Performance Evaluation of Identity and Access Management Systems in Federated Environments

Frank Schell, Jochen Dinger, and Hannes Hartenstein

Steinbuch Centre for Computing & Institute of Telematics,
Karlsruhe Institute of Technology, Universität Karlsruhe (TH), Karlsruhe, Germany
{frank.schell,jochen.dinger,hannes.hartenstein}@kit.edu

Abstract. Identity and access management (IAM) systems are used to assure authorized access to services in distributed environments. The architecture of IAM systems, in particular the arrangement of the involved components, has significant impact on performance and scalability of the overall system. Furthermore, factors like robustness and even privacy that are not related to performance have to be considered. Hence, systematic engineering of IAM systems demands for criteria and metrics to differentiate architectural approaches. The rise of service-oriented architectures and cross-organizational integration efforts in federations will additionally increase the importance of appropriate IAM systems in the future. While previous work focused on qualitative evaluation criteria, we extend these criteria by metrics to gain quantitative measures. The contribution of this paper is twofold: i) We propose a system model and corresponding metrics to evaluate different IAM system architectures on a quantitative basis. ii) We present a simulation-based performance evaluation study that shows the suitability of this system model.

Keywords: identity and access management, federated identity management, access control, scalability.

1 Introduction

The task of assuring authorized access to services in distributed environments is performed by an identity and access management (IAM) system. The distinctive feature of IAM systems to other distributed systems is the handling of sensitive user data that raises privacy concerns. The setup of such a system leads to challenges in making fundamental design decisions that have significant impact on the properties of the overall system. This is getting even more complex in federated environments where identities can be exchanged between different security realms based on trust relationships established in advance. The challenges here are the correlation of the spatial distributed identity information of single users to a federated identity and the exchange of security-related information in this heterogeneous environment.

P. Mueller, J.-N. Cao, and C.-L. Wang (Eds.): Infoscale 2009, LNICST 18, pp. 90–107, 2009.

For example, in a scenario that uses a sophisticated access control policy, e.g., attribute-based access control [25], there will certainly be a so called policy decision point (PDP) stating authorization decisions about access requests of users. This component can be implemented in different ways and positioned at different hierarchical levels in a federated environment: A PDP could be realized as a component in a service itself, in an application server hosting the protected service, in an outsourced organization-wide service, in a service of another federation partner or even as a federation-wide service. The implications of such fundamental design decisions comprise in particular on the one hand impact on performance and scalability issues and on the other hand actuality, correctness, and confidential usage of identity information.

Looking at the extremes, a PDP could be positioned directly at each service provider. This decision ensures a fast runtime behavior with a high chance that this component is available. But each service provider would need all identity information for all users that want to use this service. This raises privacy concerns and demands for synchronizing this data, which can lead to failures and can also be a complex task. Another possible arrangement is to outsource a PDP from the single service providers to a trusted partner that handles all the access decision requests for them. With respect to the identity information stored at this central provider this leads to less inconsistencies due to the more up-to-date user attributes needed for determining the access control decisions and to easier operation and maintenance of the PDP. But this causes more network traffic, an increased latency, and leads to less autonomy of the service providers.

To guide the systematic engineering of identity and access management systems regarding their underlying architectures we thoroughly analyze the characteristics of these architectures in service-oriented environments. Therefore, we extend known qualitative evaluations with quantitative metrics. We address the implications of the architectural decisions in federated environments like the number of necessary IAM components for a specific user load and a given number of services. This also helps determining on which level the different IAM components concerning authentication and authorization processes are located. To our best knowledge there is not yet a methodology defined for determining the fitting identity and access management architecture for a specific use case. Therefore, we propose such a methodology in this contribution by showing how system architects of IAM systems can choose the right architecture for their specific scenario.

The paper is organized as follows. Section 2 describes the related work. The third Section introduces the methodology used to evaluate the different IAM approaches. In Section 4 we present evaluation criteria and metrics for IAM systems. Section 5 introduces the system model by specifying components, operations, messages, and dependencies of IAM. In Section 6 we evaluate different scenarios starting from a local approach to an outsourced AAI provider and we show how the design decision affects the system behavior. A conclusion and an outlook on future work in this area conclude the paper.

2 Related Work

There are different approaches for realizing access control like discretionary access control (DAC), mandatory access control (MAC) or more sophisticated ones like role-based access control (RBAC) [14] or attribute-based access control (ABAC) [25] that are more likely to be used in a distributed environment [2]. The basic principle behind ABAC is to use attributes for making authorization decisions to achieve more scalability than identity-based access control [19]. All necessary information is represented by a set of attributes and their values, which can be gathered dynamically if required, like user attributes, (e.g. roles, date of birth), environmental attributes (e.g. actual date and time), or attributes of resources (e.g. actual usage). Access control policies are typically specified as rules, which are evaluated with these attributes to allow or deny access to protected resources. An ABAC model for web services is introduced in [25].

