CORE Working Group
Internet Draft
Intended status: Experimental

P. Urien
Telecom Paris

December 15 2020

Expires: June 2021

Blockchain Transaction Protocol for Constraint Nodes
draft-urien-core-blockchain-transaction-protocol-05.txt

Abstract

The goal of the blockchain transaction protocol for constraint nodes is to enable the generation of blockchain transactions by constraint nodes, according to the following principles:
- transactions are triggered by Provisioning-Messages that include the needed blockchain parameters.
- binary encoded transactions are returned in Transaction-Messages, which include sensors/actuators data. Constraint nodes, associated with blockchain addresses, compute the transaction signature.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 2021.

.

Table of Contents

1 Overview

In the context of this draft sensors/actuators are powered by micro-controllers comprising about 10KB of RAM and 100KB of non volatile memory. The node electronic board may include a radio SoC (System On Chip) or the micro-controller can be part of the SoC. The radio chip manages IP connectivity with another device, typically acting as a controller, which provides a full internet access with standard computing resources.

A constraint node driving sensors and/or actuators may deliver critical data dealing with safety (fire detection,...) or legacy (pollution measurement,...) information.

Blockchain infrastructure provides two important features in an Internet of Things (IoT) context:

- Authentication of data in P2P context. Blockchain signed transactions are checked by numerous nodes.
- Information publication. Transactions are stored in duplicated and distributed databases.
- Dating information. Transactions are dated during the mining process.

The goal of the blockchain transaction protocol for constraint nodes is to enable the generation of blockchain transactions by constraint nodes, according to the following principles:
- transactions are triggered by controllers. Needed blockchain parameters are included in provisioning messages.
- binary encoded transaction messages are returned by constraint nodes. A node has the ability to compute the transaction signature.

2 Overview of the Blockchain Transaction Protocol for Constraint Nodes

2.1 Architecture

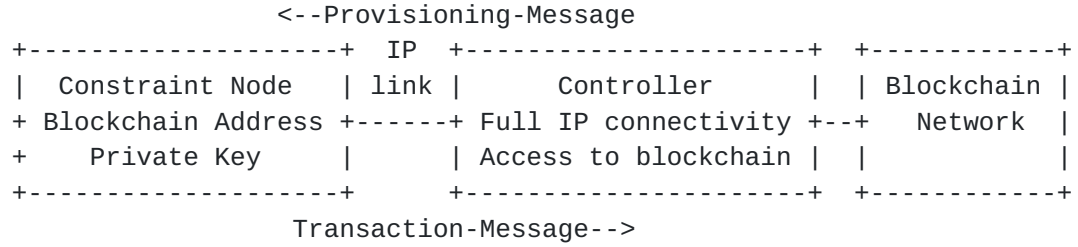

Figure 1. Functional architecture for the Blockchain Transaction Protocol for Constraint Nodes

A constraint node holds a blockchain address (BA). The blockchain address is computed from a private key (Pk). Most of today

blockchain infrastructures deal with ECDSA signatures, generated

over the Secp256k1 elliptic curve. The private key is a 32 bytes number, stored in the constraint node. The computation of hash procedures such a SHA2 or KECCAK-256 can be handled by microcontrollers. Although ECDSA signature may be generated by a microcontroller, a tamper resistant resource could be used, either embedded in the CPU, or in a chip such as a secure element[ISO7816]. As an illustration an architecture based on micro-controller, radio SoC and secure element was demonstrated in [IEEE-CCNC2018].

The controller is a device with full IP connectivity. It typically communicates with the constraint node thanks to the CoAP [RFC7252] protocol, or other legacy protocols such as HTTPS. The controller has access to the blockchain infrastructure, to which it is able to forward a binary encoded transaction, signed by the constraint node.

