July 10, 2018

STORING MESSAGE RECORDS IN A BLOCK CHAIN LEDGER FOR HIGH THROUGHPUT TRAFFIC

Lionel Florit

Justin Muller

STORING MESSAGE RECORDS IN A BLOCK CHAIN LEDGER FOR HIGH THROUGHPUT TRAFFIC

AUTHORS:
Lionel Florit
Justin Muller

ABSTRACT

Techniques are provided herein for recording every message transaction of a network inside a block chain in a scalable manner. The authenticity of a message may be verified independently without transferring any data during the verification process.

DETAILED DESCRIPTION

There have been proposals to leverage block chain technology to maintain a permanent record of received and transmitted messages in Internet of Things (IoT) or other domains. Block chain ledgers are not designed to accomplish a high level of transactions. Current technology does not permit maintaining a permanent record of every message that a router (or a set of routers) has (have) forwarded.

It is often desirable to maintain a record of every message (not packet) an IoT system produces. The transaction time of block chain technology has been well documented as one of its major limitations, and is measured in minutes. With some enhancements, the block chain infrastructure can return a transaction record within about a second. Today, it is not safe to assume that every sent IoT message has a unique record in a block chain.

Accordingly, the following setup is provided to achieve such a record. For each message received, the IoT edge routers may create a hash of the message and append it to the message, creating a tagged message. The tagged message is published simultaneously to three entities, which may reside further in the core of the network. The first entity is a historian (database) for permanent storage for the reference messages (if needed).

The second entity is a signature aggregation process whose function is to extract the received hashes from all messages of the IoT domain that the edge routers have received. The signature aggregation process creates a list, and either periodically or based on the size threshold computes a top hash for the entire list of hashes. This top hash is submitted to the

block chain (or any public ledger). The block chain records the transaction and return it to the signature aggregation process. The latter adds the transaction record to the hash list and stores that list in the historian.

The third entity is a process to remove the tag of the message, and forward the original message to its destination.

Figure 1 below illustrates an example sequence diagram for recording messages in a block chain at scale.

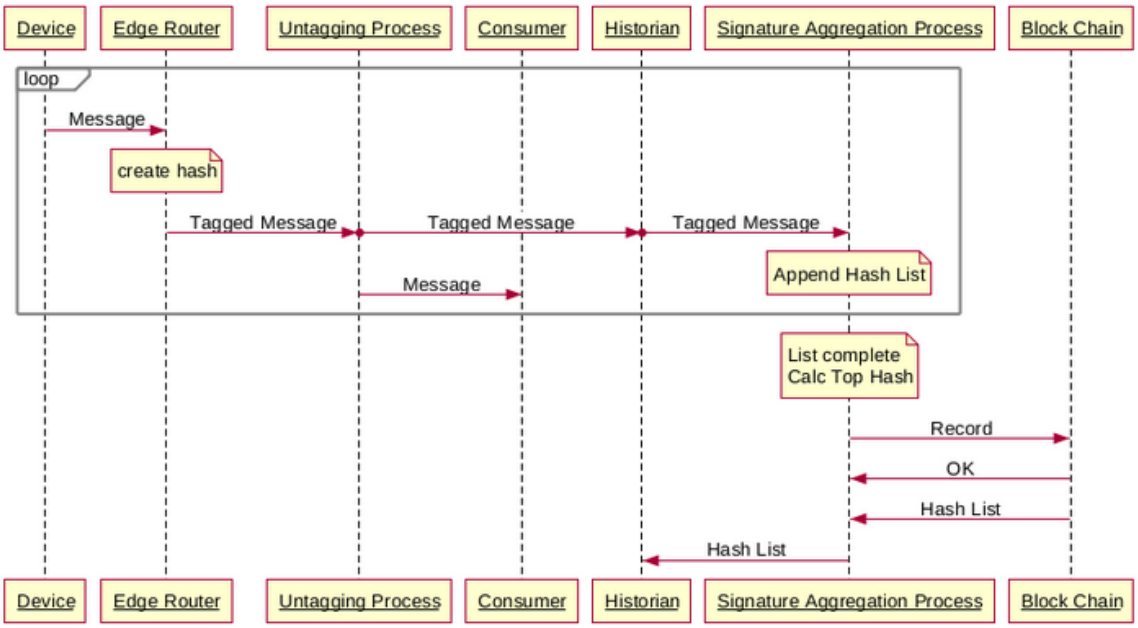

Figure 1

At a later time, a consumer of the message may wish to verify the authenticity of the message and access its metadata. The consumer may calculate the hash of the message it received from the third entity and provide it to the historian. The historian may return the associated list of the other hashes, and the consumer of the message may calculate the top hash and verify whether it has been recorded into the ledger. As an optional step, the historian may provide a copy of the original data.

This enables storing a record of every message transaction of a network inside a block chain. The authenticity of the message may be verified without mandating the owner of the message to provide the message itself (just the hash). Moreover, this permits storing and tracing back to a copy of the original message.

A hash of multiple hashes of IoT messages is thereby recorded. There is no impact on the devices, and this is a highly scalable process. Adding a ledger that the customer (or a third party) can manage independently provides additional assurance that the historian's records have not been tampered with. There is no need to place trust in an entity that manages the historian. Thus, a standard block chain system is leveraged to record every IoT message transaction, which is something that standard blockchain systems cannot do.

In summary, techniques are provided herein for recording every message transaction of a network inside a block chain in a scalable manner. The authenticity of a message may be verified independently without transferring any data during the verification process.