The eXtensible Access Control Markup Language (XACML) is an XML-based standard for specifying access control policies, which can be processed to determine authorization decisions [23]. Furthermore it defines an architecture consisting of different components called XACML entities and a sequence of operation for these components. The XML entities comprise amongst others a policy enforcement point (PEP) for intercepting access requests and a policy decision point (PDP) for making authorization decisions. These components are well-known from policy-based networking [24]. In [19] an ABAC model is combined with language and architecture standards provided by the XACML specification, which uses automated trust negotiation mechanism to address the nondisclosure of sensitive attributes.

Federated identity management (FIM) enables the dynamic exchange of identity information across security domains based on trust relationships established in advance and therefore increases the portability of digital identities. [3], [12] and [9] show the fundamental concepts of federations and give an overview of federation protocols. In [15] the authors present the benefits using the federation paradigm for establishing an identity management system at large organizations.

Authentication and authorization infrastructures (AAI) support service providers to outsource security services to 3rd party providers [18]. This raises the overall level of security, provides a flexible access control model like ABAC, and eases the usability through, e.g., single sign-on (SSO) mechanisms [17]. Furthermore specific user data, e.g., user profiles, buying patterns, and earned privileges, can be gathered and transferred federation-wide for authorizing access to service providers based on actual data of federation members. Differences between the AAIs result from the chosen architecture depending on the level of outsourcing of security-related services [4]. Surveys of existing AAIs can be found on a technical level in [8], more detailed for b2c Commerce in [18], and with effects of architectural decisions in [16]. Here, the architectures of Shibboleth [20], Liberty Alliance [7], Passport [11], etc. are evaluated. In addition, the authors of [17] propose a reference architecture for an AAI respecting privacy and flexibility and [5] conducts user centric identity management architectures supporting, e.g., SSO, on a conceptual basis.

The authors of [10] propose a comprehensive approach for simulating IAM systems. They are setting the context for *Identity Analytics* in enterprises by adopting the scientific method [22] to approach this domain. The authors use discrete-event stochastic models for simulating various human activities and behavior, policies, social aspects, legislation, etc. This approach is used to give CIOs of an enterprise support in deciding on new or existing IAM investments by predicting their impact on relevant key factors to these decision makers, e.g., operational costs, reputation, compliance and so on. Though aiming at various aspects of IAM systems in complex enterprise contexts the authors do not focus on the consequences of using different IAM architectures and they do not support system architects with their approach.

3 Methodology

In [4] we presented some evaluation dimensions for access control architectures. We examined different architectural approaches for access control on a more conceptual basis using these criteria. Now we have done another step in evaluating such architectures by extending the qualitative results with quantitative results for differentiating these architectures. To achieve this, we use the following methodology to evaluate IAM architectures.

First, we define criteria as depicted in Fig. 1 that describe an area of interest of IAM systems. We have identified several major criteria, like scalability, robustness, and proliferation of security-relevant data. These criteria are a starting point for the further investigation of IAM systems. The next step is to define metrics for each criterion, which can be used to evaluate instances of different IAM approaches. The single metrics should reflect relevant characteristics of an IAM system that enable the differentiation of the single IAM approaches. The metrics can then be used for measuring systems in operation to provide input

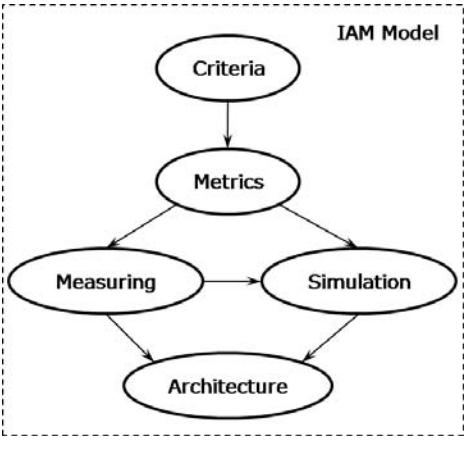

Fig. 1. Methodology for the Evaluation of IAM System Architectures

parameters for the model, e.g., duration and resource demand for determining authorization decisions. Another application of measurement is the calibration and validation of similar simulated scenarios that need to have an analogous behavior like the measured systems. The values for the metrics are gathered with measurement tools like web request tests, performance counters and network traffic analyzing tools. The metrics are also used in simulations to evaluate specific scenarios with the former specified metrics. The measuring and simulation help system architects to find the right architecture for their specific situation by matching their requirements with the results of the measurements and the simulation runs to determine their fitting scenario. Furthermore, simulation enables system designers, system architects, and other responsible persons for an IAM system to test new architectures of IAM systems before they are being deployed at all.

