Zentrum für sichere Informationstechnologie – Austria
Secure Information Technology Center – Austria

A-1030 Wien, Seidlgasse 22 / 9
Tel.: (+43 1) 503 19 63–0
Fax: (+43 1) 503 19 63–66

A-8010 Graz, Inffeldgasse 16a
Tel.: (+43 316) 873-5514
Fax: (+43 316) 873-5520

http://www.a-sit.at
E-Mail: office@a-sit.at

DVR: 1035461
ZVR: 948166612
UID: ATU60778947

E-ID IN THE CLOUD WITH SCIM

VERSION 1.0, 15.02.2015

Bojan Suzic – bojan.suzic@a-sit.at

Abstract: Cloud computing actively transforms the way information technology products and services are designed and delivered. Due to the wide range of benefits introduced with the cloud paradigm, not limited only to domains of increased efficiency, flexibility and scalability, cloud computing has been identified as one of the key technologies and innovation drivers in the industry. Numerous national initiatives and actions confirm the perception of cloud computing as important technology from the standpoint of public authorities, too. Greater flexibility and expanded deployment options introduced with the cloud however open up new use cases and new challenges. One such challenge is the integration of heterogenic cloud services in the organizational identity management processes and infrastructure.

This report provides a general overview of the topic and provides an analysis of the approach introduced with SCIM – the System for Cross-domain Identity Management. Although identity provisioning has been addressed with the SPML standard, its high complexity, limited flexibility and lack of the consideration for cloud cases were identified as main reasons of its low adoption and ceased development. SCIM, which is currently proposed as 2.0 internet draft under the IETF standards track, tries to address identity provisioning in the cloud from a minimalistic and flexible perspective. This report examines the problem of identity provisioning in the cloud, establishes the problem and terminology, and considers prominent use cases. It additionally focuses on identity provisioning considering perspective of eID. The report further deals with the potential integration of SCIM and eID by positioning it in the frame of the Austrian eID solution and STORK-based cross-border context, discussing possible challenges, solutions and further work.

Summary:

- SCIM is relatively new, JSON and RESTful based lightweight approach to identity provisioning in the cloud, providing the capabilities to encompass authentication and authorization of the users.
- There are currently 25 known implementations of SCIM 1.1 and 2.0, of them 13 are licensed under open source and 3 relate to 2.0 draft (in development).
- The integration of SCIM with eID is possible by defining extension of the core schema and integrating it in the landscape. The core documents provide guidelines on how the extension is done and registered.
- Integration of Austrian eID raises an issue of sector-based eIdentifiers of the users and their correlation.
- The integration of authorization functionality compatible with PVP (Austrian authorisation federation “Portalverbundprotokoll”) is possible out-of-the box, as the proposed schema encompasses attributes for roles and entitlements of users.
- The cross-organizational and cross-border identity provision raises the issues of privacy, data leakage and user tracking. Although SCIM does not provide an explicit solution for this problem, it enables the solution based on existing technologies or approaches to be applied. In addition, the approach provides a higher degree of control on privacy relevant data in the cloud than one present in non-structured application. This is due to the fact that the data exchange is executed under identity provisioning framework, providing processes for control, transform and audit of user data exchange.
- Adoption of eID in the terms of structured and standardized identity provisioning raises a number of benefits and issues, which are further elaborated in the section 5.8.

Revision History

Version	Datum	Author	Comments
0.1	9.10.2014	Bojan Suzic	Initial draft
0.9	29.01.2015	Bojan Suzic	Completed section 6

Table of Contents

Figures

Tables

1. Introduction

The cloud paradigm received much attention in the recent years, what led to the introduction of new technologies, markets and service approaches on a global scale. Industry mainly has pushed forward the cloud-based approach, transforming the product portfolios and the way the services are produced, delivered and consumed.

Wide adoption of cloud services transformed the way businesses establish and perform, as well as their primary business cases. New cloud-enabled solutions benefiting from reduced time-to-market requirements enabled vendors to develop new products at a faster pace, lowering market entry barriers and calling for a new entrants. The faster, modular and more diverse delivery of various services subsequently contributed to the emergence of new markets, providing the additional innovation impulse and customer coverage to the producers. The cloud paradigm and its base outsourcing approach however increased the interdependence among stakeholders, services and industries, raising the level of complexity of solutions.

One of the popular views of cloud services is represented by the layered classification based on three service delivery models: *IaaS*, *PaaS* and *SaaS* (Infrastructure, Platform, or Software as a Service) As such, they reflect the main building blocks of cloud services, based on different vertical levels of provision [1].

Many enterprises realised the benefits of the cloud and started early with the adoption and integration of cloud services within their organisations. This adoption eventually led to complex setups, stretching within or through various domains, organisations, or even countries.

On the other hand, public authorities globally recognized the benefits of the cloud too, but their adoption rate was relatively slower in comparison with their counterparts from the industry. As the issues causing slower and eager adoption of cloud solutions, public authorities often report concerns related to security, data protection, compliance and audit, as well as interoperability and data portability [2]. The topic of this report corresponds to a fifth category, identity and access management.

The trend to transfer the services to the cloud raises the issue of their efficient management in heterogeneous, cross-organisational context. Consequently, it introduces new challenges on how to securely and efficiently orchestrate access management over such resources. These issues are the central point of the research behind this report. It deals with the identity provision for the entities in cross-organizational, cross-domain scenario, with the focus on the applicability of the particular approach currently in development for the use cases of public authorities.

1.1. Problem and Motivation

Traditionally, maintenance and management of users' accounts has been considered as one of the core responsibilities of Identity Management (IdM)[1]. These responsibilities include the establishment, integration and continual execution of a set of processes, tools and contracts relating to the creation, maintenance and termination of digital identities within one enterprise [3].

Cloud Security Alliance [4] recognizes four main functions of Identity and Access Management (IAM):

- Identity provisioning and deprovisioning
- Authentication and federation
- Authorization and user profile management
- Support for compliance

[1] Also referred as Identity and Access Management (IAM). In the scope of this work it is assumed that the both terms refer to the same concept.

IdM as process usually includes the maintenance of account information necessary to establish secure and proper access of users to services and applications. Due to the heterogeneity of the systems and architectures present within one organization, one of the typical approaches applied to IdM was to establish organization-centric data facility, whose role is to store access policy information for its users. LDAP (Lightweight Directory Access Protocol) based directory is one of the examples of such facilities, providing user access information to services through structured and standardized approach.

Maintenance of account information embraces provisioning, an automated procedure applied through lifecycle of identity within the organization. As such, it encompasses not only the creation of identities (provisioning, or on-boarding), but also their association with other resources, regular update in case of changes, and finally, decommission (deprovision or off-boarding), according to the policies and other requirements.

Recent paradigm shift led to the introduction of innovative, but more diverse and decoupled services based on cloud technology. As a result of that, additional requirements and complexity have arisen for the management of IdM systems. Now, the provisioning process has to be executed throughout heterogeneous environment, connecting various services usually run by external entities, in rapidly evolving environment. Such change increased the complexity of provision and introduced the challenge of managing the provisioning process in cross-organizational, usually not standardized context.

Some of the standards have been already established with the intention to provide or enable the solution for automated, cross-organizational provisioning. The examples of these include SPML (Service Provisioning Markup Language), SAML (Security Assertion Markup Language), and recently drafted SCIM, which is gaining the attention in the community.

The fact is that today there is no widely accepted standard for cross-organizational provision, even eight years after approval of SPML 2 framework. As the cloud services gain wide momentum globally, the proper onboarding of the resources and users in complex environments become even more important for all the stakeholders involved. They include both cloud service providers and cloud service consumers, each facing specific issues from its perspective.

As a part of an agenda to increase competitiveness, reduce operational costs and provide innovative services for their citizens, many public authorities are considering or have already integrated or established cloud services in their processes. One of the additional challenges they face in this process is the integration of public eID in the process of cloud integration and service-user provision.

1.2. Objectives of this report

Based on previous introduction, the objective of this report is to: 1) introduce the issue of orchestrated provision from customer and provider perspectives, 2) investigate SCIM as a one of possible solutions, 3) analyze the actual draft and open implementations of second (draft) SCIM incarnation and 4) provide the overview on its possible integration with public eID approaches.

The management and synchronization of identity data present at several premises are the central points of the solution proposed with SCIM protocol and it's API. This this document approaches the problem and analyzes the method suggested by SCIM. Its further objective is to evaluate the applicability, benefits and disadvantages of SCIM based approach for parties who rely on public eID for identification and authentication of their employees.

1.3. Organization of this report

Starting with previously stated objectives of this report, the structure of this work as follows. Chapter 2 approaches the problem of identity provisioning, defines main terms, concepts and

challenges. Finally, it introduces SPML provisioning standard and in a limited extent compares it with recent draft of SCIM 2 standard, which is a subject of this work.

Chapter 3 continues the path started in the previous chapter by focusing on particular SCIM implementation and approach. It first provides SCIM scenarios and use cases. They are followed by introduction of SCIM schema and SCIM API. Finally, the essential differences between SCIM 1.1 and 2.0 are explained.

Chapter 4 analyzes the state of SCIM implementations. It starts with basic information on registered implementations of both 1.1 and 2.0-draft recommendations. Then, two most prominent open source and publicly available implementations of SCIM 2.0-draft are briefly described.

Chapter 5 focuses on eID in the context of SCIM. It first analyzes the scope of eID and provides an overview on two main eID approaches and their relevance in this work. Then, the national and international contexts of eID are evaluated, whereas the national approach and its integration in the international framework are explained in the example of Austrian national eID. Chapter 5 then considers the possible integration of eID into SCIM flow by providing descriptive use case, workflow description and analysis of main challenges. It continues with the particular use-case analysis, considering the case of integration of Austrian national eID. The section closes with final considerations on eID adoption from the perspectives of cloud service provider and enterprise cloud user.

Chapter 6 provides the summary and the conclusion of this report.

2. Identity Provisioning in the Cloud

Although it cannot be seen as the first activity performed in order to provide a systematized and standardised approach to the provisioning of the services, one of the important efforts in that direction has been contributed by OASIS Provisioning Services Technical Committee (PSTC). As a part of their undertaking to contribute with systematized and standardised view on provisioning, they provided a following formal and short definition of the term provisioning [5]:

> *Provisioning is the automation of all the steps required to manage (setup, amend & revoke) user or system access entitlements or data relative to electronically published services.*

As such, this process covers a wide range of activities which have to be performed on a wide range of interconnected and heterogeneous systems and integrated in the adjacent business process of the host organization. In the practice, provisioning often deals with technical and administrative burden on managing the entitlements and access rights of the users which need or have to consume electronic services inside or outside of organisation.

The purpose of *deprovisioning* is to remove user accounts, privileges and clean related data from the connected services. Such process has to be performed when, for instance, user leaves the organization, or its authorization to consume the services ceases for some other reason. Hence, the role of deprovisioning in the security perspective is to eliminate orphaned accounts and enforce full control over user's accounts and associated data across internetworked systems and during complete lifecycle of IAM.

Although intuitively tends to be assumed as a completely reverse activity, deprovisioning should be rather understood as a sub-activity of provisioning. In many, cases deprovisioning cannot completely reverse the provisioning outcomes (e.g. account creation and state changes), as there are often imposed certain security and audit requirements in the systems. These prerequisites might render parts of identity management operations not completely reversible, thus making deprovisioning a special instantiation of a more general provisioning process.

At the time when OASIS PSTC gathered to consider the general problem of provisioning the concept of the cloud has not been broadly developed. Hence, the term provisioning has been

initially related to the domain of identity management, as shown in the previous definition. Today, the term provisioning can be applied in different areas, such as provisioning of cloud services, provisioning of applications and other resources. Today, the use of term provisioning requires additional designation in order to avoid ambiguity. In the scope of the document, the provisioning refers to the domain of identity provisioning in the cloud, if not otherwise explicitly stated.

In order to further approach the problem of identity provisioning in the cloud, and provide detailed insight, it is necessary to define the terminology, actors and services referred in this document. The following terminology is based on work provided in SCIM specifications, which tend to focus on provisioning in the cloud environment [6]:

- *Cloud Service Provider (CSP):* entity operating a cloud service. Depending on the scenario, this entity might be the application provider (SaaS), or the underlying provider of the platform (PaaS or Iaas). In any case, the term refers to the entity the end user is facing in its interaction.

- *Enterprise Cloud Subscriber (ECS):* this term refers to the entity that subscribes to CSP services. Usually, this is the organization that consumes the services provided by CSP.

- *Cloud Service User (CSU):* real cloud service end user, the person acting on behalf of or within the domain of ECS. Typically, CSU is employee or member of ECS organization.

Based on previous definitions, the main use case assumes that CSP offers the services to its client ECS, whose employees are end-users or direct consumers (CSU) of CSP's cloud services. One variation of this use case assumes that particular ECS might consume diverse services by multiple CSP's.

Selected scenarios and approaches used to outsource the services to the cloud can be identified:

1) ECS subscribes to cloud services offered by several external CSPs
2) Organization[2] having private cloud(s) deployed at two distant facilities
3) Organizations sharing the common cloud infrastructure
4) Organizations federating own cloud infrastructure

These approaches illustrate the scenarios where employees of the organization (or ECS) consume the services provided by external entity, which is considered as a CSP. The variations of this scenario include the organizations having or sharing a separate instances of private cloud infrastructure.

