A Distributed Infrastructure for Democratic Cloud Federations

Andrea Margheri, Md Sadek Ferdous, Mu Yang, Vladimiro Sassone
University of Southampton
{*a.margheri;s.ferdous;mu.yang;vsassone*}*@soton.ac.uk*

***Abstract*—Cloud federation is a novel concept that has been drawing attention from research and industry. However, there is a lack of solid proposal that can be widely adopted in practice to guarantee adequate governance of federations, especially in the Public Sector contexts due to legal requirements.**

In this paper, we propose an innovative governance approach that ensures distributed and democratic control in cloud federations. Starting from FaaS, a recent cloud federation proposal, we propose a blockchain infrastructure for the federation registry that implements the proposed governance approach.

***Keywords*-Cloud Federation, Governance, Blockchain, Registry, Privacy.**

I. Introduction

The increasing proliferation of cloud systems raises new issues, particularly on the interconnectivity and cooperation of already deployed clouds. Organisations are looking for appropriate solutions to create a flexible aggregation of cloud systems, which can be formed dynamically by different individual clouds that cooperate to achieve specific business goals. Indeed, the underlying motivations can be multiple. Among these, sharing of computing resources, controlled usage of third-party services or data, collaboration among entities belonging to different administrative domains. Hence, cloud aggregation leads the constituent clouds to achieve goals that were not possible to achieve otherwise.

A prominent proposal for cloud aggregation is *cloud federation* [1], [2], [3]. It is a recent concept that allows services from different cloud providers to be aggregated in a single pool. However, there is no widely accepted proposal that organisations can adopt, and the few available lacks of adequate governance solutions: all federation members should be a network of peers equally concurring to their governance. Indeed, collusion of outacting members can cause the forgery of malicious data, thereby compromising, e.g., the integrity of the federation or the achievement of its goals. These deficiencies are more compelling in the European Public Sector, where public administrations are forced to adopt cloud interoperability solutions according to the new European Digital Single Market agenda.

In response to this need, as part of the European project SUNFISH (http://www.sunfishproject.eu/), we are contributing to design and implement an innovative cloud federation solution, called Federation-as-a-Service (FaaS) [4]. Based on the experience gathered within SUNFISH and collaborations with public administrations, this paper proposes a pioneering governance for FaaS that can lead to its wide-scale adoption in the Public Sector.

Our governance proposal is firstly based on the *distributed control* of data: all members have a consistent copy of data that cannot be corrupted by any means. Secondly, on the *democratic control* of governance actions: the federation is ruled according to consensus criteria ensuring that the rights of each member cannot be violated due to collusion of others. Finally, on *trustworthy* data services: access and sharing data services (e.g., access control and data anonymisation) must be protected to avoid confidentiality and integrity attacks [5]. It is worth noticing that the term *democratic* is used to reflect the direct participation and control of the federation by its members, in a way similar to a direct democracy, and carries no deeper meaning.

To implement this governance, we propose here a first exploitation of *blockchain* technology [6] as an infrastructure to build the federation *registry* underlying FaaS. Blockchain is a novel technology that, besides its application to cryptocurrency, features fascinating properties concerning integrity, distribution and control of data. More specifically, we utilise so-called *smart-contracts*, i.e. programs deployed and executed autonomously on blockchain.

The blockchain-based registry offers a set of core functionalities upon which our governance proposal is built. A preliminary implementation has been realised by using Ethereum (https://www.ethereum.org/). To the best of our knowledge, this is the first proposal to use blockchain to support a cloud federation, both to carry out the federation governance and to strengthen the trustworthiness of security services. On the face of it, FaaS appears to be the first *blockchain-based cloud architecture* of its denomination.

Paper Structure. Section II illustrates our governance proposal. Section III outlines FaaS. Section IV introduces the blockchain-based registry infrastructure. Section V discusses on it. Section VI concludes and touches upon future works.

II. A New Cloud Federation Governance

In this section, we synthesise and articulate an innovative governance for cloud federations that ensures distributed and democratic control of the federation, and strengthens the trustworthiness of security-preserving services.

The creation of a cloud federation is triggered by a business goal shared among the participating clouds. The cooperation of clouds to achieve the goal should be a priori defined to ensure that the governance is carried out with the consensus of all members. The cornerstone of the governance must be a *business contract* [3], which reports the types of services to be federated, the guaranteed SLA and the actions to be taken to rule the federation.

