Blockchain to Improve Security and Knowledge in Inter-Agent Communication and Collaboration over a Restrict Domains of the Internet Infrastructure

Juliao Braga\textsuperscript{1,2}, Joao Nuno Silva\textsuperscript{2}, Patricia Takako Endo\textsuperscript{3,4}, Jessica Ribas\textsuperscript{1}, Nizam Omar\textsuperscript{1}

\textsuperscript{1}Universidade Presbiteriana Mackenzie (UPM)
\textsuperscript{2}IST - INESC ID, University of Lisboa, Portugal
\textsuperscript{3}Universidade de Pernambuco (UPE), Brazil
\textsuperscript{4}Dublin City University (DCU), Ireland

\{juliao.braga,joao.n.silva\}@tecnico.ulisboa.pt, patricia.endo@upe.br
jessica.ribas@mackenzista.com.br, nizam.omar@mackenzie.br

Abstract. This paper describes the deployment and implementation of a blockchain to improve the security, knowledge and intelligence during the inter-agent communication and collaboration processes in restrict domains of the Internet Infrastructure. It is a work that proposes the application of a blockchain, platform independent, on a particular model of agents, but that can be used in similar proposals, once the results on the specific model were satisfactory.

Resumo. Este documento descreve o desenvolvimento e implementação de uma blockchain para melhorar a segurança, o conhecimento e a inteligência durante os processos de comunicação e colaboração entre agentes em domínios restritos da Infraestrutura da Internet. É um trabalho que propõe a aplicação de uma "blockchain", independente de plataforma, em um modelo particular de agentes, mas que pode ser usado em propostas similares, uma vez que os resultados no modelo específico foram satisfatórios.

1. Introduction

Autonomous System (AS) is the name given to the networks making up the Internet [Hawkinson and Bates 1996]. ASes establish interconnections through a protocol called Border Gateway Protocol (BGP) [Rekhter et al. 2006]. BGP is a complex protocol that requires a lot of knowledge from the administrators of an AS. Sometimes the human being also forgets to update information, especially those related to routing policy and that reside on important servers such as Internet Routing Registry\textsuperscript{1} (IRR), for example. IRR is a distributed database of route and route-related information [Braga 2010]. The sometimes neglected participation of the human being during the creation and update IRR objects processes, was the motivation for creating a model of agents which could replace the human interventions (made by

\textsuperscript{1}http://www.irr.net/
email). For this reason, we propose the *Autonomous Architecture Over Restricted Domains* (A2RD) into the restricted domain of an AS, applying as use case over the IRR [Braga et al. 2015]. A2RD replaces the human with your agents, *Intelligent Elements* (IEs), establishing a new IRR model, named *innovation IRR* (iIRR), shown in Figure 1.

A2RD specialized IEs, automatically create objects as defined by the *Route Policy Specification Language* [Alaettinoglu et al. 1999, Blunk et al. 2005]. Those objects that cannot be created automatically will receive support from AS administrators through a human-computer cooperation mechanism. Nothing is changed in relation to the present and future IRR structure, characterized by the expectations recommended by the stakeholders to the *Internet Engineering Task Force* (IETF) and *Internet Research Task Force* (IRTF) disseminated through their formal documents [Meyer et al. 1999, Villamizar et al. 1999, Newton 2004, McPherson et al. 2015, Kisteleki and Haberman 2016]. Neither does it affect the security concerns surrounding the IRR and Internet governance [Kuerbis and Mueller 2017]. Similarly, tools that use IRR databases can be used without any modification. A very useful, among others, is the IRR Powertools⁴.

For this paper, *blockchain* is a data structure whose components are chained, with guarantees of immutability of its contents, and consequent integrity of the chain preserved by a cryptography process, with difficult computational reversibility. This definition is much simpler but more computationally oriented than those in which blockchain is associated with crypto-economics or crypto-currencies, and often have confusing definitions, but when it is clear, blockchain is defined as a database [Nakamoto 2008, Pilkington 2015].