2.1. Provision from the Perspectives of Consumer and Provider

In order to provide the services to its users flawlessly, many CSPs need to integrate and apply a specific system for management of identities of the users who access their services and facilities. While the authentication of end-users might be based on internal or external authentication services, the management of users' accounts, identity data and attributes itself, as well as its integration in CSP's environment, are still to be done separately. In many cases, the management of user accounts is performed locally on CSPs premises, tightly integrated into their infrastructure and processes. One of the widespread approaches to account management is instantiated in the form of a locally deployed registry of users [4]. Its role is to provide the data on users to frontend and backend applications of CSP. Due to its prevalence, this work focuses on this approach and its variations.

[2] In the scope of this document and in the context of cloud service consumer, *organization* and *ECS* are interchangeable terms

There are several reasons for establishing local registry of users at CSP's premises. Of them, most prominent include the following [4, 7]:

- Integration of IdM related functions such as authentication, authorization, federation
- Integration of billing and accounting functions[3]
- Prevalence of hybrid technologies, based on proprietary concepts or architectures
- Performance and availability related reasons
- Security, audit and conformance

Additional reasons for hosting these services locally at CSP include other technical[4], organizational, legal as well as strategic and business continuity related concerns[5].

The registry of users' accounts at the premises of CSP can contain various information on accounts of consumer organization and its members (end-users). Some of those data can be produced and processed only locally at CSP in its internal context, and as such, not synchronized with other parties. Although they might not be stored directly in the registry of user's accounts, the CSP can generate and store additional data instances related to subscriber organization or an end-user. The example of such data are billing and auditing records or specific application preferences. Considering other direction, during the consumption of CSP's service users can generate data that is to be related to the user account and kept locally at CSP.

Other data might be personally related to the user, generated externally but uploaded to CSP's promises and processed by consumed service with the user's consent. All these data instances have to be correlated to particular user account and processed accordingly to its status. As the user off-boards, specific guidelines on how to handle data related to it have to be applied. Besides relying on the internal CSP's procedures and processes, these guidelines can depend on external legal and regulatory requirements, too.

Looking from the perspective of ECSes, the setup is similar. Due to many reasons, they are even more than CSPs tight to the local deployment and maintenance of IdM system. Administration of user accounts and access rules for a wide range of applications is generally done locally, at ECS premises. Their systems typically comprise of heterogeneous constituents, integrated on principles of evolution of infrastructure and services, based on a combination of building blocks from various vendors. Such environment reflects diverse technologies and architectures, often being built upon various compromises, limitations, as well as specific technical or legacy requirements.

Increased complexity is brought be the fact that each ECS can subscribe to services of several CSPs, while each CSP can serve multiple clients (multi-tenancy). Such scenarios introduce the challenges in the domain of identity management, as the same identity data – accounts of the users – is present at numerous locations, providing different, often partial (context-related) information.

The context-dependency of users' information present or processed across various domains and use-cases should not be neglected. For instance, while ECS strive to execute comprehensive IdM, storing and processing a broad range of various data on users, CSP might be interested in the subset of that data only, extended and applied to its specific applications and environments. That data, however, has to be integrated and synchronized across the organizations, in accurate, timely

[3] In a complex environment, a CSP might need to process and analyze the information related to the end-user account, such as the consumption of resources. This is often done for the purposes of billing or accounting functions.

[4] One possible approach would be that a CSP outsources such services either to some other specialized CSP, or to consume the functionality directly provided by ECS's local user management services. Both approaches would increase the complexity of the setup and pose additional technical requirements to involved parties, leading to possible delays in provision or introducing unwanted costs and risks.

[5] Users account data represents the critical service to CSP. Without access to it, CSP is not able to provide the services to its end users, leading to serious implications on business sustainability.

manner, satisfying both the requirements of security, efficiency, privacy, and regulatory compliance.

2.2. Additional Provisioning Challenges

The provisioning of users in the cloud can be examined from several standpoints. In the previous section, the main challenge has been presented from the viewpoint of cloud users and cloud providers. This section considers the other scenarios that might render the overall use case even more complex.

Although the identity provisioning can be understood as a continuous process stretching through the lifecycle of end-users' consumption of the services run inside and outside of the domain of their organizations, the practical provisioning requirements as well as their scope might vary based on the type and complexity of the services consumed by end-users. For instance, a CSP might not need to maintain the registry of user accounts only. Depending on the type and complexity of the service offered, it might additionally need to store the supplementing information on end users as well as user privileges, permissions or roles within its service. The granularity requirement of these permissions might vary across the services of diverse CSPs, commonly being application-specific. Hence, each CSP is supposed to maintain its set and taxonomy of user entitlements, related to the application offered or local CSP environment. As there are no firmly established nor broadly adopted standards in the terms of interoperable and exchangeable entitlement schemas, the increased variety and case-specific nature of the solutions applied in the field renders the inter-organizational interoperability and synchronization as a complex issue.

The assignment of users' permissions and entitlements is assumed as responsibility of the entity consuming a particular service. In the scenario where ECS takes a role of consumer of CSP's services, it is also assumed that such assignment is to be done by the entitled staff of ECS.

The role and privilege management are some of the purposes of the IdM systems, traditionally deployed and run on the organizations' premises. Among others, their function is to enable proper automation and controllability of related processes. In the hybrid scenario, where ECS hosts some of its services internally[6] and utilizes the other services offered by external subjects in the cloud, responsibility of managing user entitlements might stretch beyond the organization boundaries.

However, it can be noted that the process of provision of user accounts encompass different activities, some of them being conducted separately or in basic use cases not at all. For instance, the creation of user accounts and their synchronisation among the boundaries of various domains can be considered as less demanding challenge as the assignment and synchronisation of users' privileges in such environment. The difference arises from the fact that the user entitlements are often tied to specific application functionalities or organization roles. As a result of that, they can consist of complex and granular structure, requiring a higher level of semantic expressiveness.

When solutions applied to maintain privileges and permissions of the users are highly complex, rendering inter-organizational interoperability as not viable option or not the first priority to solve, the process of provisioning can still be conducted automatically, focusing on the user accounts. In such case, the assignment of the privileges can be done using another channel[7]. That way, the crucial advantages of cross-domain provisioning can still be exploited by increasing the levels of security and efficiency of the processes and services across the enterprises.

2.3. Relation between SCIM and SPML

The efforts to standardize the activity of provisioning reach back to the year 2001, when OASIS established Provisioning Services Technical Committee (PSTC). The primary intention of this body

[6] Internally - within the borders of the organization
[7] Such as separate web interface of CSP

was to define a XML-based framework for exchange of user, resource and service provisioning information [8]. The first version of SPML (Service Provisioning Markup Language) was approved during 2003.

The formal and short definition of provisioning contributed by this body has been already introduced in section 2. More comprehensive definition of provisioning is included in PSTC Glossary [9] as follows:

> *The process of managing attributes and accounts within the scope of a defined business process or interaction. Provisioning an account or service may involve the Creation, modification, deletion, suspension, restoration of a defined set or accounts or attributes. The process of provisioning an account or service may involve the execution of a defined business or system process.*

The second version of SPML has been approved by OASIS in 2006. It continued on the concept already defined in the first version of the protocol. Notable change in comparison to the first version was the separation of data model from the protocol and introduced possibility to extend the protocol in guided and more flexible way. The second version of SPML further introduced two standard profiles based on XSD and DSML v2[8]. The former one focuses on the usage of XML as the data model for SPML provisioning, while the later one uses DSML for data model.

In SPML, there are three actors defined: requesting authority (RA), provisioning service point (PSP) and provisioning service target (PST). The object of their activity is provisioning service object (PSO), which basically contains the attributes of the users.

Critics of the SPML approach include its great complexity, limited flexibility and subsequent lack of interoperability as a primary factors leading to its weak adoption [7, 11]. Although it received initial support by vendors such as Oracle, IBM and Siemens, the vast majority of cloud and application vendors did not provide SPML support in their products. Some SaaS vendors have even gone a different way, proposing their own approaches and taking part in the development of other provisioning standards[9].

One of the negative points of SPML is its dependence on the IAM system's information model, which limits cross-organizational interoperability of SPML based solution. Apart from that, its intra-organizational integration capabilities are less affected by this dependence, making it a possible alternative to Enterprise Service Bus (ESB) in local deployments. Another identified disadvantage of the SPML protocol is reflected through its security model, which has been labelled as incomplete, leading to possible information leak in complex environments [7, 11 and 13].

SCIM has not been initially presented as a true competitor to SPML, as its first version has been introduced nearly nine years after SPML has been adopted in its first incarnation. At that time the interest on SPML already ceased generally.

Compared to SPML, SCIM approaches the problem differently, by focusing on practical applicability, extensibility and shorter time-to-market cycle. Use cases definition in SCIM starts with the cloud, inter-organizational applications in mind, building on technologies already pervasive and accepted in the diverse landscape of providers and applications. Cloud enterprises tend more to provide their users with management APIs that are easier to integrate in various applications. That lowers initial integration costs and barriers for the cloud users, what generally follows one of the basic principles of the cloud paradigm. Recognizing that trend, SCIM does not tend to propose a new standard with every aspect defined from the ground up. The approach of SCIM is to try fit in

[8] OASIS Directory Service Markup Standard, which provides a means for representing directory structure or queries and operations on it as a XML document [10].

[9] For instance, Salesforce contributing to SCIM and Google establishing its own Google Apps Provisioning API, now deprecated by Admin SDK.

existing environment and fill the gaps by reusing and extending what is already available and accepted in the industry.

The basis of SCIM is HTTP, using a RESTful resource-based approach and HTTP verbs, backed by JSON messages. On the other hand, SPML provides a broader basis, supporting both SOAP/HTTP and file bindings. SPML additionally supports synchronous and asynchronous provisioning[10], while SCIM focuses on synchronous operations only. However, the lightweight protocol with limited scope, as defined in initial SCIM proposal, is easier to implement and integrate on various platforms than complex, wide scoped solution. SCIM demonstrated as a promising standard already during the initial phase of the protocol definition, as the reference and open source implementations started to appear quickly. SPML, due to its scope and complexity, failed to draw the development and preparation of reference, freely available and flexible implementations.

The OASIS PSTC has been officially closed in August 2012 [14]. There are mixed opinions on SPML adoption and status. It is considered already as a legacy technology by Forrester Research, labelling it as *an effort done too early, using complex approach* [12], while others perceive it as a good framework for *intra-organizational* provisioning for new applications with the requirement for a tighter integration in the enterprise's environment than LDAP or SAML solutions might deliver [15, 7 and 13]. There are however the calls to consider SPML for future deployments, such as one provided by Cloud Security Alliance (CSA), which recommended the consideration of SPML or SCIM, depending on specific use-case and provider requirements [1].

3. Identity Provisioning using SCIM

This chapter deals with the issue of identity provision in the cloud by focusing on particular standard, currently proposed in its second iteration as IETF internet-draft. First, the scenarios and use cases identified under SCIM are presented. The introduction follows with description of common schema and API.

3.1. SCIM Scenarios and Use Cases

SCIM scenarios and use cases are provided by SCIM IETF Working Group in the level of the informational document at the time of this writing [6]. Rather than being normative, its objective is to facilitate the understanding of design and application of SCIM schema and protocol. The document additionally provides the list of requirements necessary to satisfy these considerations.

Two main use scenarios described in the specification suite focus on the process flows which involve: (1) activities performed among peering cloud service providers (CSP) and (2) activities executed between enterprise cloud subscriber (ECS) and its adjacent cloud service provider. Both perspectives consider propagation of user data to the other party based on the events (or triggers) occurring during the organizational workflow or service consumption related activities.

Such triggers are, for instance, the creation, update or removal of user identities and accounts in the system(s). They are defined, respectively, as a *Service On-Boarding Trigger*, *Service Change Trigger* and *Service Termination Trigger*. Additional trigger – *Real-Time Service Access Request* – represents a special class of activities in which operations related to user identities are initiated during SSO operational flow. That encompasses the activities such as *just-in-time provision*.

The perspective of ECS considers its local IAM system and updates on it derived from organizational lifecycle events. As they perform standard operational activities, organizations often encounter the events such as *on-boarding*, *off-boarding* or *role reassignment* of their members. Such activities imply the appropriate provision of user identities, and consequently the allocation of accounts at internal organizational facilities or external services at diverse cloud providers.

[10] Asynchronous provisioning supports execution of requests in background, time-shifted manner. It enables later retrieval of status results, while operations executed in synchronous mode return the status promptly.

Due to the complexity of the systems and roles, legal requirements or provisioning overhead, user accounts in some cases have to be provisioned before the user accesses the particular service. These cases are classified as *pre-provisioning*. In some other cases, however, due to the licensing requirements[11] or based on ephemeral accounts with low provisioning overhead, the provision of accounts for end-users is done on the basis of *just-in-time* provision.

The following scenarios between ECS and CPS are considered by SCIM use-case specifications:

- Create Identity (push)
- Update Identity (push)
- Delete Identity (push)
- SSO (pull)

The first and the second case scenarios are illustrated in Figure 1[12]. It considers one Enterprise Cloud Subscriber that faces two unrelated and unconnected cloud service providers (CSP1 and CSP2).

