Towards Secure Collaboration in Federated Cloud Environments

Bojan Suzic, Andreas Reiter
Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology
Inffeldgasse 16a, 8010 Graz, Austria
{*bojan.suzic, andreas.reiter*}*@iaik.tugraz.at*

***Abstract*—Public administrations across Europe have been actively following and adopting cloud paradigms at various degrees. By establishing modern data centers and consolidating their infrastructures, many organizations already benefit from a range of cloud advantages. However, there is a growing need to further support the consolidation and sharing of resources across different public entities. The ever increasing volume of processed data and diversity of organizational interactions stress this need even further, calling for the integration on the levels of infrastructure, data and services. This is currently hindered by strict requirements in the field of data security and privacy.**

In this paper, we present ongoing work aimed at enabling secure private cloud federations for public administrations, performed in the scope of the SUNFISH H2020 project. We focus on architectural components and processes that establish cross-organizational enforcement of data security policies in mixed and heterogeneous environments. Our proposal introduces proactive restriction of data flows in federated environments by integrating real-time based security policy enforcement and its post-execution conformance verification. The goal of this framework is to enable secure service integration and data exchange in cross-entity contexts by inspecting data flows and assuring their conformance with security policies, both on organizational and federation level.

***Keywords*-cloud computing, federated environments, policy enforcement, XACML, data security, data privacy**

I. Introduction

A broad range of supporting tools, as well as extensive adoption of cloud-related technologies enabled public administrations to consolidate their infrastructure and use available resources more efficiently. Due to the heterogeneity of approaches and the ever increasing need for disposable processing power and storage, many organizations across Europe recognized the benefits to extend private cloud reach beyond their infrastructures by involving structured and cross-entity collaboration flows. This is particularly the case for public administrations who need to process large amounts of data in short time periods, or who need to perform data processing and data exchange with a range of external entities. Strict requirements in the field of data security and privacy, however, restrict the applicability of such collaborations.

The current situation across Europe is that private cloud data-centers from different public administrations are not allowed to share computational resources. Therefore, large data-centers are only fully utilized during peak times, where e.g. a monthly calculation on massive data-sets needs to be performed. The rest of the time the data-centers have loads of spare resources. The overall goal of the SUNFISH H2020 project is to address the lack of technology that allows public sector entities to federate their data centers, overcome legislative barriers and effectively utilize available computational resources. This goal is approached by considering specific requirements of these organizations and concentrating on security of cross-entity collaborative workflows.

The remainder of this paper is structured as follows. We start with providing related work and background information on the used technologies and mechanisms in Section II. We continue with the description of the defined use cases and our concrete contributions to the research community in Section III. In Section IV we elaborate on our contribution in federated intercloud security enforcement followed by integration and deployment strategies in Section V. Finally we discuss our results in Section VI and conclude in Section VII.

II. Related Work

In the context of cloud computing, security flaws or weaknesses on a low level can have impacts on dozens of virtual machines, involving hundreds or thousands of different users, compromising their privacy. Jansen [1] identifies key security issues associated with cloud computing:

- Trust: The security of data, or more generally the security of resources is in hands of the cloud provider. A tight trust relationship between the customer and the cloud provider is required.
- Architecture: The use of completely virtualized environments and resources provides large attack surfaces out of control of the customer.
- Identity Management: Identity management in the cloud, especially for federated clouds, is still subject of research and one of the goals to achieve in the SUNFISH project. As shown by Chow et al. [2] an organizations authentication system may not scale to the cloud without non-negligible effort.
- Software Isolation: Multi-tenancy and on-demand resource provisioning in a cloud environment are done utilizing virtualization technologies and executing various virtual machines on one single physical machine. The isolation of different virtual machines, different applications or just different users (depending on the utilized cloud deployment strategy) is

essentially impacted by the lower layers, like the operating system kernel and hypervisor.
- Data Protection: For multi-tenant cloud environments, data for different tenants typically is stored side-by-side, protected by the identity-based authentication system, which still is a major issue in cloud computing.
- Availability: Availability refers to the accessibility of outsourced IT infrastructure at all times, covering, amongst others, computing resources, data storage and databases.

To counter these issues various approaches are analyzed in the following sections.

A. Cloud Certifications

Cloud certifications provide certainty for end-users and businesses regarding various factors when selecting a cloud service provider. Certifications are voluntarily by its nature, but uncertified cloud providers may not be considered by end-users due to higher complexity and required effort to assess relevant factors. In fact certifications ease the selection process by the end-user and create confidence for the services and providers. Certification schemes have been developed with a focus on various different domains. In a survey conducted by the European Commission and ENISA "Certification schemes for cloud computing" [3] covering the most established and thorough certification schemes, a categorization of schemes was introduced: *General*, *Privacy/Data protection*, *Interoperability*, *Security*, *Service Management and Reliability/Access*. Using this analysis and the matrices as provided by the ENISA Cloud Certification Schemes Metaframework [4], the selection of certification schemes for cloud providers and customers is simplified.

B. Data Security

Data security in general needs to tackle the issue of *"who has access to resources/data"* in an access control fashion, but may also go beyond dedicated access control approaches. The decision if a particular entity has access to a specific resource may be governed by additional requirements. As a simple example, if a data holder allows a certain group of entities to access its data, she may additionally request to be informed each time the data is accessed. Furthermore, another group of entities may only be allowed to access the data in a masked way, where certain sensitive details are not accessible anymore.