On the other hand, by abstracting from the property of immutability, the data structure like blockchain is a well-known concept used in computer research and originated in the academic literature of the 1980s and 1990s [Narayanan and Clark 2017]. As

---

2https://ietf.org/
3https://irtf.org/
4https://github.com/6connect/irrpt
a simple data structure, for example, in works involving provenance, which is used as complementary data documentation containing the description of 'how', 'when', 'where', 'why' the data were obtained and 'who' obtained it [Braga and Banon 2008]. The blockchain model proposed in 2008 to meet the Bitcoin virtual currency has effectively aroused the interest of the research community mainly by the immutability property that ensures data integrity [Prusty 2017, Bashir 2017]. Immutability and integrity are obtained by a hash encryption mechanism [Bakhtiari et al. 1995, Rogaway and Shrimpton 2004]. The combination of these two factors and characteristics associated to the blockchain recommended the application in the A2RD model, with the aim of enhancing communication and collaboration among the IEs [Braga et al. 2017b]. This proposal is more simpler than those application of blockchain in Internet Infrastructure with fundamentals in Bitcoin technology, based in the appropriate fact that to run, Internet use resources such as numbers and names [Hari and Lakshman 2016].

There is no study directly related to this work and there are few blockchain works related to the Internet Infrastructure. Blockchain still is not a matured technology, there are challenges that need to be considered when designing a platform, to ensure security, reliability and usability. So, there is not related works associated with Internet Infrastructure, because is fact that due to the emergent nature of the topic, the reviewed literature was not published in high-ranking journals with prolonged review cycles [Xu et al. 2016].

The main goal of this paper is to present the Internet Infrastructure Blockchain (IIBlockchain), a blockchain architecture to improve the security, knowledge and intelligence in inter-agent communication and collaboration over a restrict domains of the Internet Infrastructure, developed specifically and therefore independent of any available blockchain platform. The next sections of this paper will be organized as follows. In section 2 we discuss the A2RD model and the needs for inter-agents communication and cooperation. In section 3 we present the architecture of IIBlockchain and the properties inherent to the blocks, their types and the characteristics of the designed chain. In section 4 we discuss the implementation of IIBlockchain showing the main associated properties. In section 5 we present the conclusions and in section 6 we present the proposals for future works.

2. The A2RD Model

A2RD is a project that initially proposed the creation of agents with automatic activities replacing human tasks in the environment restricted to the domain of an AS. The use case was the addition and update of objects in IRR server. The application was considered useful mainly because the tasks of the AS administrator did not guarantee the accuracy in its completion nor the permanent need to update the objects making the IRR an unreliable system from the point of view of its contents. A2RD solved this problem.

A new proposal for the A2RD model emerged from this experience [Braga et al. 2017b]. The Figure 2 shows this new proposal, in which the A2RD, (1), is represented as an agglomeration of IEs in a four layers model.
A2RD IEs, reach their autonomy and intelligence aided by three components, the Knowledge Base, (2), the Training Data Sets, (3), and wordIETF, (4). These three components are obtained from non-structured databases, in particular, from the Request for Comments database, containing documents authored by network operators, engineers and computer scientists, documentary methods, behaviors, research, or innovations applicable to the Internet, all of them, working in groups of the IETF and IRTF, and maintained by RFC-Editor\(^5\).

Each AS, of its own free will, may implement its respective A2RD, which is controlled by the IE named IE Controller, which receives the identification \(x:0\), where \(x\) is the AS Number (ASN).

IEs need to communicate in order to collaborate, learn and cooperate with each other. This communication needs to be secure, that is, the respective IE controller must recognize the origin of each pair in their information exchanges. A mechanism called Dark Think Security (DTS) has been proposed to ensure the desired security [Braga et al. 2017a]. Although preliminary implementations have revealed that DTS is indeed secure, it has proved to be complex in implementation. In the search for a simpler alternative included Pretty Good Privacy (PGP) [Garfinkel 1995]. Using PGP, an AS\(_x\) IE controller that wants to communicate with an AS\(_y\) IE controller, will use the AS\(_y\) public key to encrypt the message, for \(\forall x \text{ and } \forall y \text{ such that } x \neq y \text{ and } x, y = 1, \ldots, n, n \leq \text{total ASes present in the Internet Routing Table}\(^6\). The AS\(_y\) controller uses your secret key to decrypt the message. Thus, for this and for other reasons that we will see in the following section, the recommended solution was a variation of blockchain implementations proposed in the literature, that we named in this paper as IIBlockchain.