A key driver for the adoption of any cloud federation, especially in the European Public Sector, is the absence of a centralised governance. As a matter of fact, among different public administrations being federation members, there cannot be a designated leader (i.e., there is no *primus inter pares*), rather federation members must form a network of peers. To this aim, we identify the following key objectives to achieve a fully distributed governance

- *distributed* data, the governance data is consistently distributed among all the federation members; and
- *democratic* control, all federation members have the same obligations and rights, i.e. the same capacity of triggering and performing a governance action.

Achieving such objectives would ensure that any governance action, e.g. the enforcement of access control policies, is carried out with the consensus of all the members.

To ensure that these objectives are continuously guaranteed, the federation governance must secure the provisioning and sharing of federated service and data. Indeed, the accountability of all security-preserving functionalities of the federation (e.g., access control and data anonymisation) is of paramount importance. Therefore, the governance has to provide adequate means to ensure

- *trustworthy* data services, i.e. protecting services from confidentiality and integrity violation attacks.

A cloud participating to such a federation will be relieved from any additional security management task and enjoy advanced security-preserving functionalities.

To realise this distributed, democratic and trustworthy governance, it is needed a distributed infrastructure that, on one hand, ensures strong integrity and confidentiality guarantees of data and, on the other hand, supports the non-repudiable enforcement of the business contract.

III. FaaS: A Cloud Federation Solution

To address the need of cloud interoperability, the SUNFISH project has proposed Federation-as-a-Service (FaaS) [4], a new cloud federation solution. It amounts to a service for clouds that enables the secure creation and management of cloud federations.

FaaS is implemented via the SUNFISH software platform depicted in Figure 1; the description of its components follow. Most of all, the platform is conceived to be deployed in a distributed manner on top of all members, thus to avoid any *centralised control and component*.

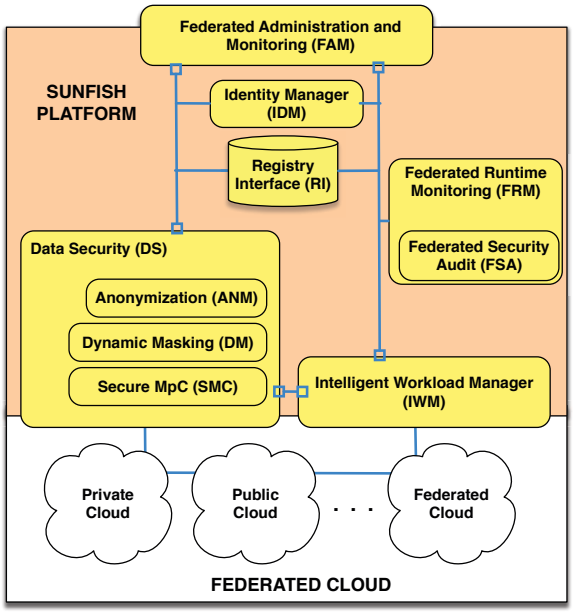

Figure 1. FaaS: Software Platform

The DS component offers a state-of-the-art attribute-based access control system distributed across all the member clouds [7]. By relying on the expressiveness of attributes, which are provided by a federated identity manager (IDM), the DS is transparently connected with security-preserving data sharing services: data anonymisation (ANM) and masking (DM), and secure data computation service (SMC). Specifically, data masking and anonymisation services are used, respectively, to ensure the privacy of sensitive datasets to be stored and released.

The inter-cloud interactions, controlled by the DS, are monitored by the FRM [8] via a distributed set of probes, and audited offline by the FSA.

The IWM and FAM are in charge of managing tenants by providing optimised workload strategies and SLA monitoring. Finally, the RI is the logical entry-point to the underlying blockchain-based infrastructure implementing the federation *registry* and realising the proposed governance.

IV. A Blockchain Infrastructure for Cloud Federations

To realise the FaaS federation registry and the proposed governance, we introduce here the use of a blockchain system featuring smart-contracts, both to store data and to offer computational resources.

The advantages of using blockchain amount to the strong integrity guarantees of the stored data and of the non-repudiable, persistence of smart-contract executions. Due to the replication of data on blockchain, the service availability is also always guaranteed.