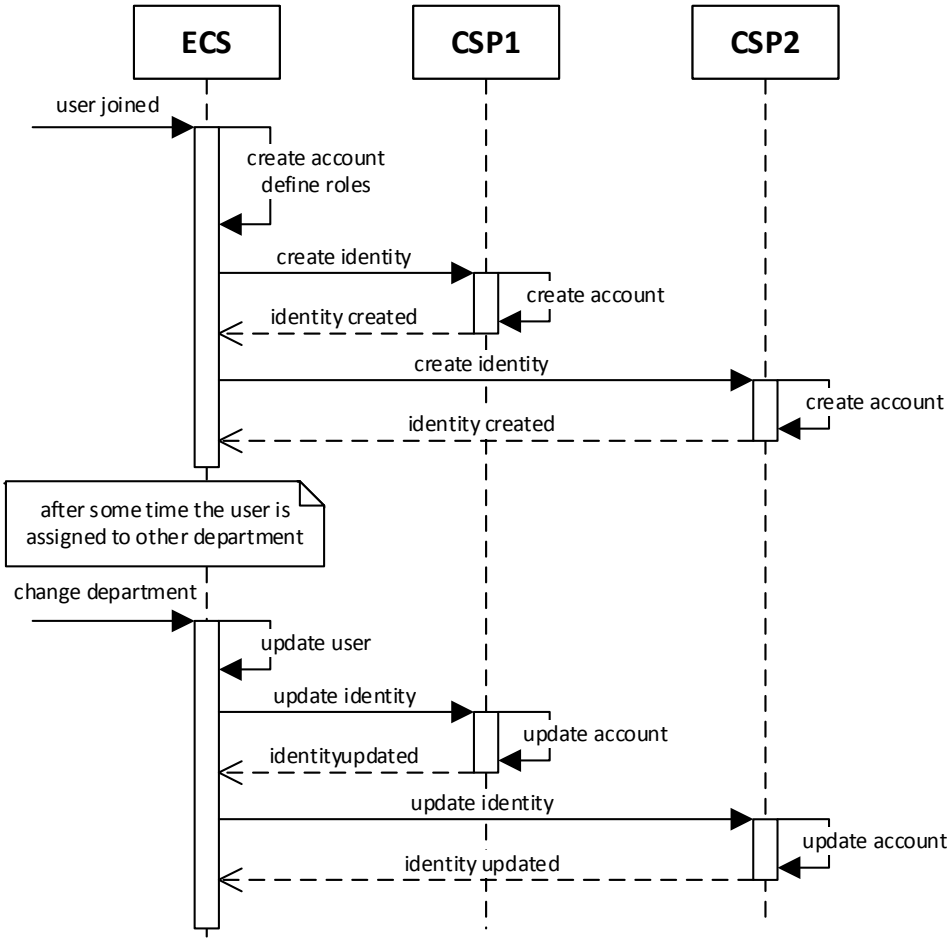

Figure 1: ECS to CSP flow

As the new user joins the organization, the provision of its account is performed internally, at organization's facilities. In the next step, the similar provisioning request has been sent separately to each CSP. By executing that request, CSPs update local user identity stores, establish user

[11] For instance, ECS is charged on the basis of number of active user accounts (CSU) at CSP facilities

[12] The figures presented in this section are delivered from [6] and serve for illustrative, non-normative purposes.

accounts and perform all other application or cloud-provider specific or prescribed activities related to account creation.

Based on that scenario, after some definite time is passed, the user has been assigned to the other department. The assignment led to the changes in its account data and the role performed in the organization. These changes are again propagated to each CSP in two separate flows.

As a result, both CSPs have synchronized their local user identity stores with the actual data provided by ECS and thus assigned the new role and related access rights to the user. Although not illustrated in the figure, the similar flow can be applied to the deletion i.e. deprovision of user accounts at both CSPs.

SSO flow (pull) considers additional scenario that presumes the provision to be done locally in the organization at the time of user's on-boarding.

The provision is however not initiated on the side of CSP. Instead, the CSP waits for the first service access of the user and performs the provision of its account at that point using pull request. This use case might be appropriate in the circumstances where (de)provisioning overhead is low or the user accesses the CSP service infrequently. It can also support the scenarios that consider cost optimizations resulting from the licensing system on the side of CSP. This way, the expenses for the creation of accounts can be lowered, if the accounts are going to be consumed at a later point in time. In some other use cases, based on the bulk creation of accounts, this approach might prevent the creation of accounts that are not going to be consumed at all.

The other perspective considered in specification suite focuses on CSP-CSP flow [6]. It starts with the assumption that each CSP maintains its local identity data store (user accounts), and performs occasional updates based on external events. The CSP-CSP scenarios involve direct exchange of user data between CSPs as a consequence of particular activities. The events can trigger the *propagation* of user identity data that activates the update of local identity stores at CSPs on a basis of *pull* or *push* requests.

The following scenarios between CPSs are considered by SCIM use-case specifications:

- Create Identity (push)
- Update Identity (push)
- Delete Identity (push)
- SSO Trigger (push)
- SSO Trigger (pull)

All these scenarios assume that CSPs have already established shared service agreements on a special basis of particular ECS tenant.

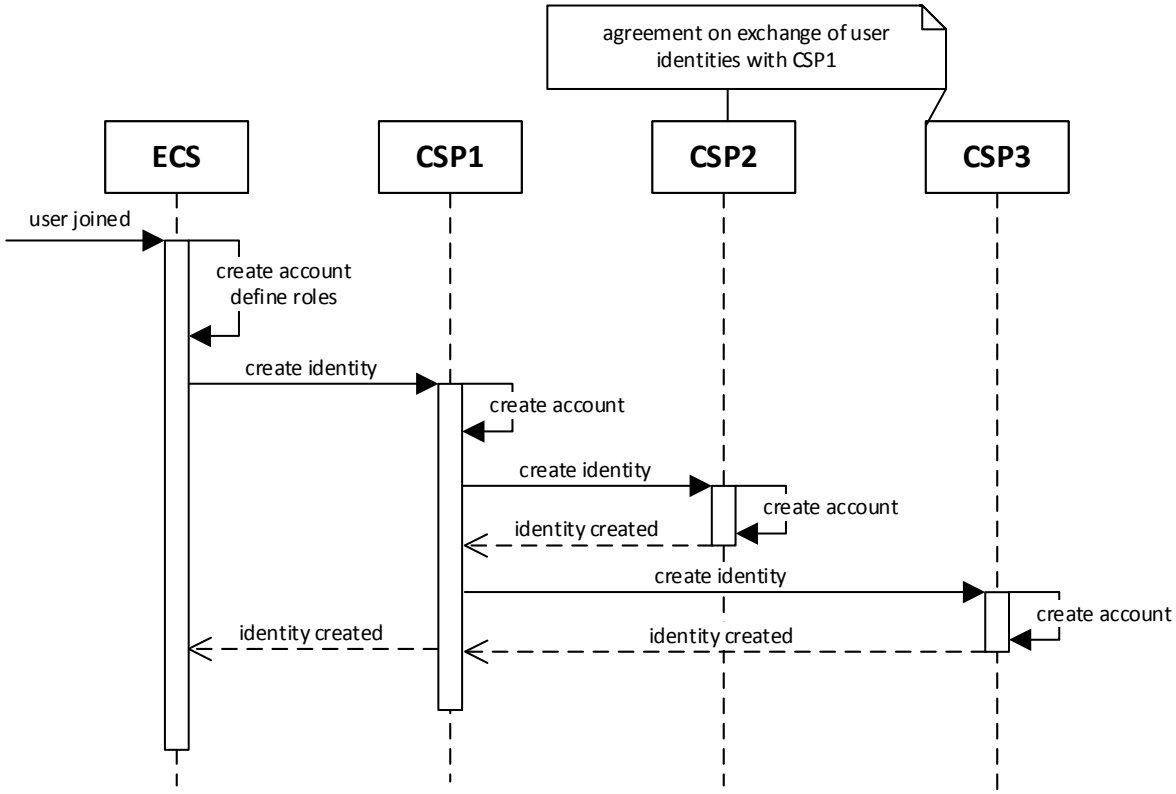

Figure 2: CSP to CSP flow

Figure 2: CSP to CSP flow describes the scenario of creating a user's account within the scope of three CSPs. They have established the shared service agreement mutually, being able to propagate SCIM flow triggers to synchronize and update user identities in their local domains. The flow represented on Figure 2 is triggered on the basis of an external request of ECS sent to CSP1. This event initiates the creation of user accounts on CSP1, CSP2 and CSP3, which both provide shared and interoperable service to ECS and its users (CSU).

This use case enables user account propagation and update in complex environments, where cloud services are provided in composite and shared manner. It supports the data exchange and service interoperability in a non-trivial way.

The next illustration, Figure 3, describes the similar flow based on SSO Trigger (pull) scenario. There, the tenant ECS initiates account provision for the user CSU at CSP1, at the time of its on-boarding in the organization. However, instead to request the account at CSP2 at the same time, the service postpones the account provisioning until the CSU accesses CSP2 service for the first time. As the first user access request is issued, the CSP2 considers its shared-service agreement with CSP1 and initiates pull request to synchronize its local identity store and provide an account for the CSU. After the provision is done, the user is informed and allowed to consume the service of CSP2.

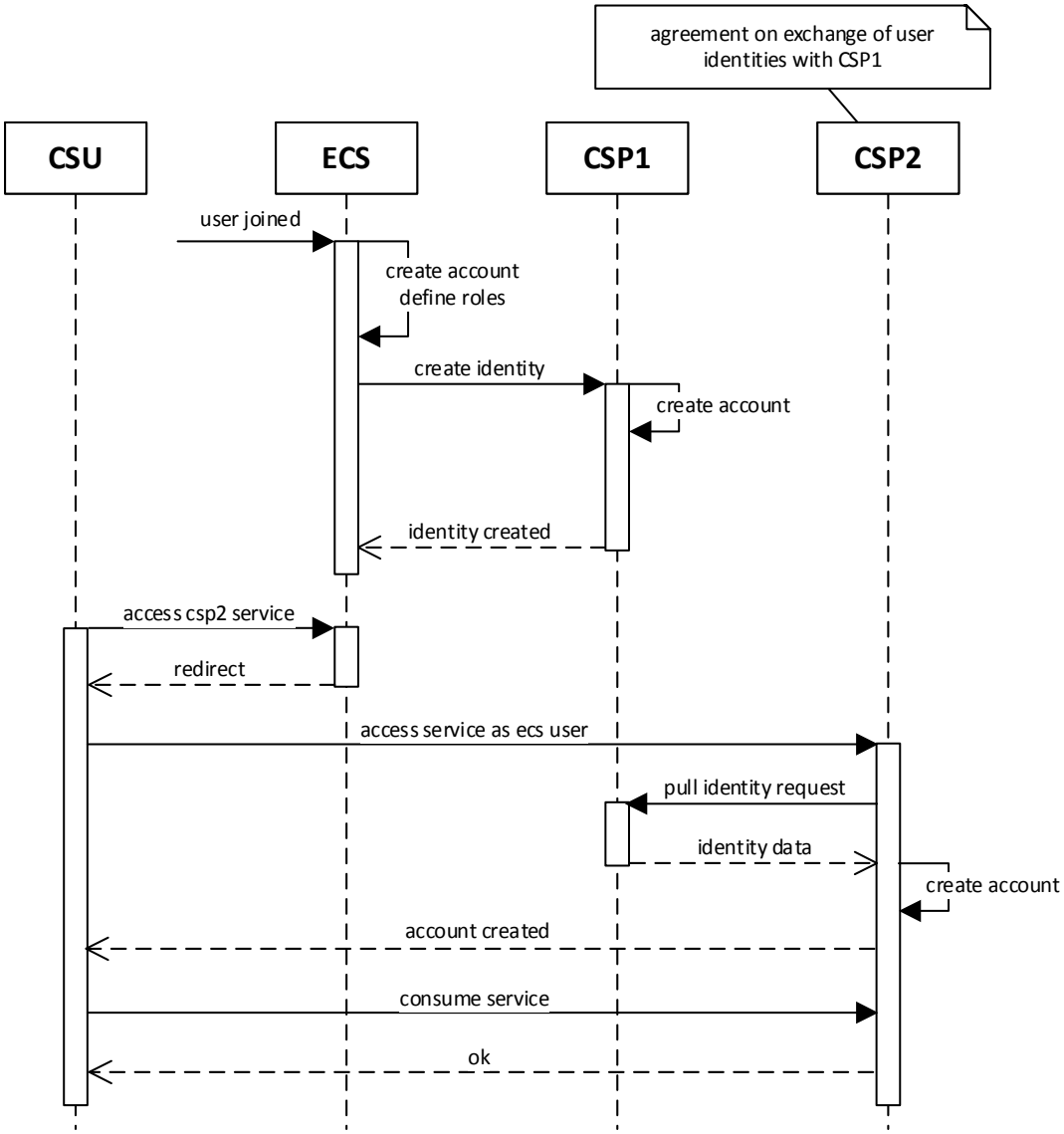

Figure 3: CSP to CSP flow with SSO pull request

This use scenario is up to the limited extent similar to SSO (pull) for ECS-CSP scenario, with the difference that it focuses on interactions on CSP-CSP level.