\(^5\)https://www.rfc-editor.org/
\(^6\)http://thyme.rand.apnic.net/current/data-summary
3. IIBlockchain Model and Implementation

The IIBlockchain model can be seen in Figure 3, which shows the implementation of A2RD in any two ASes (AS\textsubscript{x} and AS\textsubscript{y}).

![IIBlockchain Architecture implemented over ASx and ASy domains](image)

This figure shows that the respective A2RD communicate through encrypted messages. Also, the A2RDs independently maintain a blockchain with properties characteristic of IIBlockchain. These chains contain, in their blocks, data inherent to each A2RD and about the environment of the AS in which they are implemented allowing the cooperation through the exchange of knowledge and information that can help in learning and maintaining the autonomy of their respective IEs. Each A2RD locally maintains a copy of IIBlockchain from each of the other ASes. There is no need to implement an A2RD for a chain to be constructed for an ASN. Specialized IEs of an AS\textsubscript{x} any guarantee that minimal information is included in chains of other ASes.

3.1. Block Properties

A block of any chain type is equivalent to a dictionary structure of the Python language, whose configuration and summary description of the respective keys are shown in Figure 4.

The detail description of block keys are in Table 1.

3.2. Chain Properties

Any chain only exists if it has a 'Genesis' block type as its first block ('block_seq' = 1). Suppose that AS\textsubscript{x} wants to add in its chain, a block that will contain its PGP public key with which any ASN can encrypt messages that only AS\textsubscript{x} will understand. At this point, the AS\textsubscript{x} chain is empty. Suppose $x = 18782$. So, using the IIBlockchain
Python class available at GitHub\textsuperscript{7} if we add the block of type *PublicKey* we will have a two block chain as can be seen in Figure 5. It is important to note that block numbers (\textit{block_seq}) are sequential (1 and 2, respectively).

Continuing and add a third block, now an \textit{mntner} IRR object type (\textit{irr-mntner}). The data is transformed into a string to be signed by the PGP, ensuring data properties to AS18782. Once this is done, the block is added to the chain as the third block. The block added can be seen in Figure 6.

To complete these example that illustrate some properties of the chain, let’s assume

\begin{verbatim}
{  'asn': 18782,  'block_seq': 1,  'obsolete_block_id': '',  'timestamp': 'Wed,31 Jan 2018 10:57:19 +0000',  'block_type': 'Genesis',  'block_id': 'b063e676ebaf96432e66ad45018680a9313345b67490b6c49a0b791d8791d03932a',  'previous_block_id': '0',  'data': '',  'signature': '' }

{  'asn': 18782,  'block_seq': 2,  'obsolete_block_id': '',  'timestamp': 'Wed,31 Jan 2018 10:57:19 +0000',  'block_type': 'PublicKey',  'block_id': 'c18cb625f918b75bab292c2e2a94f933d55d9207174573bd0868464c76735857',  'previous_block_id': 'b063e676ebaf96432e66ad45018680a9313345b67490b6c49a0b791d8791d03932a',  'data': '',  'signature': '' }

[-----BEGIN PGP PUBLIC KEY BLOCK-----]
Version: GnuPG v1.4.7

mE0Wn5hDxESAkyp9WkKCLJ3PJg5z7F0GrpQC35u12JQJ6sMv4yJ9A5v0fNvkrq1L,
\"W8oATQMDu0zPr+z6Y\fKCDs9d8U1bJ+udqC5BNWpgraRmgmB5E7z2TvC\"oT,
\"QLTQaiyqEBSBdHxkDyvK7YyZllkxOlS9qQsomsamSBYq3vug5IqAg7A9AEBB9AG0\",
\"LAF7qHJdD5g7Kv2ZQg3Wq4e2TJSRCB6jRbCkPfF7q35JAYT7u2ZC5dKZrI\",
\"uAQTAgigjWv5hIvJL6ibLwL7Cq4hWh5kQg7QcCQYBqYxEWQcFC4AACgkq8P\
\"HGAVa0sR9qGnlR4XReaAzUzqVki8FvDz9RP1Z+73BABGNGXx956ML5xQ7wB9p,
\"YREljGnJk4nBbk1cRcTRBS8XWWXWx6qZCN2N5sN6CvDFOYx3p5R5KwLWW\",
\"0PW2VkhdePN+PjJnOrTrqX4Sm9IALCv4bURiBdRZwicz49v\",
\"=zMLP4\",
\"-----END PGP PUBLIC KEY BLOCK-----\"
}