In the remainder of this section the current most advanced approaches of data security policy definition languages are analyzed and reviewed.

eXtensible Access Control Markup Language (XACML) [5] is a standard maintained by the OASIS consortium. It is designed to provide a language where authorization policies can be specified in XML format in a structured and hierachic way. XACML can be seen as a de-facto standard on policy description languages influencing other policy related works. It primarily follows an attribute-based access control (ABAC) approach, but a profile exists to meet the requirements for "core" and "hierachical" role based access control (RBAC) [6] systems. The XACML specification not only specifies a policy language, but also components related to the enforcement of these policies and their interactions. XACML is targeted at enterprises which want to decouple the security policy implementation from their applications. All policies are stored in a single point and can be updated easily. Furthermore, XACML offers a consolidated view of the security policies in force for the system administrators. The specification is not dedicated to cloud or distributed environments, and needs adaption for these environments in terms of component interactions and regarding the distribution of components.

The *Formal Access Control Policy Language* (FACPL) [7] is heavily inspired by XACML, but provides a much more lightweight syntax with a solid mathematical foundation. Ponder [8] introduces a more fine grained categorization of access control policies into: authorization policies, information filtering policies, delegation policies and refrain policies. The *Obligation Specification Language* (OSL) as introduced by Hilty et al. [9] explicitly focuses on usage control of data. Usage control is an extension of access control with a focus not only on who may access the data, but also on how the data may be used afterwards.

C. Efforts on European Level

On European level several funded projects have been conducted with a focus on data- and cloud security. In the *Trusted Architecture for Securely Shared Services* (TAS3) [10] project a trusted architecture with adaptive security services has been developed. Its focus lies on next generation trust and security requirements, to enable a dynamic management of user-centric policies. Furthermore, it enables an end-to-end and secure communication channel for personal information between heterogeneous systems.

The *DEMONS* [11] project addresses issues in monitoring of the future Internet. It aims at building a decentralized and privacy-preserving cooperative monitoring infrastructure to detect, report and mitigate network threats in a cooperative manner.

The *EnCoRe* (Ensuring Consent and Revocation) [12] project was a research project from the UK industry and academia sector, with the focus to give individuals more control over their personal information, by means of defining what is allowed to happen to personal information disclosed to organizations. In contrast to other approaches, EnCoRe also focuses on the revocation problem, as for other consent based systems, giving consent is a final decision and cannot be undone. As SUNFISH does, the project also bases on the XACML approach. During the project some enhancements to XACML were developed which also influenced the developed SUNFISH approaches.

III. Use Cases and Contributions

The overall problem the SUNFISH project addresses is the lack of infrastructure and technology, allowing public

sector players to federate their infrastructures and allow a more efficient resource usage across different entities. To demonstrate the feasibility of the targeted solution, three use cases are defined and will be realized in the project.

A. Use Case 1 - Managing Salary Accounts

The General Administration, Personnel and Services Department of the Italian Ministry of Economy and Finance (MEF) are in charge of managing the payroll functions for more than 1.5 million Italian public sector employees. Managing a payroll system requires accessing highly sensitive data (such as health, religious orientation, information on military missions abroad) from multiple public and private entities including banks, central and local public administration or military agencies. All of the involved entities have a high interest in maintaining their client's privacy.

The procedure of producing payslips follows a complex workflow, which involves: (a) collecting data from relevant public entities, (b) checking the data, (c) maintain client's privacy by only revealing necessary data to relevant entities and (d) finally producing the payslips. This entire process needs to be completed within predetermined time, in order not to cause any delays in payment.

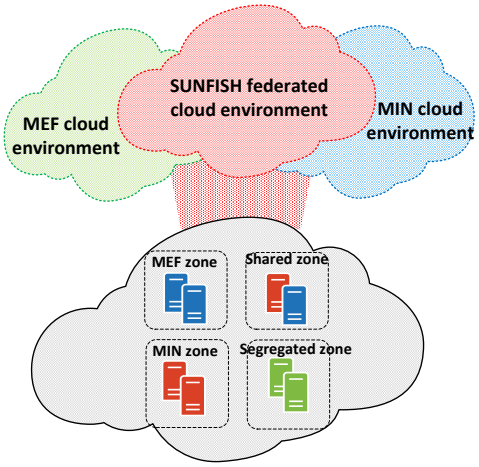

Figure 1. Use case 1 overview

The process currently in place involves multiple manual interactions and requires each department to upload prepared data files using the MEF payroll system to start the approval process. Utilizing the features provided by the SUNFISH federation this process is inherently simplified. As outlined in Figure 1, using the Ministry of Interior (MIN) as an example, the process should be enhanced to enable the calculation procedure to directly access the data on MIN systems and store data in encrypted or masked form in the internal system at MEF. To do this, a part of the MIN system will be deployed in the federated SUNFISH environment. This way, the process of extracting data from the MIN systems and to send them to the MEF systems is completely executed within the SUNFISH federated environment. Data is no more exposed outside the secured environment of MEF or MIN and always remains protected. Depending on security restrictions, permitted procedures in the federated environment have the possibility to directly access the source database. The segregated zone of the federation is subject to strict restrictions in terms of access, to perform computations on sensitive data. Sensitive data can only be accessed in plain in this dedicated zone and therefore is protected from unauthorized access.

B. Use case 2 - Financial Data Submission

The taxation department within the Ministry of Finance (MFIN) in Malta requires taxpayers, employers, banks and other third party data providers to submit information related to payroll, financial statements which qualify for deduction from chargeable income, receipts of payments and other related information to the commissioner of revenue. From medium-sized or large companies this information is currently collected through spreadsheets or data files and is submitted via web-site or web-services. Small businesses still use the paper channel. The tax authorities are seeking for a way to offer a software-as-a-service (SaaS) which helps to generate and submitting the required information. The SaaS solution should be integrated by using public cloud providers, but still enabling and maintaining the connectivity to private cloud providers.