These activities demand for an IAM system model that describes all necessary parts of an IAM system, like the system components or the behavior of these components as it is described in section 4. The measuring and simulation can both lead to new insights about IAM systems, to new criteria that have to be followed, and to new metrics that have to be evaluated in subsequent measurements or simulation runs. This can lead to an improved overall evaluation process by a step-by-step refinement of the IAM system model.

We will further elaborate in this contribution on the aspect of simulation by specifying a basic model for IAM systems and by presenting results of simulation runs regarding aspects of scalability.

4 System Model for Identity and Access Management

The simulation model for IAM architectures comprises components, messages, processes and also dependencies between these elements.

4.1 Components

First, the components of the model are introduced.

- *Authentication Provider.* An authentication provider (AuthN-P) handles requests for user authentication. An AuthN-P provides an operation *authenticate*, which validates given user credentials like login and password, certificates, tokens or a combination of them.
- *Authorization Provider.* The component to decide on requests for access to a specific service is an authorization provider (AuthZ-P) or policy decision point (PDP). Therefore it gets all necessary information from attribute providers to determine these access control decisions. An authorization provider needs to implement an operation *authorize*, which determines to grant or deny access to a specific resource.
- *Attribute Provider.* An attribute provider (Attr-P), also called policy information point (PIP), provides data about specific users for authentication providers or authorization providers.

- *Enforcement Provider.* A component called policy enforcement point (PEP) or enforcement provider (Enf-P) restricts the access to a service provider. Therefore it intercepts access requests and asks for authentication and authorization statements at corresponding components.
- *Synchronization Provider.* A synchronization provider (Sync-P) detects changes of identity information in repositories and synchronizes these changes to connected repositories. Therefore, the synchronizer needs a rate for detecting changes. A Sync-P can be used to synchronize all or just parts of the data stored in the connected stores. The synchronization provider can be implemented using different technologies [13].
- *Data Repository.* There are a few kinds of data repositories, e.g., directory or database, defined. A credential store (Cred-S) is the repository for user credentials. Next, there is an attribute store (Attr-S) that serves as a repository for attributes about a user. The third type of data repository is the policy store (Policy-S), which stores the access control policies that are necessary for the AuthZ-P to state authorization decisions. Last, there could be some service repository (Serv-S) for storing information about all services of the federation. The availability of certain user data is modeled here.
- *Service Provider.* Any kind of resource that needs to be protected can be provided by a service provider (SP). This comprises infrastructure services as well as application layer services.
- *Client.* A user needs a client to consume certain services. A client can be a browser or any other application able to interact with the service provider.

Each provider has a rate for availability and the provided operations have resource costs (cpu, memory, network). On a more abstract view each operation should have modeled at least a duration for its execution. The providers communicate with messages that should have a certain latency of exchange, some size, and a rate for message losses.

4.2 Process Model

We present a basic process that may vary depending on the arrangement of the involved components in a specific scenario. All components can be arranged together to form the authentication and authorization processes.

A client starts the process by requesting a resource provided by a service provider. This request is intercepted by the Enf-P of the service provider, which checks the request for specific access control assertions, i.e., an authentication assertion and an authorization assertion. If not a valid authentication assertion is delivered, the client is requested to authenticate at a trusted AuthN-P. After a successful authentication process an authentication assertion is delivered to the Enf-P, which checks the authentication assertions for validity. After a successful validation the AuthZ-P is requested for stating an authorization decision. Therefore, the AuthZ-P gets the appropriate access control policies and the necessary attributes to decide on the request. An authorization assertion is delivered to the Enf-P that can now grant or deny the request of the client, which initiated this access control process.

An option that has to be cleared is the sequence of enforcement, authentication, and authorization processes. There are two possible sequences:

$$Enforcement \rightarrow Authentication \rightarrow Authorization(E1)$$

$$Authentication \rightarrow Enforcement \rightarrow Authorization(E2)$$

In sequence (E1) the enforcement provider is intercepting access requests and asking for authentication and authorization of the corresponding users. Sequence (E2) allows the user to first authenticate at the authentication provider before asking for access to a certain service provider. Some IAM systems allow both approaches so both scenarios can be mixed if required. There are two options for attribute retrieval in case (E1). There is the possibility to already get user attributes while authenticating users and send these to the enforcement provider (UAR1). Another way is to let only the authorization provider get all necessary attributes (UAR2).

5 Evaluation Criteria and Metrics for IAM Systems

The evaluation dimensions specified in [4] served as a fundament for the following defined categories and metrics.