2.2 An Ethereum Use Case.

The following figure 2, illustrates an Ethereum transaction generated by a constraint node, whose total length is 118 bytes.

```
F8 74 // RLP List, length= 116 bytes
0C // nonce 1 byte =12 decimal
85 06FC23AC00 // gasPrice = 30 GWei
83 013880     // gasLimit = 80000 gas
// recipient address 20 bytes
94 6BAC1B75185D9051AF740AB909F81C71BBB221A6
80 // Null Ether Value
// Data 15 bytes "Temperature=25C"
8F 54656D706572617475726 53D323543
1B // recovery parameter, 1 byte
A0 // r, 32 bytes, ECDSA r parameter
A9B58980F76EE6284800B82A2B5DF13E456887EC0CF426A5E5D6A738EB1784ED
A0 // s, 32 bytes, ECDSA s parameter
629633C6A3ED5FEE0FB40E2D1CF251345B885D372857B1A6C4762C9BE914281F
```

Figure 2. Illustration of an Ethereum transaction, generated by a constraint node.

The identifier (TxId) of this transaction (i.e. its KECCAK-256 digest) is:

0xd6904d832462ae17718c69e9caa0c3f3bed458382ac1f4e43b1aadd8e94744ad

Given this TxId, the transaction can be retrieved in any Ethereum blockchain database, like for example:

https://etherscan.io/tx/0xd6904d832462ae17718c69e9caa0c3f3bed458382a
c1f4e43b1aadd8e94744ad

The transaction date (20-2018 09:52:42 PM +UTC) is published and

certified by the blockchain.

The binary encoded transaction comprises two parts,
- information relying on the Ethereum blockchain context, such as the nonce, the gasPrice, the gasLimit, the recipient address, and an Ether value.
- information delivered by the constraint node, data (a temperature measurement), and the ECDSA signature computed from the 32 bytes private key.

Parameters relying on the Ethereum blockchain context MUST be included in the Provisioning-Message.
The signed transaction MUST be included in the Transaction-Message.

3 Blockchain Transaction Protocol Messages Definition

The Blockchain Transaction Protocol comprises two messages, to be included in transport protocols, such as CoAP or HTTP.

3.1 Provisioning Message

This message includes the following attributes :
- A type, an integer value, specifying the message content.
- An ordered list of values, storing the parameters of the blockchain context.

3.1.1 Encoding example in JSON syntax

Here is an illustration of the provisioning message associated to the Ethereum blockchain.

```
{
  "type": 1,
  "nonce": 12,
  "gasPrice": 30,
  "gasLimit": 80000,
  "address": "6BAC1B75185D9051AF740AB909F81C71BBB221A6",
  "value": 0
}
```

3.2 Transaction Message

This message include the following attributes
- A type, an integer value, specifying the message content. The zero value indicates an error.
- The binary encoded transaction, including the signature.

3.2.1 Encoding example in JSON syntax

Here is an illustration of the transaction message associated to the Ethereum blockchain.

{
"type": 1,
"transaction":
"F8740C8506FC23AC0083013880946BAC1B75185D9051AF740AB909F81C71BBB221A
6808F54656D70657261747572653D3235431BA0A9B58980F76EE6284800B82A2B5DF
13E456887EC0CF426A5E5D6A738EB1784EDA0629633C6A3ED5FEE0FB40E2D1CF2513
45B885D372857B1A6C4762C9BE914281F"
}

4. Blockchain Transaction Protocol Messages Binary Encoding

4.1 CoAP messages

To be Done

4.2 HTTP Messages

To be Done

5 IANA Considerations

TODO

6 Security Considerations

TODO

6 References

6.1 Normative References

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, June 2014.

[ISO7816] ISO 7816, "Cards Identification - Integrated Circuit Cards with Contacts", The International Organization for Standardization (ISO).

6.2 Informative References

[IEEE-CCNC2018] Urien,P., "An Innovative Security Architecture for Low Cost Low Power IoT Devices Based on Secure Elements", IEEE CCNC 2018

7 Authors' Addresses

Pascal Urien
Telecom Paris
19 place Marguerite Perey
91120 Palaiseau Phone: NA

France Email: Pascal.Urien@telecom-paris.fr