The SUNFISH framework should enable the federation of resources between the MFIN cloud, public cloud providers and other third party cloud (e.g. payroll providers having their own data center).

On an abstract level the use case looks similar to use case one from chapter III-A described in Figure 1 with the difference that at least a third cloud environment, the public cloud environment, is added to the federation.

C. Use case 3 - Secure Cloud Storage for Data

The South East Regional Cyber Crime Unit (SERCCU) forms part of the UK response to cyber crime. The offenses investigated focus on cyber-dependent and to a lesser extent, cyber-enabled crime. Victims range from members of public through small and medium sized businesses to large corporations and government agencies.

The range of offenses investigated and the large geographical nature of internet-based investigations requires close interaction with partner agencies. The SERCCU is one of nine regional units and provides support to police forces within the South East region. However, interaction with other UK police forces is required on a case by case basis. The SERCCU also operates on a national level to assist the National Cyber Crime Unit, to investigate and prosecute offenders often based in Europe and those beyond European borders. In order to respond to dynamic changes in technology and patterns of offending they also need to work closely with industry and academia.

The unit obtains and stores large quantities of data which is highly sensitive, including high-level corporate information through to personal details about general members of the public. It also seizes and is required to securely store data produced from network servers and personal digital storage devices. The unit's investigations

will generate varying levels of intelligence, with each level requiring differing handling conditions. At present, the data is stored on a local server colocated within the unit premises. There are some limitations to this setup, one of which is that the server is not open to all the external parties that SERCCU interacts with. In this way, the unit is very limited in regards to data sharing which impacts negatively on the collaboration required in this field of work.

Law enforcement organizations need an effective way to securely share intelligence related to online criminal activity with other law enforcement parties. In addition, those organizations are increasingly sharing digital information with third parties such as victim companies. If a system were to be devised enabling sharing of digital information and intelligence, then third party secure access in addition to law enforcement agencies should be considered.

D. Requirements and Challenges

With a deep understanding of the use cases, a well-established path was followed to identify assets, related threats and finally derive requirements. This methodology is state of the art. It is e.g. seen in standards like Common Criteria security certification (ISO/IEC 15408)[13] of products, but also in organisations' risk management processes. In order to ensure completeness, the threat modeling has undergone iterative cycles of revisiting each step. Interviews with and reviews by domain experts have been carried out to complement that threat modeling team's analysis by the practical experience of partners that have comparable services in operation.

It is out of the scope of this document to repeat the complete analysis. In summary, the result of the in-depth analysis were 24 assets classified into six categories: users, federated environment, data, computational logic, operation, and infrastructure. Furthermore, 21 threats were identified, putting these assets in danger. The result were 32 requirements targeting the protection of the assets classified into four categories: general requirements, data requirements, federated environment requirements and usability requirements

On the infrastructure level the requirements target the integration of existing infrastructures and avoid building silos. The same applies on a software-layer. Existing applications should be able to run in the SUNFISH federation. From a data security perspective, the requirements target the self-determination of data providers. Even though data is stored within the SUNFISH federation, the data provider still requires full control of its data. This also enables cross-border use cases where regulations hinder the transfer of sensitive data in plain.

This work focuses on the data security related aspects of the discussed use cases. The developed solution will use the XACML policy description language as a basis for defining policies, and the specified components as a basis for the enforcement infrastructure. This decision raises the following challenges:

- *Integration of various different technologies:* SUNFISH is a heterogeneous system where different technologies from various enterprises work together: The cloud federation contains resources operated with different technologies, services need to communicate with each other regardless of the used technology, applications (legacy or dedicated SUNFISH applications) need to operate seamlessly. The challenge is to integrate all the different services and resources and make them available seamlessly.
- *Identity Management:* All enterprises have their own identity management systems, the challenge is to integrate these systems without replicating identity information, but still being able to cross-reference identities for policy definition.
- *Interoperability:* Applications and services need to run on the provided infrastructure regardless of the used hardware and technology.
- *Live transformation:* The use cases may require that sensitive data is masked or encrypted before it is stored outside secured environments. The transformation capabilities are provided by services within the SUNFISH federation. A near real-time transformation is required. This is challenging especially when it comes to large data-sets.
- *Identification of gaps and extensions of standards:* Currently no data security policy languages for heterogeneous distributed environments are available, where each data provider remains in full control of its data.

E. Contributions

In this section an overview of the contributions in this work is given.

Overall, a new security model is introduced, the *zoned* approach. In this model, the federation is grouped into zones with equal requirements on data security. On the edge of the zone, transparent gateways govern the access to and from other zones. The model is scrutinized in Section IV. From a data security perspective this work extends the XACML approach in multiple ways:

- XACML is extended to operate across enterprise borders. Data or resources in the system are associated with policies, the control of the policies remains in data providers' hands.
- The result of an XACML request, according the current specification, either is *permit*, *deny* or *indeterminate*. Another mechanism exists called *obligation* to induce further actions. This mechanism is extended to support data-security-related operations like data masking, data encryption or logging. This enables the integration of services in the enforcement process without applying any restrictions on the implementation.
- The enforcement components are extended to act as gateways and seamlessly enforce the defined policies, also for applications unaware of the XACML infrastructure.

- A decentralized decision process is foreseen, also to perform the decision-making at premises under the control of the data provider.

Beside data security related contributions a federated identity management model is elaborated, as an abstraction for the enterprise's identity management systems. The identity management model provides a mapping of enterprise identities to common SUNFISH roles. This way policies can be specified only referring to abstracted credentials, without knowing the details about enterprise-provided identity management systems.

