ABC4.IO

Executive Summary

Peter Waher

Waher Data AB, Kajutvägen 26, 139 55 Värmdö, Sweden
ABC4.IO Chile SpA, Abedules 1111, Concón, Chile.
peterwaher@hotmail.com

Abstract. This paper describes the benefits of defining interoperable APIs using Smart Contracts, and how this simplifies the creation of secure and transparent interoperability interfaces across domains. It also shows how this method can be used to create secure open interfaces to new and legacy systems and databases. ABC4.IO defines semantics that allow for the definition of decentralized communication APIs using multiple protocols, between multiple nodes. Embedding such interfaces in smart contracts make it possible to tie APIs to legal agreements (for example Data Protection Agreements including purpose definitions), manage versions and life cycles in a transparent and auditable manner.

Keywords: Decentralization, Interoperability, Security, Transparency, IEEE P1451.99.

1 Introduction

It is a common problem in modern, as well as legacy systems, that it is difficult to know what code can access sensitive data in databases, and why. This is especially true in larger processing sites with many independent teams and central data lakes. It is often the case that services are comprised of opaque, bespoke implementations, with hard-coded data access, lacking any efficient means of oversight from auditors or security personnel. Often, the only resources available are declarations made by developers and architects, which may be in different states of concordance with reality, especially over time, as systems are updated.

It is also a time-consuming process to develop and maintain secure interfaces. Decentralized architectures, furthermore, need to implement multiple interfaces for all the different types of nodes involved in the decentralized process. Getting an overview of such implementations is often very difficult, and often change more quickly, than an auditor or architect can spend mapping the architecture.

To have a clear understanding what authorizations should be provided to different services in decentralized architectures is also a challenging problem. To solve that problem, the purpose of processing must be clearly understood and agreed upon. Such

legal artefacts of decentralized architecture is most often handled manually, with poor oversight, especially as systems evolve over time.

A simpler method to create transparent, easily overviewed and interoperable interfaces tied to legal agreements between actors across decentralized domains is needed.

2 ABC4.IO

ABC4.IO (or API-Builder Contracts for Inter-Operability) is a service that creates APIs based on semantics in smart contracts. A Smart Contract between nodes in a network becomes valid as soon as all pre-defined nodes have signed the contract. As soon as the contract becomes valid, the APIs, according to the contract, are simultaneously published on the nodes. When the contract is ended (either through a contractual time-period, or through revocation, if such is allowed), the corresponding APIs are taken down simultaneously on all nodes. A single artefact, the smart contract, defines all API endpoints, on all nodes that are parts in the contract, using the compact definitions made available through the semantics defined by ABC4.IO.

Transparency

By defining all API endpoints in a single XML document (which is readable by humans and machines), maximizes the transparency of *what* data is processed. By embedding such a document in a smart contract, tying it to a legal agreement, possibly a Data Protection Agreement, *why* the data is being processed also becomes transparent. A combination of who signs the contract, and definitions in the XML document, also makes it transparent *who* processed the data. The encompassing contract also defines the life cycle of the processing, the *when*. A single artefact defines the *what*, *why*, *who* and *when* data is being processed. All cryptographically signed, for integrity. Processing is not only transparent, but also auditable.

Versions

Handling versions, and the problem of mistakenly breaking backwards compatibility, is a big problem in distributed applications, as changes on one node, made in one domain, often cause negative side-effects on other nodes, in other domains. ABC4.IO solves this, by requiring updates of APIs to be done by updating the ABC4.IO definitions and creating new smart contracts. Such new smart contracts exist in parallel with older smart contracts. As API definitions are immutable during the lifetime of the smart contract, APIs are forced to be backwards compatible when being updated. Old endpoints are always retained, as long as old contracts are still valid. Obsolete versions are automatically removed from nodes, as soon as contracts become obsolete. (Contracts become obsolete when they expire; revokable contracts can also be manually revoked.)

Inventory

By defining APIs using transparent ABC4.IO smart contracts, it is also easy to create an inventory of contracts, and their corresponding APIs, and versions, in decentralized processing. This inventory is updated in real-time, by default. This is a powerful tool, especially when it comes to generating and maintaining a registry of records of processing activities, as required by privacy legislation, such as the GDPR[1]. This is a notoriously difficult task, in the dynamic environment of Smart Cities, and Industry 4.0 installations, for example.