[\{]
\"signature\"
\}

\end{verbatim}

\textsuperscript{7}https://github.com/juliaobraga/a2rd/IIBlockchainCode
Table 1. Description of block dictionary keys

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>asn</td>
<td>ASNumber: Identifies the owner number of the string. For same string, the value of this key is always the same</td>
</tr>
<tr>
<td>block_seq</td>
<td>Identifies the position of the block within the chain. If block $i$ preceding or immediately preceding block $j$ than $i &lt; j$ and not necessarily $j = i + 1$. This is due to the fact that a block can be removed, from an ASN chain, if it becomes obsolete. Upon removal, the block is added to the obsolete chain. The immutability and integrity of this ASN chain must be restored.</td>
</tr>
<tr>
<td>obsoletes_block_id</td>
<td>If the value of this key is not empty, so this references the block_id that will be obsoletes</td>
</tr>
<tr>
<td>timestamp</td>
<td>Time moment the block was add in the chain</td>
</tr>
<tr>
<td>block_type</td>
<td>Type of the block: block types are not necessarily predefined. IEs can create different types of blocks through agreements between them during their normal activities. Important blocks are, however, predefined. For example, the Genesis block, which is necessarily the first block of any chain. Blocks that represent IRR objects always prefix the usual object name with irr_.</td>
</tr>
<tr>
<td>block_id</td>
<td>Hash that will identify the block, obtained on the whole block, after it is completely filled</td>
</tr>
<tr>
<td>previous_block_id</td>
<td>block_id of the previous block of this block</td>
</tr>
<tr>
<td>data</td>
<td>Data of the related with the block type</td>
</tr>
<tr>
<td>signature</td>
<td>Signature that ensures the owner of the data</td>
</tr>
</tbody>
</table>

A change in the object irr_mntner. A new data is signed via PGP, and included in the chain, not without first identifying in the obsoletes_block_id, the block that it is rendering obsolete. The new block is added as 4th block in the AS18782 IIBlockchain and your configuration is shown in Figure 7.

3.3. Chain Transfer

The chains are compressed and named as ASxVaaaammmddhhmmss.zip. A specialized IE will take care of this activity and follow up by compacting the chain, sending it to GitHub and update the respective version in wordIETF. All chains are public, but the secret keys are not.

4. IIBlockchain Implementation

In this section we make considerations on important topics that deserved special attention during implementation.

https://github.com/juliaobraga/a2rd
4.1. Space Analysis

Table 2 displays some data about storage values, considering the chain created for the example in this paper.

<table>
<thead>
<tr>
<th>#</th>
<th>Discrimination</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Block 1</td>
<td>1,300</td>
</tr>
<tr>
<td>2</td>
<td>Block 2</td>
<td>1,365</td>
</tr>
<tr>
<td>3</td>
<td>Block 3</td>
<td>3,451</td>
</tr>
<tr>
<td>4</td>
<td>Block 4</td>
<td>3,571</td>
</tr>
<tr>
<td>5</td>
<td>Total</td>
<td>9,687</td>
</tr>
<tr>
<td>6</td>
<td>ASes in routing table (12 Feb 2018)</td>
<td>59,789</td>
</tr>
<tr>
<td>7</td>
<td>IRR objects number (ARIN)</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>Number of protocols in TCP/IP</td>
<td>51</td>
</tr>
</tbody>
</table>