5.1 Performance and Scalability

A substantial requirement for an IAM architecture is a high performance in most conditions. Therefore, the IAM system should be able to handle a certain constant or increasing number of users and a specific amount of service providers with low delays. To rate performance and scalability issues we deal with the following performance metrics for the evaluation of different IAM approaches.

- *Response time.* Keeping track of the response time of the IAM system components as a whole is an early indication of the capabilities of an architecture. An increasing response time of single components can also give a hint for overloaded components in tense situations. This also comprises the elapsed time for authenticating single users, determining access control decisions, searching user attributes, and so on.
- *Resource usage.* The measurement of the utilization of each component like usage of CPU/memory or incoming/outgoing network-load enables the detailed analysis of single IAM components and identification of bottlenecks.

5.2 Robustness, Reliability and Autonomy

The service providers should be available even if IAM components break down due to malfunction or any other reason. Robustness comprises correct authentication and authorization decisions under these circumstances. This demands for evaluating the degree of autonomy of foreign security domains with the following metrics.

- *Wrong access control decisions.* The single components have an imperfect view on necessary access control information like out-dated user attributes, credentials, etc. due to limited synchronization capabilities or unavailable information providers. Evaluating the number of wrong authentication or authorization decisions in tense situations gives information about the robustness of the underlying IAM architecture.
- *Attribute authorities.* An authorization process depends on up-to-date user attributes that can be spread over a number of attribute providers storing this information. The aggregation of this information from too many entities can be time-consuming and lead to erroneous results due to out-dated data.
- *Trusted components.* Access control is a sensible task that requires a minimum amount of trust between cooperating entities. Therefore a metric that lists all trusted components for an access control decision is helpful to determine possible data leakage.

5.3 Proliferation and Quality of Security-Relevant Data

Regarding privacy issues the dissemination of user data in the overall system has to be analyzed for the different IAM approaches. Also the timeliness and accuracy of the distributed user data and access control policies has a direct impact on making correct authentication and authorization decisions. Therefore we define the following metrics for evaluating the proliferation and certain quality aspects of security-relevant data.

- *Access to user attributes.* The number of components with access to user attributes in plaintext gives a hint for the risk of revealing this information. The more components the attributes can access or the more clients can access a service the more it is likely that a security breach may occur.
- *Timeliness of user data.* Access control decisions should be based on up-to-date identity information, so we need a metric for evaluating the actuality of this information. This comprises the timeliness of synchronized access control policies, too.
- *Accuracy of user data.* The data stored in a repository is likely to be not exactly, e.g., due to typos at data acquisition. This leads to erroneous or wrong access control decisions. This also includes faulty specified access control policies.

5.4 Integration Costs

A protected service is to be integrated in the overall IAM system. The different IAM approaches demand for a varying effort in the development phase of a service, e.g., programming of security-related code, and in the operation phase due to configuration effort like the configuration of SSL in the application server or the configuration of more sophisticated technologies like parameters of the Windows Communication Foundation [21].

5.5 Costs of Operation

The establishment of an IAM demands for specific knowledge and time in an organization. This also comprises the installation of the overall system or the definition of administrative processes like the specification of security policies or access control policies. Furthermore, the operation of an IAM system demands, e.g., specification of roles, definition of access control policies or configurations of IAM components, and documentation of the system.

The integration and operational costs are hard to evaluate as it is known from the software development discipline. The focus of this paper are scalability issues of IAM systems. Thus, we will evaluate different performance parameters of IAM systems. Evaluation results of the remaining metrics will be presented in further publications.

6 Evaluating Identity and Access Management Systems

Each of the specified components of the system model for IAM can be arranged locally at a service provider or at one or more external provider(s). Based on the AAI security sub-services decision tree of [16], there are three possibilities for realizing external providers of a component.

For simplicity reasons, we locate in a first step the associated stores with their respective components. For example the credential store is located at a AuthN-P, i.e., if the AuthN-P is local then the credential store is positioned locally, too. The authorization provider has a policy store and if positioned externally also a service store, which holds information for resolving access control policies concerning specific service providers. Furthermore, each attribute provider has a co-located attribute store.

A brief discussion of the pros and cons of the positioning of authentication providers, authorization providers, attribute providers and enforcement providers either local, single central, few central, and distributed can be found in [16]. The possible arrangements for each specific type of the formerly defined providers of an IAM system are as follows.

- Authentication Provider: local, single central, multiple central, distributed
- Authorization Provider: local, single central, multiple central, distributed
- Attribute Provider: local, single central, multiple central, distributed
- Enforcement Provider: local, single central, multiple central as a proxy
- Synchronization Provider: between service providers or multiple central providers

Figure 2 depicts the fundamental positioning possibilities for providers of the system model for IAM. The AuthN-P, AuthZ-P and Attr-P can be arranged in all shown arrangements. The Enf-P has no store that needs to be synchronized, so the possibilities for this component can be reduced to (1A), (2), (3A) and (4A).