IV. Intercloud Security Enforcement

Intercloud security enforcement in heterogeneous environments is a challenging task, due to the different possible interactions in the system. Our proposed *zoned* security model enables tenants to scale their required level of security. In the following sections, we introduce our general SUNFISH framework and go into detail on the enforcement infrastructure and policy evaluation process.

A. SUNFISH Framework

In this chapter we briefly introduce the general SUNFISH framework and introduce our *zoned* security model. Figure 2 shows a basic overview of the SUNFISH framework and on which cloud service level (first column) particular components are operating. The second column lists potential users of components on this level. For this work the *Identity Management*, *Data Security* and the vertical transformation components are relevant.

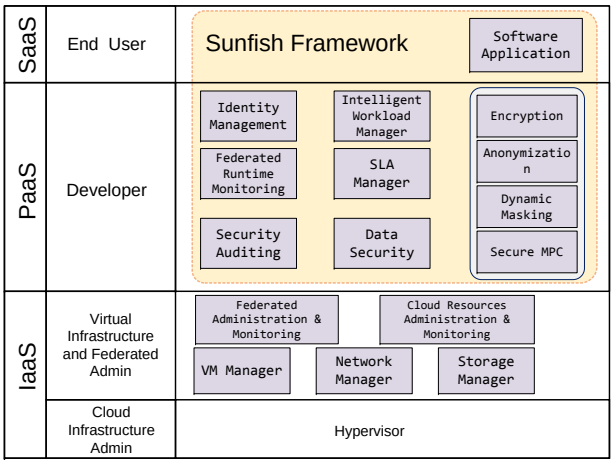

Figure 2. SUNFISH Framework

The federation generally is composed of multiple member clouds. Each entity (e.g. organization) which wants to use computing resources from the federation is called a *tenant*. A tenant acquires computing resources, adhering to certain rules (e.g. location of data center, operator of data center, regulations,...) to run services or applications. The computing resources are then isolated from other tenant's resources.

Based on this assignment we introduce our so-called *zoned* security model, where the data security component plays a major role in the realization. A zone is a collection of computing resources which operate on data with the same sensitivity-level. Each zone has a logical gateway which inspects all requests going out of or coming into the zone and enforces the defined policies. A simple example is a two-part calculation, where the first part operates on sensitive data and the more intensive second part on uncritical data. To realize this application a tenant deploys two zones with two services, one zone contains highly trustworthy computing resources which perform the first part of the calculation on the sensitive data and then forwards the data to the second service in another zone for the second part of the calculation. This zone is not qualified to operate on sensitive data, therefore the gateway masks the data before forwarding. This way sensitive data is never processed unprotected on unqualified computing resources.

Following this approach we are able to map complex data security requirements to heterogeneous cloud environments.

B. Enforcement Components

The security enforcement architecture follows and extends the approach proposed in RFC 2904 [14] and RFC 3838 [15], based on a general XACML data flow and enforcement model [5]. In this sense our architecture relies on the following components:

1) Policy Decision Point (PDP): evaluates requests and security policies, providing decisions and guidance to other components in the federated environment
2) Policy Enforcement Point (PEP): intercepts requests and enforces policy decisions made by the PDP, which may range from traditional access control decisions or more advanced transformational instructions
3) Policy Adminstration Point (PAP): used to create and administer security policies in the federation
4) Policy Information Point (PIP): provides attributes, contextual and environmental parameters used in access evaluation and the decision process
5) Policy Retrieval Point (PRP): stores the policies and acts as retrieval and synchronization point for other actors, such as the PDP and PAP
6) Data Transformation Service (DTS): a range of components used to analyze or transform the data intercepted by the PEP

The deployment of these components is illustrated in Figure 3. In this scenario, PEP acts as a transparent gateway deployed on the edge of each zone. Its task is to intercept each request coming to or originating from its subsumed zone. For this purpose, PEP interacts with other components in the system, such as associated PDP for policy evaluation, or PIP to gather the contextual parameters necessary for policy evaluation.

In the simplest case, PEP denies or grants access to data or services. More complex cases include real-time data transformation performed on particular parts of documents, potentially involving the execution of DTS

components, for each part of a workflow or operation that is executed on intercepted data.

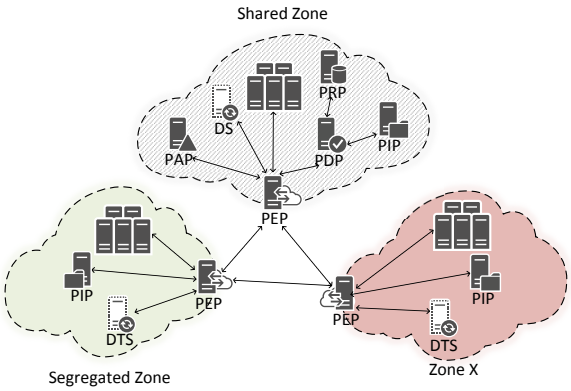

Figure 3. Deployment of enforcement components

C. Security Policies

We here briefly explain security policies, how they are instantiated and administered (to be reused in later sections).

Security policies in the proposed framework formalize and establish data security requirements, considering interactions and activities occurring in the scope of inter-cloud service integration, data sharing, and processing. Security policies hence consider the requirements from the view of the whole federation, its member organizations, as well as applicable legal frameworks and additional user requirements.