We used the `sys.getsizeof` function to determine the amount of bytes occupied by the Python dictionary structure, chosen to represent IIBlockchain. The result is not very good and so we evaluated two alternatives versions. The preferred version was that
of larger result values\(^9\) (lines 1-4 on the table). Suppose each block of the string to be constructed occupies twice as many bytes as the largest block in our example (line 4). So our block occupies \(7,124\) bytes. American Registry for Internet Numbers\(^{10}\) (ARIN) identifies ten objects to populate its IRR (line 7). Thus, only with IRR objects, the IIBlockchain of an AS spends \(7,124 \times 10 = 71,124\) bytes ~ 70 Kbytes. So, the total bytes to represent the IRR objects for all ASn are: \(59,789 \times 70\) Kbytes = 4,285,675,520 ~ 4 Gbytes. Let us now assume that for each TCP/IP protocol\(^{11}\) (line 8) we will need 20 blocks with the largest known double size (knowledge in-formation, for example): \(20 \times 7,124\) bytes = 139 Kbytes, value that corresponds to 0.003% of the space spent by IRR objects. Certainly there are other types of blocks that IEs will produce. But the largest number of them are obsolete blocks. Very difficult to measure the space to be occupied by obsolete blocks. Only an in-accurate estimate would be possible. One estimate is that 25% of the blocks will be obsolete. So the total estimated storage space for the IIBlockchain is 5 Gbytes. Any operation on IIBlockchain do not require additional space. Therefore the space complexity is \(O(1) \sim O(n)\) [Costa 2015].

---

\(^9\)https://goshippo.com/blog/measure-real-size-any-python-object/
\(^10\)https://www.arin.net/resources/routing/templates.html
\(^11\)http://www.comptechdoc.org/independent/networking/protocol/protnet.html
4.2. Time Complexity

The heaviest algorithm we have in operations on IIBlockchain is to search linearly over an array or eventually over a linked list. Then, in the worst case, the complexity of time is $O(n)$ [Costa 2015].

4.3. Security

IIBlockchain is public. The security that matters to IIBlockchain will only be verified when a non obsolete block needs to be used. In two stages this is necessary: (a) the integrity of the block and (b) the reliability of the information contained in the block. Stage (a) consists of checking the validity of the hash that identifies the block $id$. Stage (b) is the verification that the signature guarantees ownership of the information by the respective AS. If any of the above stages fails, an alert is sent to all implementations of A2RD. Immediately look for the block in the previous version and use it. The existence of the block in the previous version can be verified by the parameter $timestamp$ and the name of the version. Meanwhile, specialized IEs will analyze the chain, in order to identify the cause of the breach of trust in the block.

5. Conclusions

The authors consider that the objective of allowing a mechanism of relationship between IEs of the various A2RD implementations was achieved. Also, Blockchain is effective in ensuring co-operation and distribution of knowledge that can be shared among IEs in the various domains of ASes. It is a simple, easy-to-understand, and implementation-oriented design with no additional effort required in any programming language. The IIBlockchain has both public and private characteristics and has no inherent concerns or additional difficulties, for this reason. Also, it is worth remembering that IIBlockchain is oriented to the application of Blockchain by agents and not by humans, which certainly decreases complexity.

6. Future Works

At this point, it is not possible to determine how the presence of obsolete blocks will influence the operations on an IIBlockchain of some ASN. Implementations in programming languages like Python and others one, does not seem to be a big problem, because dictionaries are indexed and obsolete blocks can be ignored. However, it is necessary to evaluate the possibility of creating a new type of chain: the chain of obsolete blocks, that is to say, the chain consisting of blocks that become obsolete in each ASN chain.

At some point, one A2RD IE may checking the state of the chain and remove obsolete blocks, passing it to the obsolete chain considering:

- The chain from which the block was removed will be reconstituted to maintain the immutability and integrity. This is achieved by having the next block point to the previous block removed, and a new hash is calculated to identify the next block and successively to the blocks thereof until the end of the chain.
The block removed will be inserted in the obsolete chain pointed to the last block of this chain. The block’s block number (‘block_seq’) should be concatenated by a hyphen and another sequence number to the number of the last block of the obsolete chain. After this a new hash will be determined to identify this block and the block can be inserted in the obsolete chain.

Complementary, the IIIBlockchain design is simple enough for applications in several other networking areas or not. New versions of the implementation will seek to establish independence from block structure and coding.

7. Thanks

From Juliao Braga and Jessica Ribas: Supported by CAPES – Brazilian Federal Agency for Support and Evaluation of Graduate Education within the Brazil’s Ministry of Education.

References