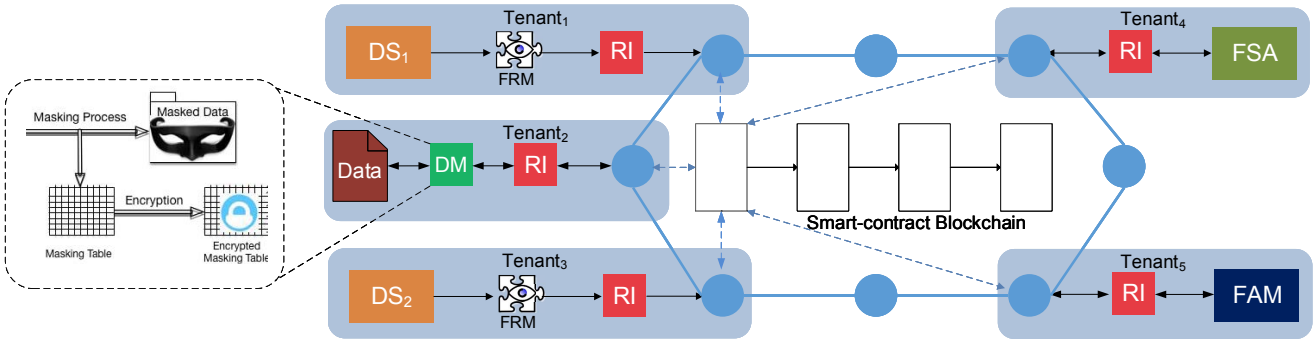

Figure 2. Blockchain Infrastructure and its interaction with FaaS platform components (Coloured boxes are components from Figure 1)

A. Functionality

The blockchain infrastructure implements the proposed governance by offering the following functionalities.

Federation Contract: It offers the storage of the business contract and the contract signature of each federation member. According to the needs, it supports the evaluation of precise metrics to evaluate the contract rules.

Federated Services: It offers up-to-date snapshots of the whole federation state, thus to correctly manage the allocation of and access to available services.

Access Control and SLA policies: It stores access control and SLA policies concerning provisioning of federated services. It also supports administration actions on policies.

Data Sharing Services: Although security-preserving techniques are proven individually secure, various studies (see, e.g., [5]) show that they can be circumvented, e.g. by linkage attacks. Thus, this functionality offers supporting functions for strengthening their reliability.

Federation Monitoring: It stores and processes logs gathered by the access control monitoring system.

B. Architecture

The infrastructure consists of a private blockchain system shared among multiple FaaS federations. Each federation has its own smart-contracts, one for each of the functionality, and relies on RIs to interact with them.

Focussing on a single federation, the infrastructure and its interaction can be represented as in Figure 2. Indeed, the platform components willing to use a functionality interacts via the APIs of the RI[1]. In its own turn, the RI invokes the corresponding smart-contracts with the given parameters and returns the received outputs. Push notifications are also supported to allow multiple side-effects of single invocations.

The architectural design based on API fully decouples the functionalities from the specific software used to implement the smart-contract blockchain. Additionally, exploiting one infrastructure for multiple federations has significant advantages. On the one hand, the more the nodes of the blockchain network are, the higher the integrity and availability guarantees are. On the other hand, as the infrastructure is standalone, new federations can be built upon by only setting up the needed contracts and their access parameters. Obviously, every RI can *only* access its federation data: a trusted computing platform is used to store a federation membership token needed for interacting with smart-contracts.

[1]https://github.com/sunfish-prj/SUNFISH-Platform-API/tree/master/RegistryInterfaceAPI

C. Implementation

A preliminary implementation of the blockchain-based registry for FaaS is based on Ethereum. Other blockchain solutions, e.g. Hyperledger (https://www.hyperledger.org/), could be similarly used. In the rest, we comment on the smart-contracts implementing some of the functionalities.

1) Data Sharing Services: This functionality concerns the security enhancements offered to the data masking and anonymisation services. To the former, it offers a reliable storage to secure the masking ingredients. To the latter, it prevents on the fly the degradation of ensured privacy levels.

Data masking. As shown in Figure 2, a masking table is generated and used by the DM to carry out the data (un)masking process. To avoid a centralised, untrustworthy storage of the table, we rely on a smart-contract. Specifically, the table is first encrypted with the public key of the party authorised to access the table. Then, it is divided into chucks and stored, together with the masking table identifier, via a smart-contract. Hence, the authorised party can download the table, decrypt it and unmask the data.