The data security framework relies on the concept of *policy sets*, *policies* and *rules*. Defined at lowest granularity-level, a rule represents a basic unit of a *security specification* that establishes relationships between resources, subjects, operations and contextual conditions. Integrated using a *combination algorithm*, a set of rules forms a policy, a basic unit for *security evaluation and enforcement* in the system, which can be further aggregated in *policy sets*.

In Figure 4 we present an illustrative abstract policy consisting of one rule and based on attributes that characterize each of the considered perspectives. An essential building block of security policies is an *obligation*, a specification included in authorization decisions describing operations that must be performed by the enforcement point prior releasing or denying access to particular resources or contexts. In the example provided in Figure 4, the obligation includes masking a data resource by applying a masking context that corresponds to a particular transaction.

```
Allow access to resource R with attributes {R1,…,Rn}
  If action A is read
  and subject S matches attributes {S1,…,Sn}
  and context C matches attributes {C1,…,Cn}
with obligation
  on Permit: doMaskResource(M*(R,A,S,C) and doLog(…)
  on Deny: doLog(…)
```

Figure 4. Abstract security policy

D. Cross-tenant Communication

The Policy Enforcement Point (PEP), deployed in each zone separately, is responsible for both the incoming and the outgoing traffic of the zone, performing *active* data flow inspection that assures the conformance with the federation and tenant-specific security policies.

To perform the access evaluation (including obligations) at the PDP component, PEP needs to issue an *authorization request* for each transaction that crosses the edge of the zone. In this process, PEP collects the necessary data and contextual parameters from the application and intra-zone PIP, finally preparing an authorization request and submitting it to the associated PDP for evaluation. The originating application request may be performed only after such decision is provided by the PDP.

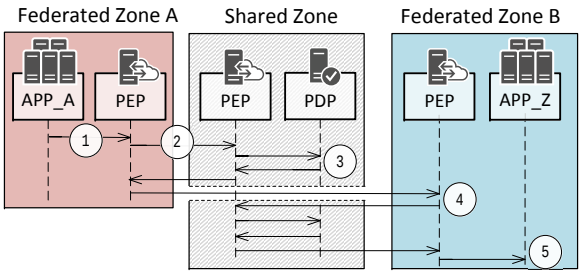

Figure 5. Cross-tenant interactions

Figure 5 depicts a flow between the entities in the framework, which for the purpose of simplification excludes PIP and DTS components inside a zone.

Based on the description provided in the figure, following steps describe the interaction process between $Application_A$ in $Zone_A$ and $Application_Z$ in $Zone_B$:

1) App_A prepares and issues a request, which is intercepted by the local PEP_A
2) PEP_A identifies the originating application, establishes the assurance level and gathers the contextual data from the PIP_A. At this point PEP_A completes and issues the authorization request to PDP_{SZ}.
3) PEP_{SZ} intercepts the incoming request and forwards it to the PDP_{SZ}, which evaluates the request and provides the decision to PEP_A. This decision may include obligations to reduce the information footprint by applying transformational operations.
4) Instructed by PDP_{SZ}, PEP_A forwards the request to PEP_B, which gathers the contextual data and issues a new authorization request to PDP_{SZ}.
5) Following the evaluation performed by PDP_{SZ}, PEP_B follows the instructions, performs necessary actions and delivers the request to App_Z.

The presented workflow enables the *dynamic* and *transparent* enforcement of security policies that are evaluated at *run-time*, based on a particular *context* and actors' *attributes*.

E. Data Transformation

Data sharing between zones may require data transformation to a view that in a context-specific manner reduces

information provided to adjacent party. The primary purpose of this transformation is the enforcement of organizational policies that deal with information security, privacy protection, and legislative compliance. Performed by Data Transformation Service (DTS), a component deployable in each zone, this processing can be applied before the data leaves its origin, or before the data reaches its destination environment. This component acts as a service, exposing its functionality to local systems and its PEP on the edge of the zone. In both directions it acts as an intermediary, providing the data only in the view that is necessary to accomplish the task and conform to legal or organizational requirements [17], [18].

The primary use case considers the deployment of DTS in a segregated zone, with the aim to provide the following functions: (a) *encryption and decryption*, (b) *data masking and tokenization* and (c) *key and tokenization management* There are basically two approaches for integration of DTS.

The first approach is a PEP centered approach, where the PEP performs all the interactions with the DTS. This scenario is appropriate for cases that assume data processing in batches or work on smaller data sets. For the applications that require data-streaming and the processing of big data sets, the intermediary role of PEP might require additional resources or could add unacceptable overhead to complete the processing cycle. These cases can be addressed with the second approach.

The second approach assumes the deployment of DTS in a proxy configuration, acting as an unidirectional intermediary between PEP and the internal processing services.

The DTS is meant to be accessed as a service, with the exposed functionality for each type of supported processing through transformation interfaces. In its practical realization, this service might integrate or depend on other services and components.

This component may support the transformation of data in both directions. The first case considers transforming the data to their privacy-preserving equivalent for a particular service and purpose like tokenization of structured data, with the goal to remove personally identifiable information. The second case considers the reverse operation with the purpose to get the data in its original form. This functionality, however, depends on the particular transformation type and might not be available for all transformations. One example of reverse transformation is the decryption of previously encrypted (transformed) data. In this case, the service outputs decrypted data, for the purpose of local processing.

F. Proactive Monitoring

The proposed framework integrates various subsystems using different interfaces, where each component performs a particular and isolated task in a broader workflow. In order to gain an aggregate view of all the flows and processing that occur across the systems, the proposed framework includes additional component, responsible for monitoring events, their correlation and integration into coherent views for the purpose of separate analysis.