(1A) shows a local provider at the service provider don't having dependencies to other service providers at all. (1B) depicts the same local providers, but the

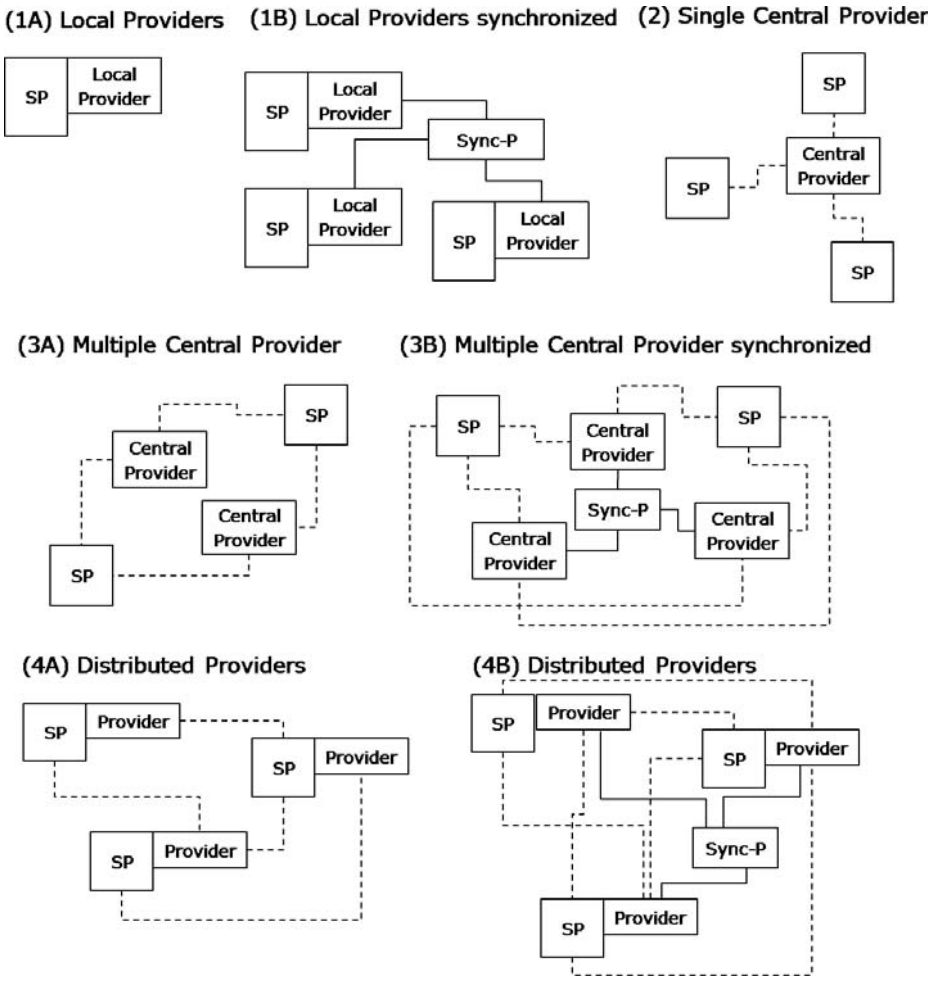

Fig. 2. Possiblities for the Arrangement of Providers

stores of the local providers are synchronized. Arrangement (2) describes a single central provider that is used by all participating service providers. So there is just one single central entity that provides this functionality. This could be an authentication provider like it is known from Passport [6].

Next, a few, still centralized, providers might provide this kind of component like a few Identity Providers given in a typical Shibboleth Federation [20]. (3A) shows these multiple central providers, which can be used by different service providers. The service providers may use more than one of these providers. In Arrangement (3B) there is a Sync-P added that can synchronize all or parts of the data stored at the multiple central providers to achieve consistent data.

Another possibility is the total distribution of this type of component to all participating services providers, so that each of the service providers are able to act, e.g., as an authentication provider for another service provider. (4A) depicts this distributed case, where all service providers have recourse to the providers of the other service providers. The arrangement (4B) shows an additional Sync-P that synchronizes data for this provider between the service providers.

We state the following assumptions for the simulated scenarios.