Based on previously described general use scenarios, the SCIM specification delivers the following representative use cases:

- Change of ownership of a file
- Migration of the identities
- Single Sign-On Service (SSO)
- Provisioning the user accounts for a Community of Interest (CoI)
- Transfer of attributes to a relying party web site
- Change notification

These use cases include the descriptions of specific pre-conditions, post-conditions and requirements. They do not represent a final set of use cases, but rather serve to illustrate the usage of SCIM schema and protocol and provide the basis for common requirements. Due to this reason, this document provides the graphical illustration of two representative use cases, while the details on all use cases are available in [6].

Figure 4 describes the change of file ownership. It is assumed that both Cloud Service Users (CSU1 and CSU2) are members of common Enterprise Cloud Subscriber (ECS) and access the service of Cloud Service Provider (CSP) in that role. Each of those users has rights to consume particular portions of CSP services. At some point in time, CSU1 leaves the organization, what leads to deprovision of its accounts and data. Instead to remove the data, ECS initiates request to assign the rights to particular data (*some_file* instance) to other employee, CSU2.

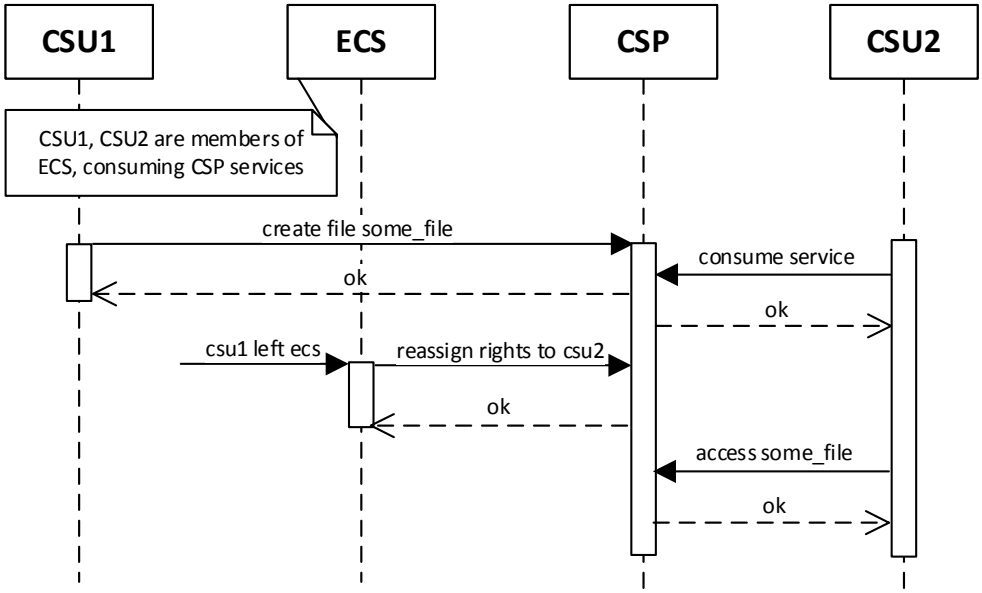

Figure 4: Change of ownership

This example demonstrates the propagation and update of access rights to particular portions of service and its data. By illustrating real-world usage example, the sample case explores the extension capability of SCIM schema, as well as role transfer capabilities of the protocol.

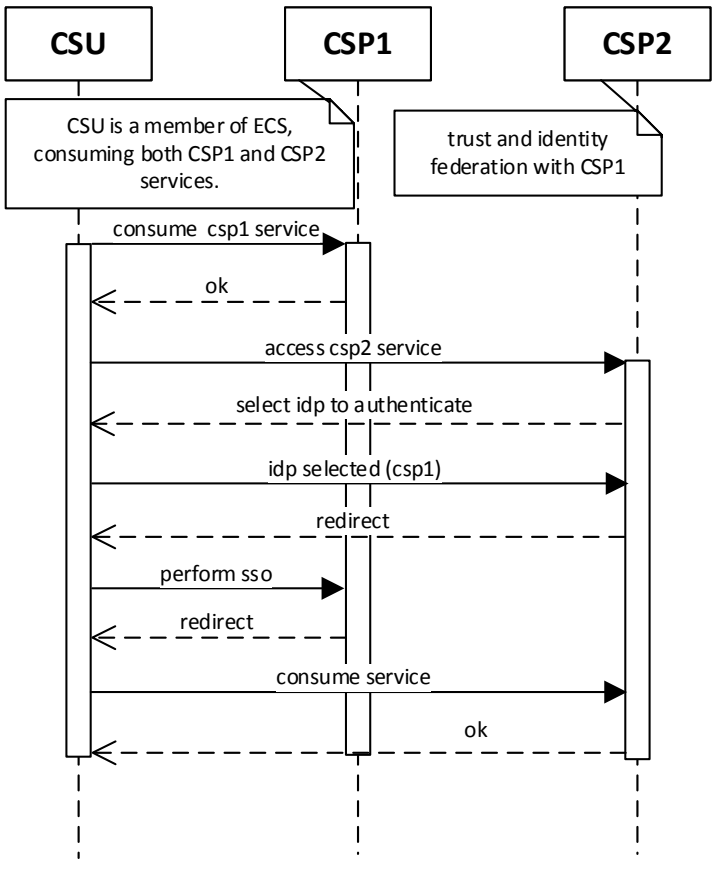

Figure 5: SSO service

The second use case considered in this section deals with the identity federation between CSPs. As described in Figure 5, it is assumed that CSP2 and CSP1 have mutual trust agreement and previously established identity federation. In presented flow, CSP1 serves as an Identity Provider for CSP2. However, as presented in SCIM protocol, this use case does not directly imply a particular technology, protocol or approach to be used for user authentication, data transfer or organization of local stores and accounts at both CSPs.

3.1. SCIM Schema

The normative draft schema for SCIM 2 is administered by IETF Networking Group and classified under standards track [16]. At the time of this writing, the schema is updated in its 14th revision, set to expire in June 2015th.

The work on version 2.0 of schema specification has been started during the August of 2012. It builds on the work previously done on SCIM 1.0 and 1.1 standards, which were released as the specifications under Open Web Foundation (OWF) in December 2011 and July 2012, respectively.

SCIM provides a minimal schema used to represent users and groups as resources, as well as to facilitate a standardized means by which the schema can be extended to define new resources.
A resource is determined as a collection of attributes identified by one or more schemas.

An attribute minimally consists of the attribute name and at least one simple or complex value. Data types used to represent attributes are derived from JSON Data Interchange format [17].

Based on that, the set of supported types for attributes includes *String*, *Boolean*, *Decimal*, *Integer*, *DateTime*, *Binary* and *Reference*. Depending on the resource definition, attributes may support singular or multi-valued representation. Additionally, they can be denoted as complex attributes based on the composition of one or more simple attributes.

The resources used in SCIM flows are represented in JSON format, where each resource object consists of the following components: *Schemas Attribute*, *ResourceType* and *Common Attributes*, as well as *Core Attributes* and *Extended Attributes*.

The role of *Schemas Attribute* is to define the URIs used to indicate the namespace of SCIM schema that describes the structure and attributes of particular object instance. The purpose of *ResourceType* is to specify the core attribute schema, possible attribute extensions and the endpoint used to access the objects of that type.

For every SCIM resource, there is a designated set of required and optional common attributes which are considered to be part of base schema for each core or extended resource, excluding server discovery endpoints such as *ServiceProviderConfig* and *ResourceType*. These attributes should not be understood as schema extensions and are not separately defined or described under the particular resource URI. This group includes three attributes: *id*, *externalId* and *meta*.

Defined as a mandatory attribute, *id* represents a read only, unique identifier for SCIM resource, as defined by service provider. Optional attribute *externalId* is a resource identifier nominated by provisioning client, with the purpose to obviate the necessity to define and maintain the mappings

between service provider id and the identifier of provisioning domain, as used locally at the premises of provisioning client.

The third common attribute, *meta*, is of complex type and optional. The purpose of this attribute is to provide meta information about related elements, such as the type of resource, its location, version as well as time of the creation and last modification.
In object's instantiation, these attributes are complemented by *Core* and *Extended Attributes*, which basically embrace top-level attributes predefined for a particular resource and the attributes as specified in the definition of extended schema resource in the latter case.

3.1.1. Common entities

The specification suite establishes the definitions of three entities that take part in common flows between SCIM endpoints. The overview of URIs of these entities is given in Table 1. There are two types of structures defined: *User* and *Group*, whereas the extension of basic *User* schema is delivered under Enterprise User Schema Extension.

In the typical case of identity provision, the primary use cases are focused on the provision of user identities across different domains. The central element of this schema defines how the *User* data is structured.

User resource	`urn:ietf:params:scim:schemas:core:2.0:User`
Enterprise User Extension	`urn:ietf:params:scim:schemas:extension:enterprise:2.0:User`
Group Resource	`urn:ietf:params:scim:schemas:core:2.0:Group`

Table 1: The overview of URIs of SCIM schema resources

Example minimal object representation of *User* resource is shown in Figure 6. The described resource consists of schemas attribute, which references the schema URI used to describe the resource type of the encompassing structure. The example object further includes the common attributes such as *id* and *meta*, and the attribute *userName*, which are specific to *User* resource type particularly.

```
{
 "schemas":[
 "urn:ietf:params:scim:schemas:core:2.0:User"
 ],
 "id":"12a5-d251-19283772123",
 "userName":"user@domain.com",
 "meta":{
 "resourceType":"User",
 "created":"2014-10-10T13:22:11Z",
 "lastModified":"2014-08-12T03:24:22Z",
 "version":"W\/\"ff789909a123e\"",
 "location":"https://domain.com/v2/Users/12a5-d251-19283772123"
 }
}
```

Figure 6: Example of minimal representation of User in SCIM

The *User* resource definition includes the range of other optional attributes of singular and multi-valued enumeration types. The list of singular attributes of *User* resource is shown in Table 2. In addition to that, Table 3 contains multivalued optional attributes for *User* resource.

Attribute	Description	Type[13]	Mutability[14]
username	Unique identifier for the user, typically used by the user to directly authenticate to the service provider. Can be displayed to the user as its unique identifier in the system.	S	RW
name	Complex attribute representing the user name. It contains components such as family name, given name, middle name, honorific prefix and suffix. Alternatively, the user name can be provided as a single sub-attribute containing fully formatted name	C	RW
displayName	The name of the user, suitable for display to end-users. The value is applied as a textual label by which the user is normally displayed by the service provider to end-users	S	RW
nickName	Casual way to address the user in real life.	S	RW
profileUrl	A fully qualified URL to a page representing the user's online profile.	R	RW
title	The user's title	S	RW
userType	Defines the relationship between user and organization, such as employee, external etc.	S	RW
preferredLanguage	Indicates the user preferred written or spoken set of languages. This attribute is primarily applied to provide localization of user interface in non-user present interaction, where HTTP Accept-Language negotiation cannot take place. Based on [18]	S	RW
locale	Indicates the user's default location for purposes of localizing items, such as currency, numerical or date-time representations. Based on [19]	S	RE
timezone	The time zone of the user, based on [20]	S	RW
active	A Boolean value that indicates the user's administrative status. The exact interpretation and usage of this attribute are left to the service provider.	B	RW
password	User's clear text password, intended to be used to specify an initial password when creating users or resetting their accounts.	S	W

Table 2: Optional singular attributes for User resource

Attribute	Description	Type	Mutability
emails	E-mail addresses of the user. Based on [21]	C	RW
phoneNumbers	Phone numbers for the user, based on [22]	C	RW
ims	Instant messaging addresses of the user	C	RW
photos	Address (URL) pointing to the photo of the user.	C	RW
addresses	A complex attribute used to provide a physical mailing address for the user.	C	RW

[13] S - string, C – complex, R – reference, B - boolean
[14] W – write, R – read only, RW – read/write

groups	Contains a list of groups the user belongs to. This attribute is indented to enable application of access control models. However, no explicit authorization model is specified; the interpretation and definition of the semantics are left to the service provider.	C	R
Entitlements	A list of entitlements for the user. There is no syntax of vocabulary specified – it is up to service providers and clients to encode a necessary information.	C	RW
Roles	The list of user roles. A role value is expected to provide a collection of entitlements.	C	RW
x509Certificates	A list of DER encoded X.509 certificates issued to the user.	C	RW

Table 3: Multi-valued optional attributes for User resource

EnterpriseUser schema extension defines additional attributes that are used to represent users that belong to or act on behalf of a legal entity, such as enterprise organization. The list of attributes of *EnterpriseUser* resource is provided in Table 4. The mutability of all the types is read-write.

Attribute	Description	Type
employeeNumber	Identifier assigned to a person, defined on the level of organization	S
costCenter	Name identifier for a cost center	S
organization	Determines the name of an organization	S
division	Specifies the name of a division	S
Department	Specifies the name of a department	S
manager	Specifies the user's manager. This attribute reflects organizational hierarchy; its value is used to reference id of other user.	C

Table 4: Attributes of EnterpriseUser resource

Group resources are used to represent the collection of users who belong to the same logical construct, such as a group. The intention of the group resource is to facilitate expression and application of common group or role based access models, without presuming or requesting the usage of some particular approach. Therefore, the definition and interpretation of semantics of group membership is left to service provider and its internal processes.