The monitoring component consists of a remote agent and a central processing service. This service can potentially be replicated to meet performance and security requirements. The agent is attached to other components in the system, such as PEP, PDP or DTS, intercepting and logging their activity. By gathering and correlating event-related data, the processing service is able to identify anomalies in the system. In addition to that, the monitoring component can be used to verify the functionality and correctness of particular services by correlating processing results obtained in different contexts and under separate environments. This way, the correctness of authorization evaluation performed in the scope of PDP, or data masking performed by DTS, can be verified in near-real time.

V. Deployment and Integration

A. Integrating Applications

Integrative potential of applications and services deployed in federated zones may differ in a range of features. In particular cases of legacy, costly-to-maintain or applications that exhibit a fair level of complexity, the adjustments and integration in the federated framework may introduce additional overheads in terms of integration costs and security. For this reason the proposed framework considers that PEPs may take one of two principal roles:

1) Transparent intercepting entity acting as a proxy
2) Intermediate service for framework-aware and customized applications

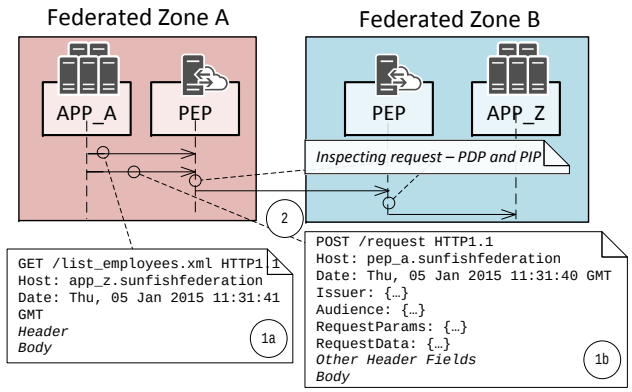

Figure 6. PEP interactions

In the first case, applications from the internal zone are part of the intercloud federation but their awareness of the processes and general integration level are reduced to a minimum. This scenario is suitable for legacy and complex applications that need to be integrated in a framework in cost-effective manner. Such applications do not have to implement additional protocols to support their integration in federated environment. Instead, they may issue the requests in a standard way, allowing PEPs to transparently intercept and process them in accordance with security policies and workflow prioritization, as defined in configuration environment and decided at run-time.

In the second approach, the PEP acts as an intermediate service. Applications are aware of the federation framework, actively integrating its flows and functionalities in

their workflow. Such applications can utilize the federated framework and its functions in full extent, allowing the integration of dynamic, application-specific claims at request-time.

Figure 6 depicts interactions between App_A and App_Z considering both models. The first model, illustrated as (1a), presents a request issued by originating application that is not aware of federation environment. This request is intercepted by PEP_A and transformed to conform to framework's flows. For this purpose PEP_A may gather configuration data from local PIP in order to make requests compatible with the intermediate components and target application.

The model denoted as (1b) illustrates the flow initiated by framework-aware applications. In this case APP_A directs its request to PEP_A and its endpoint, consuming the functions, options and headers available in the federation framework. This way, the application may augment settings provided for dynamic adjustment. Upon receiving such request, PEP and its adjacent PDP are able to evaluate *dynamic*, application and context-specific requirements and act upon them in an *adaptive manner*.

B. Cross-request Authorization

The standard flows, as introduced in Section IV-D and depicted in Figure 5, assume the generation and evaluation of authorization requests for each interaction separately. Some collaborative flows may introduce excessive amount of repeating requests that include a limited range of replicated and similar requests. Such environment may benefit from aggregated authorization request and reponses that span across several interactions.

In the scope of this work we introduce the concept of *cross-request authorization* that enables PEPs to issue authorization requests, and PDPs to provide authorization responses spaning across multiple transactions. These trasactions may conform to particular requirement based on *relaxed requirements* or *activity context*.

The first example of such cases applies particularly to batch processing of huge amounts of data, involving interactions between the same entities. Instead of adding processing overhead for each request, a PDP may decide to frame the conditions to a particular workflow and instruct PEPs to enforce its decision to a range of requests.

The same flow can be applied to a range of activities. These include interactions between parts of the monitoring infrastructure, the security governance infrastructure or configuration data flows.

C. Decentralized and Distributed Policy Decision

The model introduced in Section IV-D primarily considers the existence of a centralized PDP, deployed in a shared zone, which represents a recommended approach for a federation of entities. However, additional requirements may arise building on complex configurations, use-cases or employed trust models. We identify two main deployment and integration models that specifically address additional requirements from domains of security and performance optimization.

The *decentralized* deployment and integration model considers the possibility to utilize several shared zones, and as part of them, different PDPs. In this scenario, PEPs issue authorization requests to disparate PDPs in parallel and enforce their decisions in terms of *composite* and *consesus-based* evaluation. The rules determining the endpoints for decision requests, their applicability and scopes, as well as suitable cases and combinatorial logic to be used, are stored in the common framework configuration and are dynamically replicated to zone-specific configuration registers across the federation. These rules are provided to each component on initialization time with changes dynamically pushed in real-time.

Applicable cases supporting the integration of this scenario include additional security requirements, enabling evaluation of rules to be done by distinct parties or infrastructural components. This may reduce risks from potentially compromised and manipulated governance infrastructure components. Furthermore, enabling PEPs to acquire several PDPs may be used as a load-balancing mechanism to adhere to performance related requirements.

The *distributed* deployment model considers zone-specific and local deployment of PDPs, assigning them with the responsibility to handle all authorization requests coming from their perimeters.

This model raises two additional requirements. First, the policies residing in a centralized repository in a shared zone need to be dynamically synchronized to each zone. These policies represent a subset of the global federated policies, considering only the ones that are relevant for the particular local environment. The relevancy of policies is determined by the centralized PAP.