- *Workload.* We observe all scenarios with different user workloads trying to get access to a specific service provider. For each scenario we are evaluating different conditions, e.g., from 1 up to 1000 users working in parallel on a single provider. The time for the single users between two runs is set to zero.
- *CPU processing rate.* The processing rate of the CPUs is set to 1 GHz. Each service provider and outsourced provider has a single CPU available.
- *CPU resource demands.* We state the following resource demands for the single operations.
  - *AuthN-P.* We assume a distribution of 90% usage of passwords for authentication with a resource demand of 1000 cycles and a 10% usage of certificates, e.g., authentication of administrators or access request to more restricted resources, with a resource demand of 2000 cycles, due to the higher computational costs of asymmetric encryption.
  - *AuthZ-P.* An authorization provider aggregates the necessary access control policies and computes them with a resource demand of 2000 cycles, due to the inherent complexity of the rules.
  - *Attr-P.* The task of transforming attributes requested by an authorization provider is relatively expensive due to the given complexity of evaluating these policies by specific rules. We assume a resource demand for each policy of 2000 cycles.
  - *Enf-P.* An enforcement provider intercepts access requests and validates assertions given by the user for validity. This operation costs 500 cycles.
  - *Stores.* We assume the same costs for all of the operations provided by the stores with 500 cycles.
- *Dependencies.* All simulated scenarios use the sequence of enforcement (E1) and the user attribute request strategy (UAR2), so that the enforcement provider intercepts the access requests and asks for the authentication of the requesting principal and demands for an authorization decision at the AuthZ-P, which aggregates the necessary attributes for determining the access control decision.

For conducting the simulations we use the Palladio Component Model (PCM) [1]. PCM is designed to enable early performance predictions by specifying a domain specific modeling language for component-based software architectures. Therefore, PCM uses UML-like models that allow different roles involved in the overall development process of software systems, like developers, software architects, system deployers, and domain experts, to specify their respective part of a system.

The components of a system are specified in a repository that uses service effect specifications (SEFF) to model the internal behavior of the single components. A system model declares the components used for a specific simulation and the connections between the components to realize the desired system. The resource environment model describes the provided hardware like cpu, memory, and network resources. The system model and the resource environment model are used by the allocation model to define the resources that will be used by specific components. Furthermore, the usage model specifies the behavior of the users that use the system.

PCM is implemented as an Eclipse plugin, which allows the creation of PCM model instances in a graphical editor that is based on the Eclipse Modeling Framework. With that, it enables developers to derive performance metrics from the models using analytical techniques and simulation.

6.1 Scenarios

We define the following scenarios as fundamental IAM architectures based on the aforementioned model. All the scenarios are depicted in Fig. 3.

Scenario (A) - Local Providers. The service providers act in this scenario as Enf-P, AuthN-P, AuthZ-P and Attr-P for themselves. They do not share any provider with other service providers, so each service provider has to implement its own providers and there is no interaction necessary to a central provider or between the single service providers.

Scenario (B) - Single Identity Provider. Like scenario (A) all service providers are acting as enforcement and authorization provider for themselves here, but the authentication and attribute provider are outsourced to a single entity called identity provider (IdP) as it is known, e.g., from Passport. There is no need to update the credentials, due to the single, centralized identity provider. Furthermore the service providers don't need a service repository for storing information about the other service providers, because they don't use capabilities of any other service provider.

Scenario (C) - Single AAI Provider. The next scenario is a single authentication and authorization infrastructure (AAI) provider, which provides authentication and authorization processes for the federated service providers. It stores the needed information, credentials, attributes, and access control policies, in central repositories. It also uses a service repository to find the right service provider for retrieving attributes, which are demanded for access control decisions. Therefore, the single service providers just need to implement an enforcement provider to use the AuthN-P, AuthZ-P, and Attr-P of the AAI provider.

Scenario (D) - Identity Provider and Policy Decision Point. Scenario (D) is an outsourced identity provider, i.e., AuthN-P and Attr-P, and an outsourced policy decision point, i.e., AuthZ-P, each as a single service, but all