Attribute	Description	Type
displayName	Determines a human readable name for the group	S
Members	Defines a list of members of the group.	M

Table 5: The attributes of Group resource

It should be noted that attribute *members* must include values that refer to URI of a SCIM resource, such as User or another Group. Thus, this attribute supports the representation of nested groups.

3.1.1. Common definitions

Similarly as in the case of common resources, the schema defines the URIs related to the functionality of SCIM server. The list of the URIs is provided in Table 6.

Service Provider Configuration Schema	`urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig`
Resource Type Configuration Schema	`urn:ietf:params:scim:schemas:core:2.0:ResourceType`
Definitions Schema	`urn:ietf:params:scim:schemas:core:2.0:Schema`

Table 6: The overview of URIs related to SCIM server

The schema resources presented in the previous table serve to represent the configuration of the service provider, as well as to communicate supported schemas, their extensions and resource types recognized by the service provider. Using the information structured according to this representation, the clients can obtain the information on service provider capabilities and perform necessary transformations when using API endpoint.

3.2. SCIM API

SCIM API supports creation, deletion, modification, retrieval and discovery of core identity resources. The specification of SCIM API [23] defines SCIM Protocol built on top of HTTP Protocol, based on REST architectural style. As such, it represents a platform and language independent approach that leverages existing architecture and standards, facilitating transparent and efficient integration. Derived from the conventions applied in the specification suite, the terms SCIM API, REST API and SCIM HTTP Protocol can be considered as equal, referring to the same concept.

SCIM API defines the endpoints for managing resources following the core schema definition. These resources are accessed and altered by applying HTTP methods described in [24] and [25]. Table 7 provides an overview of supported methods by SCIM API and defines their general semantics, without considering particular context.

Method	Action
GET	Retrieves one or more resources, complete or partial
POST	Creates a new resource or search request
PUT	Modifies a resource by replacing existing attributes
PATCH	Modifies a resource with a set of client specified changes (partial update)
DELETE	Deletes a resource

Table 7: HTTP methods and their roles in SCIM protocol

The list of endpoints supported by SCIM API is provided in Table 8. The endpoints described in this table are derived from base URI, which most often consists of https protocol scheme, a domain name and initial path [26]. The first column of the table refers to the analogous SCIM schema resource, which corresponds to the particular API endpoint defined in the second column. Each endpoint supports a limited set of methods, which are listed in the third column for each specific endpoint context.

The SCIM requests are performed by applying the supported HTTP methods on a URL consisting of the base URL of SCIM service provider and an endpoint providing requested functionality. From eight endpoints, as presented in Table 8, five deal directly with identity and SCIM related data, while the last three endpoints facilitate the discovery of SCIM service provider features and schema.

Resource	Endpoint	Operations	Description
User	`/Users`	`GET, POST, PUT, PATCH, DELETE`	Retrieve, add and modify users
Group	`/Groups`	`GET, POST, PUT, PATCH, DELETE`	Retrieve, add and modify groups
Self	`/Me`	`GET, POST, PUT, PATCH, DELETE`	Alias for operations against a resource mapped to an authenticated subject (user)
Bulk	`/Bulk`	`POST`	Bulk updates to one or more resources
Search	`[prefix]/.search`	`POST`	Search from system root or within a resource endpoint
Service Provider Config	`/ServiceProvider`	`GET`	Retrieve the configuration of a service provider
Resource Type	`/ResourceTypes`	`GET`	Retrieve supported resource types
Schema	`/Schemas`	`GET`	Retrieve one or more supported schemas

Table 8: Overview of SCIM API endpoints

The representation of SCIM requests and responses is based on JSON structure, using UTF-8 encoding. The body of a sample request sent to SCIM service provider is shown in Figure 7. This call creates a new user at a service provider by performing POST request over */Users* endpoint. The request contains JSON representation of *User* resource, as shown in Figure 7.

```
{
 "schemas":[
 "urn:ietf:params:scim:schemas:core:2.0:User"
 ],
 "userName":"jsmith",
 "externalId":"jsmith",
 "name":{
 "formatted":"Mr. John Smith II",
 "familyName":"John",
 "givenName":"Smith"
 }
}
```

Figure 7: Sample JSON request for creation of a User resource

Figure 8 contains the body of HTTP response sent by server. The successful creation of *User* resource is confirmed with response containing HTTP status code 201 and a JSON representation of *User* resource, as shown in the figure.

```
{
 "schemas":[
 "urn:ietf:params:scim:schemas:core:2.0:User"
 ],
 "id":"45324a-4f42-1234",
 "externalId":"jsmith",
 "meta":{
 "resourceType":"User",
 "created":"2014-10-01T11:32:44.991Z",
 "lastModified":"2014-10-01T11:32:44.991Z",
 "location":"https://example.com/v2/Users/45324a-4f42-1234",
 "version":"W\/\"e220aa24b0671e2\""
 },
 "name":{
 "formatted":"Mr. John Smith II",
 "familyName":"John",
 "givenName":"Smith"
 },
 "userName":"jsmith"
}
```

Figure 8: Sample JSON response confirming creation of User resource

SCIM-API specification further extensively considers the retrieval – querying, search and filtering of resources, specifying an extended set of filtering capabilities. The details on these definitions, as well as on other specifications are available in [23].

Multi-tenancy in API specification is optional. It however recognizes four main cases:

- All clients share all resources equally – no tenancy
- Each client manipulates with a private subset of resources
- Sets of clients share sets of resources
- One client manages different subsets of resources

The specification does not prescribe the mechanisms used to agree multi-tenancy between clients and service providers or perform partitioning and associations of clients to tenants. It however provides the suggestions and considerations on how multi-tenancy can be implemented.

4. SCIM 2 Implementations

The overview page available at SimpleCloud [27] provides a list of known SCIM implementations, both for versions 1.1 and 2.0-draft of the specification suite.

The list refers to 22 implementations supporting 1.1 standard, and three implementations based on the version 2.0 of the standard, which is under active development. The summary of those implementations is given in Table 9[15].

From the summary it can be noticed that the most implementations support the version of 1.1 of SCIM. These are backed both by open source or commercial organisations, some of them include support for SCIM 1.1 in their product portfolio.

	Implementations		Supporting organizations
	Total registered	Open Source	
SCIM v1.1	22	10	Cisco, McAfee, IBM, Salesforce, Ping Identity, SailPoint, WSO2, Gluu, UnboundID
SCIM v2.0	3	3	Apache, Atlassian, OSIAM

Table 9: Summary of SCIM implementations

There are only three publicly registered implementations of 2.0-Draft, all of them in the form of open-source software. The reason behind relatively low number of implementations of the actual draft can be traced to the fact that it is still considered as the working document, published as an Internet Draft according to IETF [RFC2026]. Thus, the specifications are not stable, they are subjected to change and there are no guarantees that the draft will be elevated as a standard eventually.

4.1. OSIAM

OSIAM (Open Source Identity & Access Management)[16] is a lightweight identity store, based on open standards and intended to handle authentication and authorization processes.

The system is based on Java and published under a MIT licence. The main part of the OSIAM solution suite is an OSIAM server, which provides SCIM functionality according to draft 2.0 version. It is complemented with the modules for web-based server administration and with connector plugins, which provide integration support of OSIAM APIs for Java and Python.

As shown in Figure 9, the core component, *osiam-server*, consists of two main server components: 1) resource server and 2) authentication server. Those components can be hosted on the same or different machines. The role of the authentication server is to provide a facility to control access to the resource server, including its functionality and provided data. Access control is enforced by authenticating client and user that try to access a resource server by implementing the flow defined by OAuth 2.0 standard [18]. In this sense, the authentication server provides authentication and authorization (technical) interfaces for the users and clients involved in the flow. The other

[15] As of December 2014
[16] https://www.osiam.org

component, the resource server, provides SCIM (functional) interface, as defined in version 2.0-draft of standard and API [16, 23].

OSIAM recognizes two types of parties trying to access its functionality, according to OAuth 2.0 standard:

- Clients, as the applications or services that are attempting to access
- Users, as the entities that approve access to resources in defined scope

Both of these parties have to be registered in the server. In order to access the resource interface, client needs to be authorized. The authorization is done by the user, in standard flow. Based on OAuth 2.0, there are three grant types supported by OSIAM:

- Authorization code grant
- Resource owner password credentials grant
- Client credentials grant

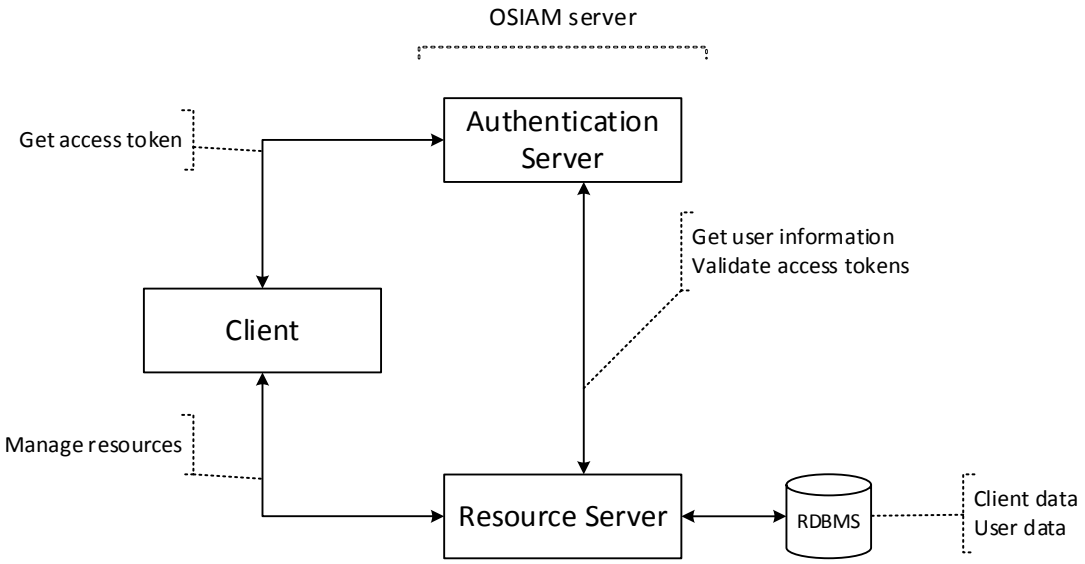

Figure 9: Architecture model of OSIAM server

The source code of the project is publicly available through GitHub account of publisher[17] and is actively maintained at the time of this writing. The latest release 1.3.2 has been published in November 2014.

Regarding future development, OSIAM focuses on three tracks in its development queue[18]:
1) cross section, 2) resource and data security and 3) authentication and authorization. In this sense, the future development will concentrate on provision of additional connectors, implementing flexible SCIM data extensions and field based encryption. Other points include development of data trail audit facility, risk based authentication and enhancements to enable UMA and OpenID Connect compatibility.

4.2. Apache

Apache eSCIMo[19] represents another SCIM 2.0-draft implementation, which is a part of larger Apache Directory Project[20]. Apache Directory Project (ADP) is an initiative established in October

[17] https://github.com/osiam
[18] https://www.osiam.org/display/ZLIintranet/Get+Started#GetStarted-What'snext?
[19] http://directory.apache.org/escimo

2002 with the aim to increase LDAP awareness and facilitate its adoption and integration in complex environments, while providing a facility for experimental and innovative capabilities.

Two main components encompassed by ADP are ApacheDS – a directory server written in Java, and Apache Directory Studio, a directory tooling platform based on Eclipse. ADP further integrates subprojects such as LDAP API, Mavibot and eSCIMo. The latter provides SCIM implementation that supports LDAP as a backend by default and can be integrated with ApacheDS.

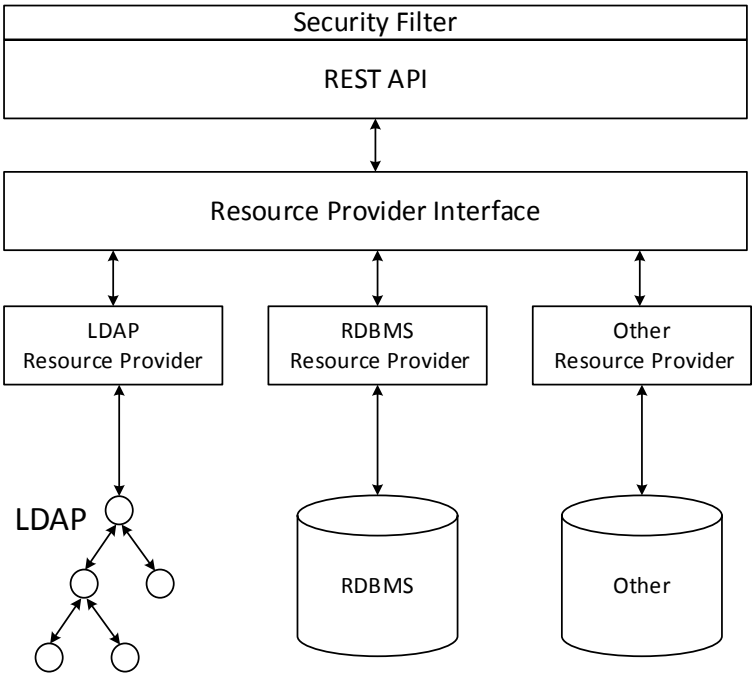

Figure 10: Architecture model of Apache eSCIMo

Apache eSCIMo is intended to simplify provisioning of identities between heterogeneous systems. It is written in Java and published under Apache 2.0 licence. Apache eSCIMo can be embedded as a component of ApacheDS or integrated with any other LDAP server as an independent application. The application aims to support HTTP Basic and Digest authentication, as well as OAuth 2 bearer token authorization [28]. Currently, it supports LDAP as a resource backend with integrated SCIM-LDAP mappings. The support for RDBMS based and other backends is planned for the future.