This second requirement introduces *layered authorization* by establishing dependencies of local PDPs to the centralized PDP for types or scopes of requests that cannot be decided locally. Thus, in selected cases or undecidable requests, locally deployed, zone-specific PDPs issue additional authorization requests to centralized PDPs and use their results in the local decision process.

The distributed deployment model is particularly beneficial for federations esablished across distant or resource-limited networks. By increasing the levels of resilience and scalability, this model supports the scenarios that include highly variable loads with massive spikes in authorization requests, as well.

D. Identity Management

One of the dependencies that enable effective enforcement of security policies in federated environments refers to the integration of the identity management systems. This integration enables the security governance platform to derive data on collaborating entities, their roles and permissions in regards to resources and services, and to include this information in the overall process of the access decision. In the proposed framework we acknowledge the fact that the cloud federation consists of entities from various organizations, using diverse internal policies, techniques and models for authentication. For the purpose

of successful operation, this data needs to be integrated, translated and provided both to the critical security governance components and adjacent collaborating entities.

Based on the use cases presented in Section III, the primary type of interactions ocurring in the scope of the federation framework include automated data processing and request execution, triggered mainly by automated application flows. Focusing on transactions taking place in cross-zone and cross-domian context, we identify two main categories of interactions and involved actors:

1) Accesses initiated in the scope of interactive sessions controlled by human operators - *users*
2) Accesses initiated by automated *agents* that perform background tasks independently of human actions

In this section we introduce both categories and establish distinction points relevant for their integration.

User-driven interactions encompass activities initiated by users registered either on the level of federation, or on the level of its individual members. Considering that each federation member may run and maintain its own identity management system, the common framework does not aim to impose particular requirements to the flows that occur inside its infrastructure. This framework, however, considers identity management as a common functionality additionally provided at the level of federation. The proposed framework, therefore, aims to enable the integration of existing and heterogeneous identity management systems present at the member level, with the purpose of enabling 1) cross-zone user-driven interactions, and 2) federation-level administrative operations on common framework components.

The approach proposed in the SUNFISH framework relies on RBAC administrative model [16], establishing the roles on the level of federation. The roles are assigned to users in the scope of interactions with common framework components or federation member services that request users to be authenticated in common role. For this purpose, the PIP component introduced in Section IV-B is used as an intermediary entity that integrates with the local identity management system, providing authentication assertions and role translation defined in the common framework.

Agent-based interactions enclose the flows executed between federated services in cross-zone context, effectively establishing *machine-to-machine communication*. In contrast to user-driven interactions, these flows are subjected to deep inspection and transformation performed by the PEP at the edge of the zone. Hence, the local PEP, in coordination with the local PIP, ensures the authentication of applications, providing assertions to adjacent parties in the federation.

Figure 7 illustrates the process of authenticating applications using *SAML* protocol. In the first step, APP_A issues a request to APP_B from $Zone_B$. PEP transparently intercepts the request and, after fetching application specific configurations from local PIP, redirects APP_A (2) to the local identity provider administered by one of the federation members. After performing this process, the PEP receives the assertion (3), and transforms and

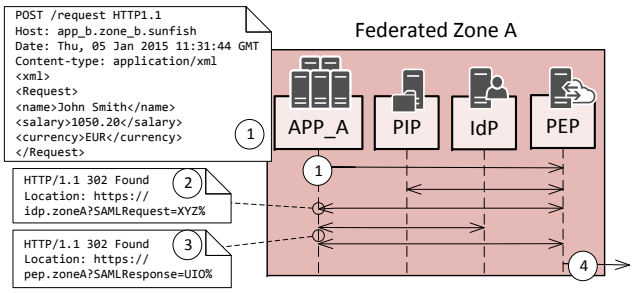

Figure 7. SAML-based authentication for cross-zone flows

forwards the initial request to the target zone (4). The simplified response sent to other party is shown in Figure 8. This response, delivered to PEP_B, includes the assertion prepared and signed by PEP_A, as well as the original request which may be subject to data masking, data tokenization or data encryption[1]. Besides *SAML*, the proposed framework includes the support for *OpenID Connect* and *key-based authentication flows*.

```
POST /incoming HTTP1.1
Host: pep_b.sunfish
Date: Thu, 05 Jan 2015 11:31:44 GMT
Content-type: application/json
Issuer: {"zone":"zonea", "entity":"pep_a", "source":{"entity":"app_a"}}
Audience: {"targetZone":"zoneb", "targetService":"se28ajs"}
Validity: {"notValidBefore":1450115468, "notValidAfter":1450115468}
Context: {"anonymized":true, "authenticated":true, "authdata": {...},
  "masking":{"masking-context":"c8102e"}}
RequestData: {"method":"POST", "host":"app_b.zone_b.sunfish",
  "path":"/request", "content-type":"app/xml",
  "headers" : {"header1":"x", "header2":"y"} }
Signature {"type":"jwt","value":"eyJhcI..." }
<xml>
<Request>
<name></name>
<salary>1050.20</salary>
<currency>EUR</currency>
</Request>
```

Figure 8. Transformed request delivered to the adjacent zone

VI. Discussion

In this work we approached the topic of security in federated clouds from the perspective of cross-organizational collaborations. Emanating from project-specific use cases, requirements of public administrations and general data protection legislation, this perspective puts strong focus on objectives of data security and privacy. Serving as a starting point for other requirements, these objectives represent prerequisites for a broader adoption of cloud services and establishment of more efficient, structured and trusted collaboration workflows across public entities.