Harmonization

Different organizations may process similar data, but in different formats or structures. When such organizations want to open interfaces for 3rd parties to use, the problem of interoperability across indirect domains become important. When domains interact directly, interoperability is a natural part of designing interfaces, as there is a self-interest to do so. But when the only thing domains share, is a common consumer, interoperability across domains is not always cost-effective or of interest, and sometimes apparently counter-intuitive. Legal obligation may force organization to adapt to common standards in some cases, but the problem is much bigger and more general.

APIs designed using ABC4.IO simplify the harmonization across domains, including domains that cooperate only indirectly through 3rd parties. Definitions in contracts can be easily reused and modified to match the corresponding back-end data store. By reusing definitions in smart contracts in this way, harmonized APIs are generated, looking the same from a 3rd party consumer perspective, but adapted to the underlying data store in each case.

Contract-based interfaces

ABC4.IO API definitions are not only defined in smart contracts, but they also allow for the simple creation and use of smart contracts themselves. This allows the APIs to create smart contracts of any type on demand. As contracts are transparent and informative, and can have an expected lifetime, or be revokable on demand, such contracts form an excellent basis for modelling consent agreements, as well as other forms of agreements. This makes ABC4.IO an excellent tool for creating APIs that provide access to data based on authorization by consent by the data subject, or through contractual obligation, as defined in the GDPR[2].

---

[1] GDPR, Article 30.
[2] GDPR Articles 6 & 7.

Software-Defined Networking

Using ABC4.IO it is possible to create advanced network topologies for processing of data, suitable for different usage and security requirements. Using the XMPP protocol, it is possible to securely tunnel requests from a node public on the Internet, to a node in an internal protected local area network, for instance. Such an internal node can then be given controlled access to a data store, without exposing the data store to entities on the Internet. Communication between public nodes can also be done using XMPP, or HTTPS and mutual TLS, for example. The architect defining the interfaces can choose what network topology and security infrastructure to use, to best comply with existing requirements and regulations.

3 IEEE P1451.99 IoT Harmonization

To maximize the interoperability and reach of ABC4.IO, all interfaces[3] used for smart contracts and digital identities are defined in the proposed standard IEEE P1451.99 IoT Harmonization[4], whose goal is to create an interoperable, open, secure and globally scalable real-time communication infrastructure for smart cities and smart societies in general. This includes Industry 4.0, Open Banking, Smart Agriculture, etc., as specific use cases. This also makes it possible to use the TAG Digital ID app[5] for Android and iOS as a tool for signing, reviewing and managing digital identities and the corresponding smart contracts.

4 Run-time environment

ABC4.IO can run in any of the following runtime-environments:

- As a service running on the TAG Neuron™ and TAG Neuro-Ledger®[6], publicly on the Internet. This is the best option for running ABC4.IO APIs that need to be accessed from the Internet. Neurons form a federated global network, which simplifies interoperability and global scalability. Access to the Neuro-Ledger® permits ABC4.IO interfaces to write entries and interfaces that interact with a Distributed Ledger that complies with requirements stipulated in the GDPR in a way a block-chain based Ledger does not[7].

[3] https://gitlab.com/IEEE-SA/XMPPI/IoT
[4] https://standards.ieee.org/project/1451-99.html
[5] https://github.com/Trust-Anchor-Group/IdApp
[6] See Neuro-Ledger™, Executive Summary, p. 4, 2019-10-11.
[7] European Parliamentary Research Service (EPRS), Panel for the Future of Science and Technology, “Blockchain and the General Data Protection Regulation – Can distributed ledgers be squared with European data protection law?”, PE634.445 – July 2019.

- As a service running on LILS.IS Little Sister™ information collaboration software. This is the best option of services running in protected environments and local area networks internally, or by teams. LILS.IS Little Sister™ adds tools for information collaboration (posting, messaging, publishing), as well as file sharing and folder synchronization between humans in a protected and end-to-end encrypted manner across domains. Together with ABC4.IO it is possible to create both H2H, H2M and M2M interfaces.
- As a service running on the IoT Gateway[8]. This makes it possible to run ABC4.IO on any machine or device (such as the Raspberry Pi), supporting .NET Core (Windows, Linux, MAC, etc.).

5 Summary

ABC4.IO is a service that simplifies the creation of secure interoperable interfaces between machines across domains. It can be run on multiple hosting environments, depending on requirements, and support multiple protocols. A single document defines the interfaces involved on all participating nodes, which increases transparency and harmonization. Using smart contracts to define APIs further increases transparency, simplifying auditing and GDPR records keeping, as well as forcing interfaces to be backwards compatible as interfaces are immutable for the lifespan of the corresponding contracts. It also makes it much easier to implement processes complying with end-user consent or contractual obligation regulations.

[8] https://github.com/PeterWaher/IoTGateway