At the moment eSCIMo is under active development. However, the features are incomplete and the documentation both for users and developers is missing. The source code can be found in the repository of Apache Foundation[21].

[20] http://directory.apache.org
[21] http://svn.apache.org/viewvc/directory/escimo

5. Application of an e-ID in SCIM Flow

One of the cases for SCIM usage sets upon the synchronisation of directories or other user stores across the enterprise and cloud provider entities. Many of currently applied approaches in organizations still practice a simple (and less secure) password-based authentication in their workflows.

As the authentication process in many organisations is often backed by user directories and stores, stretching of organizational processes and identity management to the cloud might require controlled synchronization of cloud-based user-store with its organizational counterpart. For this purpose SCIM introduces the optional *password* field in its schema, with the primary intention to provide initial password for the user when its account is provisioned on the site of adjacent entity (e.g. cloud service provider). Similarly, the specification provides the field *x509Certificates*, which can be used to transfer the list of certificates issued to the user. This way, the ground for authentication methods beyond simple password-based authentication is enabled explicitly within specification.

The next application, authorization of the users, is covered by SCIM specification too. In the scope of the core schema, the *user* resource contains optional fields foreseen to describe *entitlements* and *roles* of the users. However, beyond defining the fields as lists, the exact type, syntax and format of both of the *entitlements* and *roles* is not provided, leaving particular specification and implementation to service providers and consumers. Additionally, the authorization can be enforced through the *group* memberships, too.

Being technology and approach-agnostic, SCIM standard does not prescribe exact mechanisms and methods to be applied for authentication or authorization of end-users. It is namely left on the particular use case and implementation to set up and communicate suitable mechanisms across the entities. From this point, the integration of eID into SCIM workflow requires extension of the schema in order to deliver the capabilities needed to communicate specifics and execute flows related to eID-based authentication flow. The base is already offered by SCIM specification, as it allows for extensions to be implemented.

5.1. Establishing the Scope of an eID

Electronic identity (eID) is represented with a collection of digital information that is associated with an entity. It can be used to ensure unambiguous authentication of a person in online transactions. However, the level of that unambiguousness is related to the *assurance level* of *electronic identification*, which corresponds to the degree of confidence in establishing the identity of a person.

The legal basis for eID in European Union is Regulation N^{o} 910/2014 (eIDAS), which recognizes electronic *identification as the process of using person identification data in electronic form uniquely representing either a natural or legal person, or a natural person representing a legal person.*
The regulation, in addition, recognizes authentication as an *electronic process that enables identification of persons, or the origin and integrity of electronic data*, by using *electronic identification means* [29].

An important role in the process of electronic identification is given to *identity intermediaries*[22], the entities that assure for identity information provided during the execution of electronic identification processes. Zarsky and Andrade distinguish between hard-eID and soft-eID intermediaries [30].
The former ones are referred as traditional intermediaries subjected to legal frameworks, which authenticate identity upon its issuance by applying structured and legally recognized procedures. The examples of such electronic identities are national eIDs, which provide robust and reliable e-identities to citizens. Those electronic identities can then be used in online procedures under clear

[22] In the literature also referenced as *Identity Providers (IdP)*

legal and formal basis, enabling and facilitating the accomplishment of various procedures and interactions electronically.

The entities identified as soft eIDs intermediaries are, however, perceived as subjects that provide less confident and structured identity assertions under relaxed legal environment, if any. Zarsky and Andrade characterize them by three central traits:

- Identification process is incidental to the company's business plan,
- Initial and subsequent identification and verification process are carried out remotely, and
- The eID intermediary is operating in lightly regulated settings.

The examples of those intermediaries are online services such as Facebook, Twitter or Amazon, which base their business models on user identities and enable other parties to consume or integrate user information. The information on electronic identity provided by soft-eID intermediaries is often based on non-verified user[23] identity claims. Even the provision of eID data can be incidental to the online process, which can be focusing on authorization flow and not the authentication itself. Example for such process is OAuth 2.0 flow, frequently integrated with online services, which primarily serves to provide user consent for third party to access its online data or resources.

5.2. eID in National and International Contexts

During the previous decade many countries have rolled out national eID facilities and integrated them in domestic online landscape. According to Acuity Market Intelligence, in the year 2013 there were more countries in the world issuing national eIDs than ones providing their citizens with traditional means of national IDs [31]. Their market research predicts national eIDs to be issued by 127 countries worldwide by 2018, covering nearly the half of the world's population. At that time most of the European countries are expected to be covered with national eIDs.

The initial process of establishment and development of eIDs in the context of European states has been analysed by Arora in 2008 [32]. The research focused further on motivations, risks and challenges arisen in the eID implementation phase. A couple of years later the European eID landscape has grown and now it consists of a diverse range of national eIDs that are realized by using various approaches.

As the pace of proliferation of online technology in everyday lives and processes increased its pace, so has the usage and application of national eIDs evolved and enlarged its reach. Consequently, the national perspective and usage of eIDs have been complemented with the international context, posing additional requirements and leading to the new challenges. From that point, the project STORK[24] has been initiated and backed by EU ICT Policy Support Programme, with the aim to establish European eID interoperability platform and enable citizens to use their national eIDs in a cross-border environment. The initial STORK project has been run in the period from 2008 to 2011, while its successor, STORK 2.0, has been planned to be executed from 2013 to 2015. The aim of STORK 2.0[25] is to leverage and extend the results of STORK by establishing interoperability of different national and international approaches, including eID for persons, legal entities and mandate capability. The up-to-date list of national eIDs is provided in the report on STORK 2.0 Member States eIDs [33].

Focusing on national perspective, the activities to establish national eIDs in Austria have been started already in the year 2000, when the Austrian Cabinet Council decided to amend the planned health insurance card with qualified electronic signature and eID functions. The adoption of eGovernment Act [34] in 2004 provided the legal framework for the identity management system, as well as establishment and integration of eGovernment services. Following a technology-neutral

[23] Subject of identification
[24] http://www.eid-stork.eu
[25] http://www.eid-stork2.eu

model, first eIDs were issued in 2003, mass rollouts followed in 2005 (bank cards, heath insurance card, mobile ID). The technology-neutrality providing the basis for nearly 100% coverage of eID tokens end of 2005 [35], their activation is however voluntary (when providing this study, about 650k active eIDs were in use).

The integration of national eID in Austria is done by employing various protocols and software components. The interactions between components involved in the process of citizen authentication are shown on Figure 11.

The principal components interacting in this workflow are the Citizen Card Environment, MOA-ID, and the Online Application (Service Provider). The Citizen Card Environment is the software or service that provides the functions of the Citizen Card, which might be run locally in the citizen's system or accessed as a remote service. MOA-ID is a suite of open-source software components that enable and simplify the integration of citizen identification and authentication processes in online applications [36, 37].

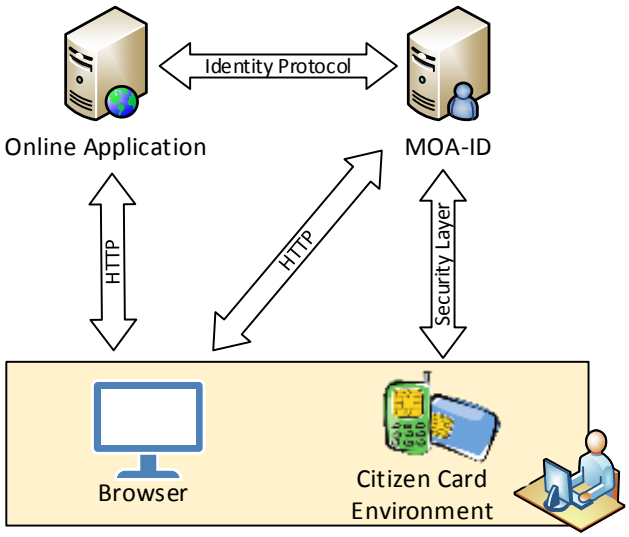

Figure 11: Authenticating citizen using MOA-ID

In the workflow shown in Figure 11, the citizen tries to access the online service using its personal citizen card. The online service leverages the functionality of MOA-ID and initiates the authentication workflow. The workflow further involves the communication between MOA-ID and the Citizen Card Environment, eventually providing online service with authentication data of the citizen. Those data include personal data of the citizen and additional information relevant to authentication process.

The technical basis of this structured process is SAML 2.0 [38] based on an authorisation federation protocol referred to as PVP[26] (portal group protocol) [39], which is used in the communication between online application and MOA-ID instance on Figure 11. PVP was derived as a deployment profile combining the STORK profile and the Kantara eGov 2.0 Implementation Profile [40], PVP follows standards and provides an interoperable approach for federated identity management, allowing government organizations to extend the reach and application of their internal user and access control facilities to use cases involving external applications (federated authorisation). The complete list of supported attributes in PVP 2.1 profile is provided in [41]. The functionality of citizen card environment is accessed by using the so-called Security Layer interface [42], which employs the Security Layer transport protocol [43].

[26] *Portalverbundprotokoll* (German) or *portal group protocol* (English)

Figure 12 shows a simplified and adapted excerpt of a sample response derived as a part of PVP 2 communication flow. The response contains the attributes corresponding to the family name (PRINCIPAL-NAME), given name (GIVEN-NAME), as well as the attributes containing the citizen's sector-specific personal identifier[27] (BPK) and the sector involved in the data processing (EID-SECTOR-FOR-IDENTIFIER). As a consequence, in the Austrian case, the citizen's personal identifier is *unique* and *non-reversible* for each public sector authority involved in the transaction, according to the predefined sector coverages. Additionally, the personal identifier used in the transactions is unique for every of the entities involved in transactions that do not belong to the public sector. Every entity[28] that takes part in the authentication process thus receives citizen's personal identifier specific and unique to that entity's sector (field of state activity), making it practically unviable to trace the citizen's e-activities across the sector-scopes of different subjects.

```
...
<saml2:AttributeStatement>
<saml2:Attribute FriendlyName="PRINCIPAL-NAME" Name="urn:oid:1.2.40.0.10.2.1.1.261.20"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
<saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">Mustermann</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute FriendlyName="GIVEN-NAME" Name="urn:oid:2.5.4.42"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
<saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">Max</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute FriendlyName="BPK" Name="urn:oid:1.2.40.0.10.2.1.1.149"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
<saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">EA:fkK+ZDGFNrasdfsWdsnS4fkt5Yc=</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute FriendlyName="EID-SECTOR-FOR-IDENTIFIER"
Name="urn:oid:1.2.40.0.10.2.1.1.261.34"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
<saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">urn:publicid:gv.at:cdid+EA</saml2:AttributeValue>
</saml2:Attribute>
</saml2:AttributeStatement>
...
```

Figure 12: Sample PVP 2 response, excerpt

The attributes listed in Figure 12 can be for this reason considered as the ones that enable the process of the eID application under the scope of SCIM use cases and Austrian jurisdiction.

[27] *ssPIN* (English) or *BPK* (German)
[28] Such as association or company

Figure 13 additionally shows the analogous SAML response provided under the cross-border authentication workflow done under STORK 2.0 framework. The excerpt has been simplified and adapted for the purpose of readability.

```
...
<saml2:AttributeStatement><saml2:Attribute Name="http://www.stork.gov.eu/1.0/eIdentifier"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
stork:AttributeStatus="Available"><saml2:AttributeValue
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:anyType">AT/IT/s9xsEadkDCWXDhiUN95xfsXPT0Y=</saml2:AttributeValue></saml2:Attribute>
<saml2:Attribute Name="http://www.stork.gov.eu/1.0/givenName"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
stork:AttributeStatus="Available"><saml2:AttributeValue
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:anyType">XXXÉliás</saml2:AttributeValue></saml2:Attribute><saml2:Attribute
Name="http://www.stork.gov.eu/1.0/dateOfBirth" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" stork:AttributeStatus="Available"><saml2:AttributeValue
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:anyType">1955-10-
11</saml2:AttributeValue></saml2:Attribute><saml2:Attribute
Name="http://www.stork.gov.eu/1.0/eMail" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" stork:AttributeStatus="NotAvailable"/><saml<saml2:Attribute
Name="http://www.stork.gov.eu/1.0/surname" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" stork:AttributeStatus="Available"><saml2:AttributeValue
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:anyType">XXXTörőcsik</saml2:AttributeValue></saml2:Attribute></saml2:AttributeStatem
ent>
...
```

Figure 13: Sample SAML response in STORK 2 environment, excerpt

As the STORK framework is applied on the top of national infrastructure, the user attributes provided under the STORK framework are comparable to the ones provided in Figure 12. It should be noted, however, that the personal identifier[29] privacy and non-reversibility requirements stretch through the international context too. In the current specification and implementation of STORK, personal identifier is specific and unique at least for each particular country where the authentication data has to be provided [44]. The uniqueness of the personal identifier at deeper levels, such as the level of a particular application in the destination country, is dependent on the receiving country [44].