This work addresses the challenges elaborated in Section III-D by relying on *holistic* and *integrative* principles. The latter is employed by introducing and extending a range of building blocks that provide complementary functionalities, enabling synergistic mixture of their features. This way, we combine real-time enforcement of security policies with their post-executional conformance verification performed in a separate context on a replicated infrastructure, as introduced in Section IV-F. The resulting reliance on multiple verification processes deployed at distinct premises reduces the attack surface and provides

[1]Interaction is omited in figure for the purpose of simplification

an additional confidence layer covering the security enforcement. The same applies to the decentralized deployment, described in Section V-C, which enables parallel evaluation of security policies across different federated members. Incorporated and executed in a transparent manner, these two approaches together strengthen security of collaborative processes in the federation.

The challenge of data protection, approached by Schneider's *least privilege principle* [17] and formulated in a range of legislations [18], is addressed by relying on a range of technologies. Namely, the proposed framework implements data protection by reducing information footprint in cross-domain interactions, allowing the actors to consume only minimal and allowed data sets. This process is supported by applying techniques such as data masking, tokenization or anonymization, or alternatively, by protecting structured data parts with the application of format preserving or standard encryption techniques.

The integrative principle in the framework is demonstrated with lightweight and transparent deployment of existing technologies and heterogeneous systems. In this sense, optional transparent policy enforcement for legacy applications allows this category to benefit from security functionalities with a minimal integrational overhead. Similarly, the identity management subsystem presented in Section V-D allows transparent reuse of existing systems and infrastructure, supporting authentication of agents and users, as well as integrity and assurance of the flows.

VII. Conclusion and Future Work

Public administrations are progressively adopting a diverse range of cloud technologies, actively benefiting from increased efficiency, reliability, and control of their infrastructures. Growing amount of data that needs to be processed in short time periods and provided to other organizations impose the need to share infrastructure and execute collaborative processes in a cross-domain manner.

In this paper, we presented an ongoing work that enables these goals by establishing private cloud federations for public administrations. Extending a range of standards and technologies, the proposed framework considers a range of additional requirements in terms of data security and privacy. In the scope of future work, we intend to validate the proposed framework by employing the prototype currently in development in three use cases that encompass public organizations in three European countries.

Acknowledgment

This work has been supported partially by the SUNFISH project (N.644666) funded by the European Commission H2020 Program.

References

[1] W. A. Jansen, "Cloud Hooks: Security and Privacy Issues in Cloud Computing," *System Sciences*, pp. 1–10, 2011.

[2] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, "Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control," *Proceedings of the 2009 ACM workshop on Cloud computing security*, pp. 85–90, 2009.

[3] Trilateral Research & Consulting, *Certification Schemes for Cloud Computing*, 2014. [Online]. Available: http://bookshop.europa.eu/en/certification-schemes-for-cloud-computingpbKK0414719/

[4] ENISA, "ENISA Cloud Certification Schemes Metaframework." [Online]. Available: https://www.enisa.europa.eu/news/enisa-news/enisa-cloud-certification-schemes-metaframework

[5] OASIS, "eXtensible Access Control Markup Language (XACML) Version 3.0," 2013. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html

[6] ——, "XACML v3.0 Core and Hierarchical Role Based Access Control (RBAC) Profile Version 1.0," 2014. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-rbac-v1-spec-en.html

[7] A. Margheri, M. Masi, R. Pugliese, and F. Tiezzi, "Formal Access Control Policy Language (FACPL) User's Guide," Tech. Rep., 2016.

[8] N. Damianou, N. Dulay, E. Lupu, and M. Sloman, "The Ponder policy specification language," *Proceedings of Policy 2001: Workshop on Policies for Distributed Systems and Networks*, pp. 18–39, 2001.

[9] M. Hilty, A. Pretschner, D. Basin, C. Schaefer, and T. Walter, "A Policy Language for Distributed Usage Control," pp. 531–546, 2007.

[10] TAS3 Consortium, "TAS 3 Architecture," Tech. Rep., 2011.

[11] S. Niccolini, F. Huici, B. Trammell, G. Bianchi, and F. Ricciato, "Building a Decentralized , Cooperative , and Privacy -Reserving Monitoring System for Trustworthiness: the Aproach of the EU FP7 Demons Project," *IEEE Communications Magazine*, no. November, pp. 16–18, 2011.

[12] E. A. Whitley, "Informational Privacy, Consent and the "Control" of Personal Data," no. October, 2009.

[13] Common Criteria, "Common Criteria for Information Technology Security Evaluation Part 1 : Introduction and general model September 2012 Revision 4 Foreword," *ISO/IEC 15408 Common Criteria, Part 1:2012*, no. September, 2012.

[14] P. Calhoun, S. Farrell, G. Gross, and D. Spence, "AAA Authorization Framework [RFC 2904]," *RFC 2904*, pp. 1–35, 2000.

[15] A. CalhoBarbirun, O. Batuner, A. Beck, T. Chan, and H. Orman, "Policy, Authorization, and Enforcement Requirements of the Open Pluggable Edge Services (OPES) [RFC 3838]," *RFC 2904*, 2004.

[16] D. Ferraiolo, J. Cugini, and D. R. Kuhn, "Role-based access control (rbac): Features and motivations," in *Proceedings of 11th annual computer security application conference*, 1995, pp. 241–48.

[17] F. B. Schneider, "Least privilege and more," in *Computer Systems*. Springer, 2004, pp. 253–258.

[18] A. Kertesz and S. Varadi, "Legal aspects of data protection in cloud federations," in *Security, Privacy and Trust in Cloud Systems*. Springer, 2014, pp. 433–455.