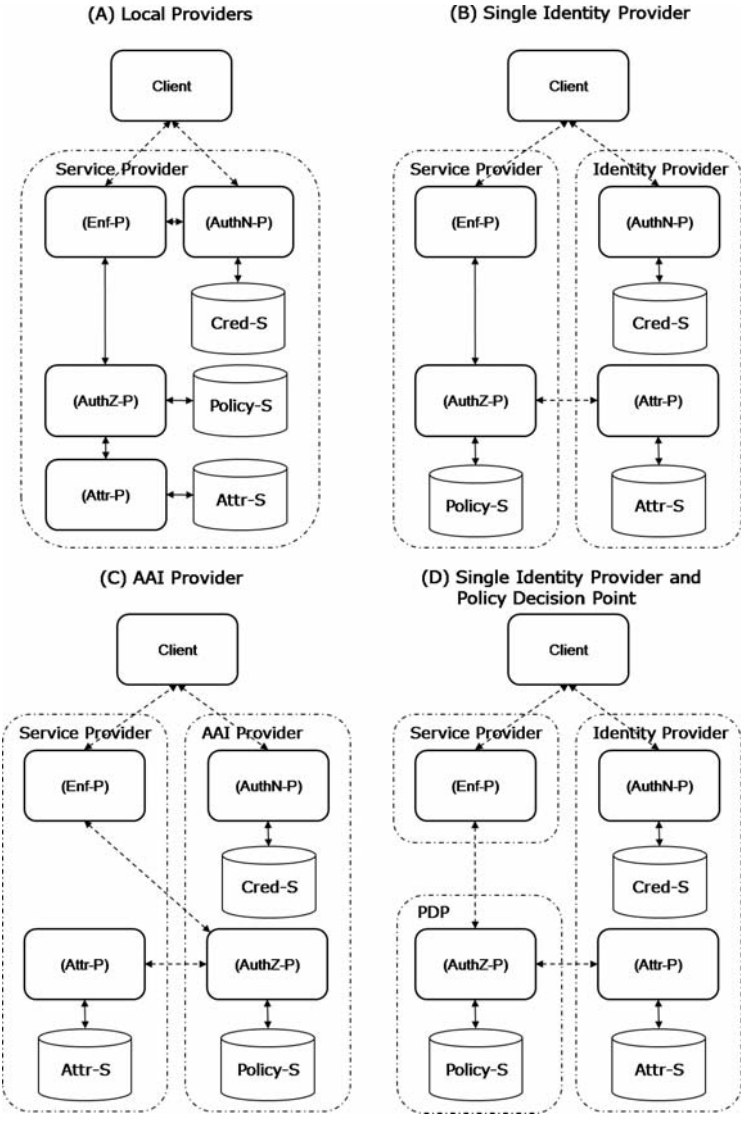

Fig. 3. Simulated Scenarios

service providers are acting as enforcement provider for themselves. There is no need for a Sync-P due to the single, centralized providers. Furthermore the service providers don't need a service repository for storing information about the other service providers, because they don't use capabilities of other service providers.

6.2 Comparison of the Scenarios

We evaluate the aforementioned scenarios regarding performance aspects in particular response times. Figure 4 depicts the cumulative distribution functions (CDF) for different user workloads from 1 to 1000 users in parallel for scenario (D). The x-axis shows the response time for requesting a resource until the authentication and authorization processes successfully granted or denied access in milliseconds from 0.1ms to 10000ms in a logarithmic scale. A value on the y-axis represents the probability of the IAM system to respond in this or less time.

A first glance at this figure shows that the probability for a longer response time increases with the number of users trying to gain access to resources provided by the service providers as expected. The maximum response times for the different workloads are as follows. 1 single user gets a result at the latest after 2ms, 10 users in parallel in 15ms, 100 users in 167ms and 1000 users in 1671ms. A ten times higher user workload leads to a 10 times higher latest response time. So the number of users correlates with the response time of the IAM system in this case. The load is dispersed in scenario (D) to the identity provider and the policy decision point leading to a distributed computation of the various IAM tasks.

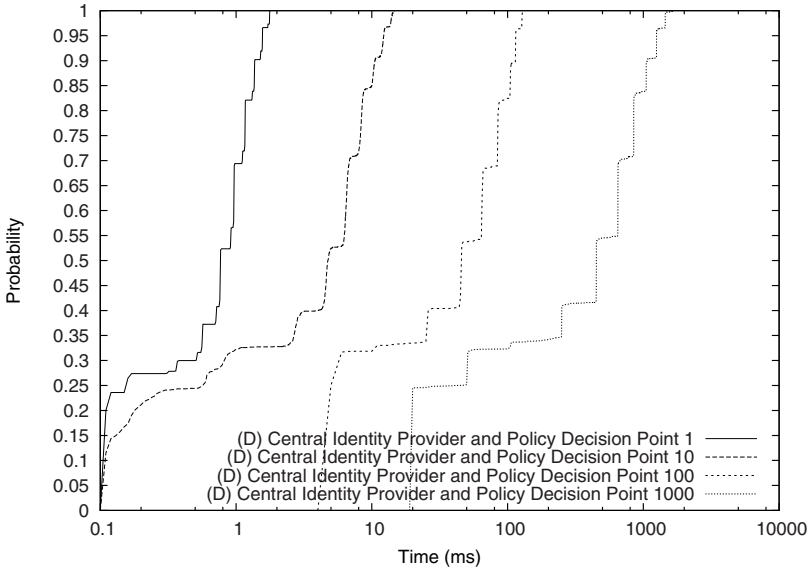

Fig. 4. Scenario (D): Probability of Response Time for 1 up to 1000 Users in Parallel as Cumulative Distribution Functions (Logarithmic Scale)