Based on the information presented in this section, it can be concluded that one of the building blocks used in the authentication processes that rely on national eIDs in national and international context, is represented by the personal identifier, or e-identifier. Some of the requirements for the generation of this identifier are non-reversibility and non-traceability across different subjects, which is enforced through sector-specific uniqueness of personal identifier.

[29] Attribute *eIdentifier* in the context of STORK 2.0

5.3. Integrating eID into SCIM flow

This section considers a possible integration approach of eID in SCIM flow. For the purpose of integration of eID in the SCIM flow, based on the definitions presented in section 2, three actors are identified as relevant. The scope of action hence includes Cloud Service Provider, Enterprise Cloud Subscriber (ECS) and Cloud Service User (CSU), which has a specific relationship with ECS (e.g. being the employee).

Figure 14 provides a simplified and generalized overview of interactions of these entities, when ECS consumes CSP services.

The ECS as shown in Figure 14 maintains local user store, which contains user login and passwords, other user data, and eID related user data. That information can include the reference to eID providers, protocols and identifiers used to identify the user.

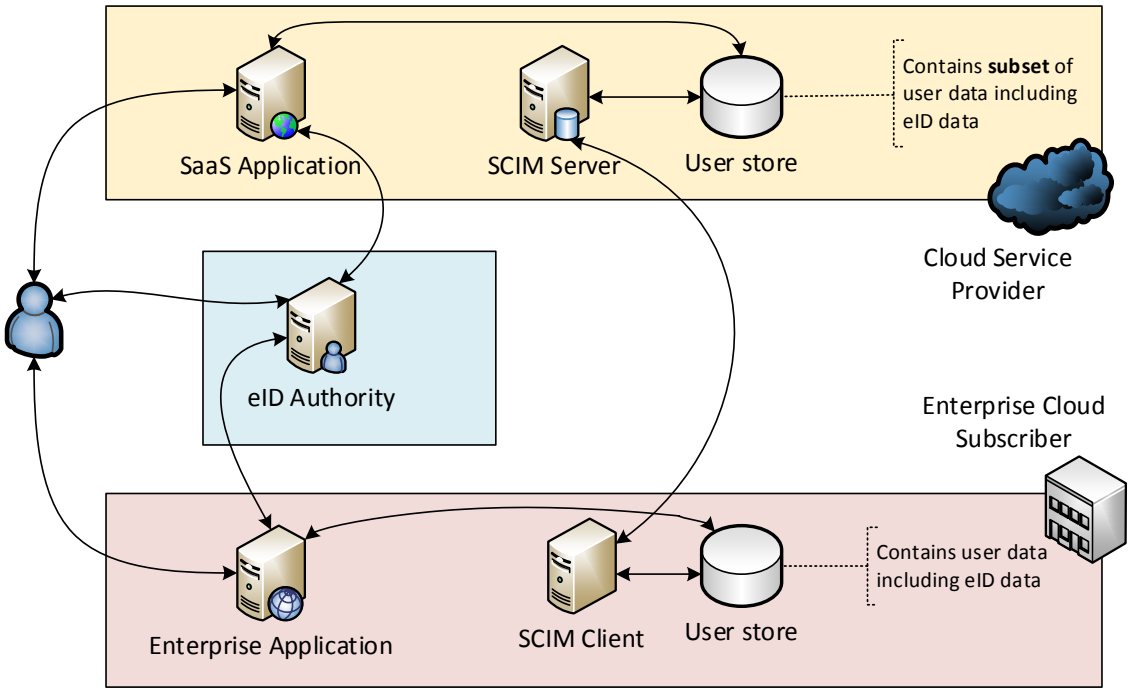

Figure 14: Overview of interactions when using eID with SCIM

The first requirement in this process is the application of user identity data – and namely eID – within the organization, and on the site of CSP for the purpose of organizational workflows. The SCIM workflow is applied for the purpose of synchronization of the identities between organizations. The synchronization on ECS side is initiated by the client, which connects to the SCIM server instance located in adjacent organization. It should be noted that the SCIM workflow can be bidirectional or involve multiple stakeholders, including other cloud providers. That depends on the use case, as described in section 3.1. The considerations in scope of this section hence focus on unidirectional flow, that the user store is maintained and updated only on the premises of the enterprise organization.

The SCIM client on ECS connecting to SCIM server on the side of CSP does not necessarily perform blind copies and updates of server identity store. The user information kept in local user store might be *filtered*, *transformed* and *processed* for particular CSP prior to its transfer using API methods. The transfer (synchronization) is usually triggered by the event related to identity management in the organization. The scope of this processing is left to internal organization and policies of ECS.

Being neutral in the terms of technology and procedures applied to the authentication, SCIM schema does not predefine the authentication means beyond the simplest user-password based approach. From that point, the integration of eID based flow requires an extension of SCIM schema. This is possible according to the section 3.3 of SCIM Core Draft [16], which considers the attribute extensions to resources.

There are namely two changes that need to be performed. First, the user resource, as defined by core schema specification, should be extended in order to provide information on user identifier data and attributes, as obtained by using external eID provider[30]. In addition, it will be necessary to introduce a new resource type in order to relate previously announced user identifier[31] with particular eID provider and protocol. This is also possible according to the section 3.2 of SCIM Core Draft [16]. The new resource type is necessary to enable storage and description of the particular eID provider and its supported capabilities.

By querying the SCIM server API endpoint and applying the methods described in Table 8, ECSs will be able to retrieve the list of eID providers supported by the CSP, and based on the information retrieved, appropriately process and provide user data to CSP. This way, ECS can process, transform and update the users' identities on the side of CSPs, for each particular CSP if necessary.

Back to the Figure 14, considering the previous extension of SCIM *user resource* and the introduction of new *eID provider resource*, the following sample workflow would be imaginable:

- User (CSU) requires to consume SaaS Application (CSP). The login has to be done using external eID.
- The user accesses SaaS's front-end and chooses eID login option.
- After selection among supported providers is done, the user gets forwarded to the selected provider and performs the authentication using its eID.
- After successful authentication is done, the user gets forwarded back to the SaaS and is able to consume the services.

It should be noted that, in the later step, user might be presented with the list of different accounts manageable on SaaS premises under its eID. These accounts can be related to several ECSs with which the user maintains an active relationship. This is possible as CSU's e-identifier might (or might not)[32] be available in the same form for SaaS and its tenant users. As tenants understood are different ECSs, each maintaining relationship with the particular CSU. This way, the user might use one eID to access the CSP accounts for several organizations (ECSs).

5.4. Challenges and Approaches

The integration approach presented in the previous section considers eID in the terms of generic eID intermediary (IdP). However, depending on the required assurance level, the technology applied and the integration complexity, the IdPs can be classified into soft-eID and hard-eID categories, as described by Zarsky and Andrade [30].

Soft-eID providers are external intermediaries, such as the online services that provide third-party integration for authentication purposes. Examples of such services are Facebook, Google, Amazon and others. As identification and authentication are incidental to their business core, they usually provide less confident identity assertions with limited application for enterprise environments. However, the integration of soft-eIDs is less demanding and straightforward.

[30] Identity provider

[31] Personal identifier or e-Identifier

[32] The form and value of the e-Identifier for the ECS, CSP and its tenants (several ECSes) might be subject to data protection regulations, depending on the jurisdiction and type of the authentication workflow involved in the process. Thus, obtained e-identifier might be different for each of the tenant accounts.

National eID platforms, classified as hard-eIDs, are built exactly for the purpose of identification and authentication that rely on high assurance levels. As such, those entities, as well as their operation, are subjected to additional legal terms and frameworks. The integration of their services with SCIM landscape might require more effort and include additional challenges due to the legal frameworks that drive their application.

The integration of both soft and hard eID approaches require extension of SCIM schema and API in order to enable the possibility to specify external eID provider to be used for user authentication, as envisaged in section 5.3. The exact authentication flow can be based on OpenID Connect, OAuth, SAML or another protocol in both cases.

In the terms of soft-eID, considering section 5.3 and the overview provided in Figure 14, the integration with CSPs can be done by involving common protocols such as *OpenID Connect*, *OAuth* or *SAML*. Even the ECS can be positioned in the role of soft-eID by exposing its interface to external clients for the identity integration purposes. Such integration has the potential to provide further application of SSO between parties. That is however out of the scope of this document.
Considering the integration of hard-eID, the approach used for integration might be a bit more sophisticated than the one applied in the case of soft-eID. Besides the requirement to extend SCIM schema and API, the integration of national eID infrastructure poses additional technical and legal requirements. It should be however noted that the integration of a national eID, and on national level, is specific particularly for each state considered, as the implementations and the approaches used vary among the countries. The approach realized in Austria might not apply in some other country; it might demand less or more technical or legal requirements.

The integration of eIDs on an international level, however, might pose less complexity and diversity. A recently introduced eIDAS regulation [29] promises to deliver the harmonization both in technical and legal aspects in the coming years by providing a higher level of interoperability and unification of eID solutions across the EU. From that point, the integration of national (official) eIDs and their application in cross-border (EU) context is expected to be simplified.

5.5. Integration of Austrian eID – Use Case

This section considers possible integration of eID into SCIM processes on the case of Austrian national eID. In this case and from the technical perspective, the integration would include the deployment of MOA-ID software component, serving as an intermediary used in the authentication process, as described in section 5.2. A MOA-ID instance can be hosted on the sites of CSP, ECS, or by third-party, providing that it is previously configured to allow access to the CSP. In both cases, CSP should be able to connect to MOA-ID instance by initiating SAML authentication request[33] or by applying OpenID Connect authentication flow. It should be noted that the usage of MOA-ID instance is not obligatory – the integration of eID can be done without it, but the overhead involved in the integration process might be higher.

The integration of Austrian national eID in this process raises the necessity of *identity reconciliation*. As shown in the Figure 12, the *eIdentifier*[34] of the user is provided in *sector-specific form*. This means that derived eIdentifier is unique for each distinct organizational sector (for public authorities) or for each distinct organisation entity (for private sector), including ECS and CSP. If the system would be integrated to rely on eIdentifier only, then the user's eIdentifiers for ECS and CSP have to be correlated prior or during the process of identity provision. This can be done on ECS premises, but would require user's consent. In addition, it should be considered that the Austrian eGovernment Law forbids the storage of user eIdentifiers generated for other sectors[35]. The conformant solution would require either to store the encrypted foreign-sector eIdentifier on

[33] Using PVP 2.1 protocol, which is based on SAML 2.0
[34] *BPK* attribute, while the sector is denoted with *EID-SECTOR-FOR-IDENTIFIER*
[35] Article 14 (2) eGovernment Law (BGBl. I Nr. 10/2004)

ECS premises, or to perform additional correlation with CSP's stored eIdentifier by employing other references[36].

Figure 13 shows an excerpt of the authentication response transmitted in the international context. In this example Austrian citizen is authenticated at Italian service. Based on the current approach applied in STORK project, the foreign country is considered as one sector for eIdentifier. Thus, the eIdentifiers derived for different subjects (online applications) located in some foreign country have the same values. Although a bit less demanding, this approach still requires identity reconciliation for the level of each CSP's country. This comes from the fact that the ECS located in Austria gets and stores user's eIdentifier derived for Austria, while the foreign CSPs receive and store user's eIdentifier specific to the location of particular CSP.

5.6. Integration of Authorization Capabilities

Going beyond the account maintenance and authentication of users, the schema proposed by SCIM 2.0-draft encompass the elements that can be employed to update and synchronize the authorization data. The fields defined for that purpose are *entitlements* and *roles,* under *User* resource. In addition, the authors explicitly notice that the authorization can be enforced on limited degree by using group memberships.

It should be noted that SCIM schema does not provide the exact object types, formats, or standards to be used when communicating user entitlements and roles. Only formal definition provided in the document requires that both attributes are represented as a lists. The format of that representation, vocabulary used and canonical types are left to the service providers and consumers to define.

This requirement is partially clear, having in mind that there are no widely accepted standards or approaches to exchange authorization information in the cross-organizational context, beyond the ones that use simple string notations of roles and entitlements understandable and specific for the particular purpose only. This however renders limited applicability in the broader context by requiring to apply semantic which is not defined, accessible or understandable in the form appropriate for automatic processing.

Comparing with the approach applied in the Austrian eGovernment[37] [41] is generally compatible with the technique proposed with SCIM *roles* attribute.

However, the lack of semantic expressivity limits the integration of partially structured roles and entitlements to organization and use-case bound applications. To attain the broader interoperability, it is necessary to go beyond simple string based role representations without deeper semantics behind them. The other authorization approaches should be considered too, such as attribute-based authorization, which is not supported out of the box.