Anonymisation. The ANM offers, among others, a differential privacy [9] service for obfuscating sensitive datasets before release. To avoid linkage attacks that will degrade the privacy level of already released datasets, we rely on a smart-contract. It stores a *privacy budget* that (i) controls the amount of noise generated in the obfuscation; (ii) evaluates on the fly data release queries; (iii) adapts the used differential privacy parameters. As outlined in Listing 1, the contract

Listing 1
DATA QUERY RECORD (*'query' is a data structure*)

```
contract QueryRecord{
  struct record {
    string dataset_id;
    unit budget;
    mapping (unit => query) queries;
  }
  mapping (string => record) queryRecord;

  function evalQuery(query param) public returns (...){
    uint requestedBdgt = evalBudget(param);
    if (queryRecord[param.name].budget > = requestedBdgt)
        //query authorised, return differential privacy
             parameters, update remaining budget
    else
        //query not authorised
  }
}
```

maintains, via the data structure *record*, all the information on managed datasets. When a new query arrives, the function *evalQuery* checks the query parameters according to the available budget and provides the appropriate information to tune the release; the budget is then updated accordingly.

2) Access Control Monitoring: The distributed set of probes of FRM [8] are used to intercept and monitor inter-cloud interactions. The sensed logs are stored and evaluated by means of smart-contracts. Specifically, they perform semantics checks on the attributes forming access requests and on how the distributed DS components operate to carry out the distributed authorisation process.

Additionally, these checks are paired with off-chain, intensive checks on the policy evaluation process; this policy analyser is developed by using the formal framework in [10].

V. DISCUSSION

The proposed governance needs no trusted-third-party to base a federation upon. Hence, there is no single-point-of-failure and it advocates the democratic control and enforcement of the federation business contract, thus to avoid collusion attacks against federation members. This is realised by exploiting a blockchain-based registry.

The registry is also used to improve the security of the whole federation. In fact, on the one hand, it mitigates well-known vulnerabilities of data sharing services and, most of all, it ensures the availability of services. On the other hand, it puts in place the ingredients to support decentralised runtime monitoring of the federation.

It is also worth mentioning that we are aware of typical disadvantages of blockchain (i.e., limited speed, limited computing resources, possible scalability issues, etc.), but some preliminary research activities we carried out [11] exemplified that a balance between security guarantees and performance can be achieved. It is there introduced a layered blockchain deployment which anchors a fast blockchain, e.g. Hyperledger, to a slow one in order to enhance the overall security, while offering adequate performance.

VI. CONCLUSION

In this paper, we have presented a blockchain infrastructure for implementing a cloud federation registry and realising an innovative governance for cloud federations. The distributed and democratic governance properties ensured by this blockchain-based solution will pave the way to a wider adoption of cloud federation solutions, especially in the Public Sector.

In future, we plan to finalise the implementation of the infrastructure and to introduce new governance functionalities like, e.g., a reliable reputation system.

ACKNOWLEDGMENT

This work has been supported by the EU project SUNFISH project, grant agreement N. 644666.

REFERENCES

[1] T. Kurze, M. Klems, D. Bermbach, A. Lenk, S. Tai, and M. Kunze, "Cloud Federation," in *Cloud Computing, GRIDs, and Virtualization*, 2011, pp. 32–38.

[2] A. Celesti, F. Tusa, M. Villari, and A. Puliafito, "How to enhance cloud architectures to enable cross-federation," in *CLOUD*. IEEE, 2010, pp. 337–345.

[3] M. R. M. Assis and L. F. Bittencourt, "A survey on cloud federation architectures: Identifying functional and non-functional properties," *J. Network and Computer Applications*, vol. 72, pp. 51–71, 2016.

[4] F. P. Schiavo, V. Sassone, L. Nicoletti, and A. Margheri (Eds.), "Faas: Federation-as-a-service," *CoRR*, vol. abs/1612.03937, 2016.

[5] S. Garfinkel, "NIST SP 800-188: De-Identifying Government Datasets," 2016.

[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008, available at https://bitcoin.org/bitcoin.pdf.

[7] B. Suzic, B. Prünster, D. Ziegler, A. Marsalek, and A. Reiter, "Balancing Utility and Security: Securing Cloud Federations of Public Entities," in *C&TC*, ser. LNCS, vol. 10033. Springer, 2016, pp. 943–961.

[8] M. S. Ferdous, A. Margheri, M. Yang, F. Paci, and V. Sassone, "Decentralised runtime monitoring for distributed access control systems," in *ICDCS*. IEEE, 2017, *To appear*.

[9] C. Dwork, "Differential privacy," in *Proceedings of the 33rd International Conference on Automata, Languages and Programming*, 2006, pp. 1–12.

[10] A. Margheri, M. Masi, R. Pugliese, and F. Tiezzi, "A rigorous framework for specification, analysis and enforcement of access control policies," *CoRR*, vol. abs/1612.09339, 2016.

[11] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Blockchain-based database to ensure data integrity in cloud computing environments," in *ITA-SEC*, vol. 1816. CEUR-WS.org, 2017.