach is also possible: a colored coin which represents property ownership will send to 2-of-2 multi-sig address, which requires signatures both from the registry and from the owner to unlock. In this case owner cannot transfer his property without interaction with registry, however, neither can registry do transactions without owner's consent. This sceheme can provide extra security: a registry can perform additional authentication steps to make sure that transfer is correctly authorized.

For extra transparency, details about the transfer can be embedded into the transaction, and thus preserved in the blockchain.

Ownership overrides

In an ideal world, we would all rely on cryptography and distributed consensus. But our world isn't ideal, thus we have to deal with the fact that ownership can be changed, for example, through litigation. A system which cannot address this issue can be impractical.

We believe that the best way to address this is to override association on the registry level, as we are relying on a trusted third party anyway. A registry should comply with court's orders to re-assign ownership.

If we assume that append-only log is used as a catalog (i.e. each entry is published in the blockchain), then a registry will need to publish another entry with same identifier and a flag that an old one is replaced. A buyer who is interested in the property will be able to detect the situation and pay closer attention.

Conclusion

A blockchain-based property ownership recording system described in this article eliminates most potential failures and attacks through transparency and use of cryptographic primitives for authentication. Thus it can be used to reduce reliance on trusted third parties, reduce costs (through automatization) and reduce number fraud and errors.

Appendix A: Technical implementation

Overview

In this appendix we will cover an implementaton of blockchain-bnased property ownership recording system using the Bitcoin blockchain and colored coins.

We need to take into account that the Bitcoin blockchain space is a scarce and valuable resource, thus it cannot be used for publishing arbitrary information. Instead of that, we will hash information we wish to publish, and embed those hashes into transactions. Information itself can be obtained from a party which generated it, i.e. the registry. This way we still get a consensus over what information was published, but only as long as registry is accessible and can provide information.

Headers-only clients

Ideally, we want clients to be able to verify information without the need to scan the whole blockchain, i.e. having only headers and relevant data. But we won't address this issue in detail, and instead will assume that client is able to scan the whole blockchain.

Registry

We assume that a registry is associated with a certain Bitcoin address or a set of addresses which it will use to publish property association transactions.

Each such transaction:

- contains an input spending coins from registry's Bitcoin address
- has data contained in OP_RETURN output (see below)
- has a genesis output which assigns colored coin to property owner

If epobc color kernel is being used, transaction must also be a valid epobc genesis transaction.

Data consists of a 'property association entry' tag and a hash of the registration entry.

In the most basic form registration entry is just a property identifier, however, it can also contain meta-data, such as a date of registation, a hash of a document which was provided during registation, a hash or registration request and so on.

A registry must provide an API which allows clients to fetch registation entries by their hashes.

Chain of entries

In order to make it possible for thin clients to verify that the complete catalog is downloaded, all property association transactions must be organized into a chain: each such transaction must have an in input linked to an output of previous such transaction. Conceptually, we can see it as a colored coin which is associated with the registry itself: it must be used in every assocation transaction. Having registry's chain genesis transaction and current UTXO, a thin client can obtain the whole chain.

Transfer transactions

In the most simple cases, something as simple as epobc color kernel can be used for transfer transactions. The advantage of using epobc is that it is well-tested and will be interoperable with other colored coins which are based on epobc. (E.g. it is possible to buy a colored coin-represented house using colored coin-represented gold using an atomic transaction.)

If it is desirable, it is possible to embded a hash of meta-data about the transfer in OP_RETURN output. This meta-data can include the name of new owner, date of transaction and so on.

epobc is compatible with multi-sig scripts, thus it isn't necessary to modify color kernel to enable multi-sig use.

Client software

Client software can be considered a normal colored coin wallet with extra functionality:

- It should be able to obtain an entire catalog of a specified registry, either by scanning the whole blockchain, or through the process which scans the chain.

- It should be able to represent property identifiers in a human-readable form, to provide a way to search for a specific entry, check for duplicates etc.
- Provide a way for an owner to authenticate himself, i.e. sign a message with a private key corresponding to property he owns.
- Provide a way to verify authentication: find a public key corresponding to a specific property and check message signature.
- If a special multi-signature is being used, client software must be able to create transactions with it.