As other authorization approach gain attention, such as XACML [46], their integration in SCIM flow should be further considered. The integration of other authorization techniques and workflows would provide benefits to users and organizations by enabling more complex authorization scenarios.

5.7. Privacy, Identity Provisioning and the Cloud

The topic of the privacy is gaining a great attention in recent years. As the cloud paradigm introduced complex environment that includes cross-organizational and cross-jurisdictional interactions and data exchange, it is of great importance to consider privacy in such environment.
Based on the analysis of the SCIM approach, it can be stated that the privacy was not of the highest concern during the design of the approach. It is basically left to the implementations to consider

[36] E.g. *externalId* user attribute of SCIM schema
[37] X-PVP-ROLES defined under Authorization section of [41]

appropriate privacy and data protection models. The section 9 of the schema however discusses privacy issues and provides a suggestions on that matter [16].

Storing the data on employees, such as name, address, manager, department employee number and so on might raise privacy concerns, especially when interactions with cloud provider go beyond the borders of jurisdiction. However, the data subset to be stored at CSP facilities in SCIM flow is optional to the greatest extent. Additional resources, or resource extensions that employ cryptographic functionality can be defined for the purpose of elevated privacy, but it is left to enterprise user and cloud service provider to provide an appropriate schema and its implementation.

It is, therefore, possible to provide minimal and anonymized user data to the cloud service provider. It should be further noted that, as currently many organizations use cloud services in the way that user information is manually and voluntary provided[38], the existing approach to user privacy is even less appropriate. By employing structured identity provisioning process, the issue of privacy and the control over provided data can be brought to the higher level.

5.8. Considering eID Adoption for CSP and ECS

Going further from basic SCIM flow that involves plain authentication based on username and password, the integration of eID in the user stores both at the sides of CSP and ECS would bring various benefits. This section provides several points necessary to be included in consideration.

High assurance
National eID solutions provide for different assurance levels. The eID framework for EU regulates the scope and state of liabilities and assurances upon which eID intermediaries operate[39]. The assurance levels clearly defined under legal frameworks enable the application of authentication among ECS and CSP subjects by leveraging existing legal frameworks. By using high-assurance eIDs the subjects are not required to take part in costly processes, such as verifying identities and correctness of data of users involved in the transactions. Depending on their required security levels and business requirements, organizations can select appropriate assurance level without the need to reinvent or re-implement existing technologies and approaches.

Liability
Existing regulations, as well as the regulations going to be fully implemented, provide the framework for organizations to manage the risk by leveraging liability guarantees of third parties, including eID intermediaries involved in the authentication processes. Operating under such framework enables the organizations to lower their costs by reducing or dislocating the risks related to guarantees and liabilities.

Security and conformance
Developing and running own sensitive infrastructure requires a lot of resources in order to be fully conformant to security and legal requirements posed during the whole solution lifecycle. By using external authentication infrastructures, the organizations are able to ensure conformance in a less demanding and a less complex way. The damage caused by security breaches in organizational premises can be reduced by using external infrastructure. However, the dependence on external eID providers might open new attack and risk surface for the organizations.

Focusing on the primary business case
Organizations that adopt high-assurance eIDs are not required to dedicate their resources to development and maintenance of their local authentication and integration infrastructure in full extent. The requirement to be conformant with various technologies in a fragmented market and dynamic landscape is less pronounced when such activity is outsourced to third parties. By reusing external infrastructures, the organizations can relocate their existing resources and focus them on

[38] E.g. by manually administering organizational user accounts using CSP's configuration interface
[39] Defined as *trusted service provider*

primary business activities. The integration of external solutions, however, requires to consider and manage related risks.

Market and use-case reach
By exploiting network effect and proliferation of long-term solutions, organizations can develop additional or extend existing use cases and allow for wider market reach. Following the example given in section 5.3, the integration of existing eID solution can enable users to control several accounts at one CSP by using one eID. Previous solution, based on username-password combinations, would require for CSUs to maintain credentials for each ECS or tenant organization they are working with.

Reducing costs and complexity
Considering previous points, both CSP and ECS organizations benefit from eID integration in their workflows. The complexity is reduced by using existing, widely accepted and standardized solutions under definite risk and liability frameworks. The costs reductions are attainable through shorter time-to-market requirements and lessened investment demands necessary to reach security requirements for successful deployment.

Managing identity lifecycle independently
By using external eID, user's identity and authentication data are decoupled both from ECS and CSP. Consequently, the management of this data is done outside of the premises of involved parties. The user's e-identity defined and managed at ECS or CSP premise is generally active during the lifecycle of the project of that entity – as such it can be considered as *ephemeral* and *organization-bound*. On the other hand, user's e-identity provided by external eID intermediary might go well beyond the lifecycle of both ECS and CSPs. That approach provides a more stable and permanent solution that serves as an enabler for further integration cases.

6. Conclusion

In this document the problem of identity provisioning is approached both from perspectives of cloud usage and application of eIDs. After introducing the topic of provisioning, including standards, scenarios and use-cases particularly related to the cloud, the document provided the short overview of SCIM schema and API.

SCIM is recently proposed approach that focuses on cloud-use cases, with the aim to provide an interoperable and solution for simple and quick integration of cloud platforms and applications into organizational identity management environment.

According to the registered list[40], there are 25 implementations of SCIM, many of them being published under an open-source licence. The focus of this analysis was, however, on the SCIM 2 standard, proposed as a mature IETF internet draft under the standards track. Although there are no revolutionary changes observed between SCIM 1.1 and 2.0-draft, the second iteration provides more stable version that is improved according to the inputs and experiences obtained from the version 1.1 already in use in many environments.

Focusing on Austrian eID implementation, and considering the cross-border approach based on the STORK 2.0 project, this document further analyses the advantages, potential and obstacles to apply eID in a cloud identity provisioning workflow using SCIM. Although the integration of eID in SCIM workflow is possible, it would require the extension of SCIM schema. Additional issue might be due to the fact that eIdentifier of the users is sector-based and as such requires correlation to be done prior to the provisioning process. The approach is analysed by deriving national-specific use case and considering international context based on STORK project approach.

This work further insects the capabilities to complement basic identity management with the cross-organizational exchange of authorization data. The work further provides a short consideration of

[40] http://www.simplecloud.info

privacy issues in identity provisioning. Finally, the analysis provided in this document ends with the overview of the benefits and risks that arise from potential integration of eID and cloud identity provisioning.

7. Literature

[1] Hoff, Chriss, at al. "*Security guidance for critical areas of focus in cloud computing v3.0*" Cloud Security Alliance (2011).

[2] Zwattendorfer, Bernd, et al. "*Cloud Computing in E-Government across Europe.*" Technology-Enabled Innovation for Democracy, Government and Governance. Springer Berlin Heidelberg, 2013. 181-195.

[3] Pato, Joseph, and One Cambridge Center. "*Identity management: Setting context.*" Hewlett-Packard, Cambridge, MA (2003).

[4] Kumaraswamy, Subra, et al. "*Domain 12: Guidance for identity & access management v2. 1.*" Cloud Security Alliance. (2010).

[5] "*An introduction to the Provisioning Services Technical Committee (Draft).*", OASIS PSTC (2001)

[6] Hunt, Phil. "*SCIM Use Cases (draft-ietf-scim-use-cases-03).*" IETF SCIM Working Group (2014).

[7] Poortinga-van Wijnen, Remco, et al. "*Provisioning scenarios in identity federations.*" (2010).

[8] OASIS Provisioning Services TC. "*Service Provisioning Markup Language (SPML) Version 1.0.*" OASIS Standard. (2003)

[9] OASIS Provisioning Services TC. "*OASIS Provisioning Services TC Glossary of Terms.*" OASIS PS-TC (2001).

[10] OASIS Directory services Markup Standard. "*Directory Services Markup Language 2.0.*", OASIS Standard (2001)

[11] Hommel, Wolfgang, and Schiffers, Michael. "*Supporting virtual organization lifecycle management by dynamic federated user provisioning.*" Proceedings of the 13th Workshop of the HP OpenView University Association: HP-OVUA. Vol. 6. 2006

[12] Maler, Eve. "*TechRadar™ For Security Pros: Zero Trust Identity Standards, Q3 2012.*" Forrester Research , Inc. (2012)

[13] Gietz, Peter. "*SCIM System for Cross-domain Identity Management.*" Presentation at Sitzung des ZKI AK Verzeichnisdienste, Leipzig (2014).

[14] Ensign, Chet. "*The OASIS Provisioning Services TC has closed*", *https://lists.oasis-open.org/archives/tc-announce/201208/msg00000.html*, Retrieved on November 2014

[15] Gietz, Peter and Widmer, Markus. "*Using OpenLDAP accesslog for SPML and SCIM based provisioning.*" LDAPCON 2013 (2013)

[16] Grizzle, Kelly, et al. "*System for Cross-Domain Identity Management: Core Schema (draft-ietf-scim-core-schema-14).*" IETF Network Working Group (2014).

[17] Bray, Tim. "*The JavaScript Object Notation (JSON) Data Interchange Format.*" (2014).

[18] Hardt, Dick. "*The OAuth 2.0 Authorization Framework (RFC6749).*"

[19] Phillips, Addison, and Davis, Mark. "*Tags for identifying languages.*" BCP 47, RFC 4646, September, 2006.

[20] Lear, E., and P. Eggert. *"Procedures for Maintaining the Time Zone Database."* BCP 175, RFC 6557, February, 2012.

[21] Klensin, John. *"RFC 5321—Simple mail transfer protocol (SMTP)"*. RFC 5321, 2008.

[22] Schulzrinne, Henning. "*The tel URI for telephone numbers.*" (2004).

[23] Ansari, Morteza, et al. *"System for Cross-Domain Identity Management: Protocol (draft-ietf-scim-api-14)."* IETF Network Working Group (2014).

[24] Fielding, R., Ed. and J. Reschke, Ed., *"Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content"* (2014).

[25] Dusseault, Lisa, and James M. Snell. *"Patch method for http."* (2010).

[26] Berners-Lee, Tim, Roy Fielding, and Larry Masinter. *"RFC 3986: Uniform resource identifier (uri): Generic syntax."* The Internet Society (2005).

[27] *"SCIM Implementations."* Available at: http://www.simplecloud.info/#implementations. [Accessed: Nov 2014]

[28] Ayyagari, Kiran. "*User Provisioning Over Web.*" LDAPCON 2013 (2013)

[29] *"Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC".* Official Journal of the European Union (2014)

[30] Zarsky, Tal Z., and Norberto Nuno Gomes de Andrade. "*Regulating Electronic Identity Intermediaries: The Soft eID Conundrum.*" Ohio St. LJ 74 (2013)

[31] *"The Global National eID Industry Report "*. Acuity Market Intelligence (2014)

[32] Arora, Siddhartha. *"National e-ID card schemes: A European overview."* Information Security Technical Report 13.2 (2008): 46-53.

[33] *"STORK 2.0 Member State's eIDs (January 2015)"*. STORK 2.0 Project (2015)

[34] *"Bundesgesetz über Regelungen zur Erleichterung des elektronischen Verkehrs mit öffentlichen Stellen".* BGBl. I Nr. 10/2004 (2004)

[35] Leitold, Herbert. *"The Austrian Citizen Card. A European Best Practice."* Innovation Forum 2009, Milano (2009)

[36] Lenz, Thomas et al. *"Identitätsmanagement in Österreich mit MOA-ID 2.0"*. eGovernment Review, Nr. 13 (2014)

[37] Federal Chancellery Austria. *"The Austrian Citizen Card"*. Available at: http://www.buergerkarte.at/konzept/securitylayer/spezifikation/20140114-en [Accessed: Nov 2014]

[38] Cantor, Scott, et al. *"Assertions and protocols for the oasis security assertion markup language."* OASIS Standard (March 2005) (2005).

[39] Pichler, Peter. *"Portalverbundprotokoll Version 2. Allgemeiner Teil"*. AG Integration und Zugänge (AG-IZ) (2011)

[40] *"SAML Interoperable Implementations, Tools, Libraries and Services".* Kantara Initiative. Available at https://kantarainitiative.org/programs/iop-saml [Accessed: Nov 2014]

[41] Pichler, Peter. *"Portalverbundprotokoll Version 2. eGovernment Attribute Profile"*. AG Integration und Zugänge (AG-IZ) (2011)

[42] Federal Chancellery Austria. *"The Austrian Citizen Card. Security Layer Application Interface"*. Available at: http://www.buergerkarte.at/konzept/securitylayer/spezifikation/20140114-en/core/Core.en.html [Accessed: Nov 2014]

[43] Federal Chancellery Austria. *"The Austrian Citizen Card. Security Layer Transport Protocols"*. Available at: http://www.buergerkarte.at/konzept/securitylayer/spezifikation/20140114-en/bindings/Bindings.en.html [Accessed: Nov 2014]

[44] WP4 Core Team. *"D4.2 First version of Functional Design"*. STORK 2.0 Consortium (2013)

[45] Zwattendorfer, Bernd. *"Identitäts-Protokolle für MOA-ID"*. E-Government Innovationszentrum (2013)

[46] Rissanen, Erik. "OASIS eXtensible Access Control Markup Language (XACML) Version 3.0." OASIS committee specification 1.