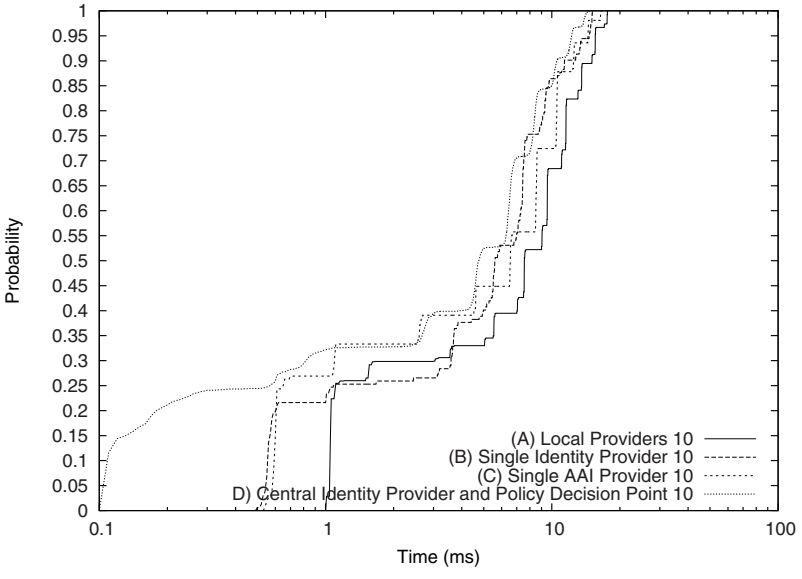

Fig. 5. All Scenarios: Probability of Response Times with Workload of 10 Users in Parallel as Cumulative Distribution Functions (Logarithmic Scale)

If we compare the different simulated scenarios we see distinct response times. Figure 5 shows the CDFs for the response time for 10 users in parallel trying to access resources. The different systems nearly have the same behavior for this user load. 50% of the requests for a single user are served in less than 10ms and the maximum response time is under 20ms in each case.

If we increase the number of users to 100 as depicted in Fig. 6 we can see that scenario (A) has the worst response times of all simulated scenarios. The maximum response time 219ms is nearly twice as high as the response time of scenario (D) with 128ms. The response times of both scenario (B) and scenario (C) are nearly similar under these circumstances, 40% of the access requests can be satisfied in 10ms or less.

Figure 7 shows the scenarios with a workload of 1000 users trying to gain access to the resources in parallel. Between 30% and 35% all scenarios are reacting nearly similarly, but as we can see clearly in higher percentages there is a gap between the local provider scenario and the other scenarios. The local providers are on heavy load due to the resource demands of all providers having a maximum response time of over 15000ms. This is a factor of nearly 10 to the central AAI scenario, which has a maximum response time of 1706ms. Scenario (B) has slightly faster reaction times than scenario (C) with a maximum of approximately 90ms. Best scenario regarding the response time is scenario (D), due to the distribution of resource demands to outsourced servers.

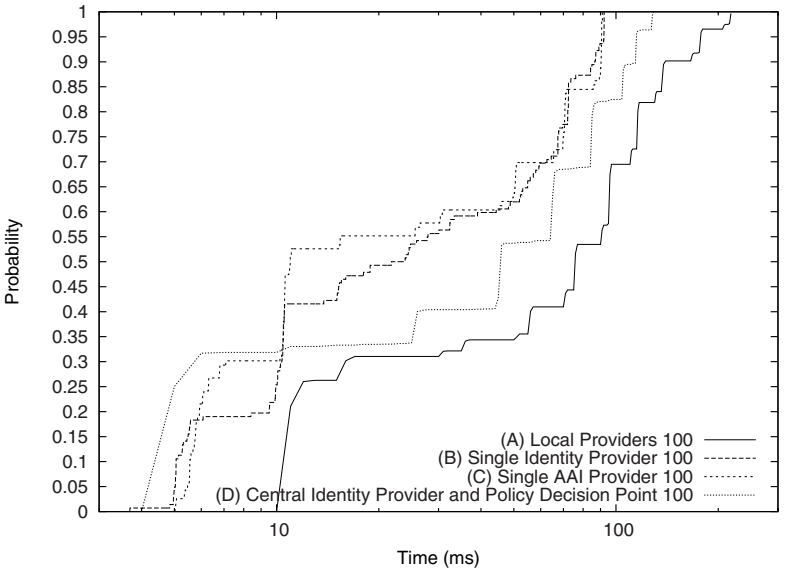

Fig. 6. All Scenarios: Probability of Response Time with Workload of 100 Users in Parallel as Cumulative Distribution Functions (Logarithmic Scale)

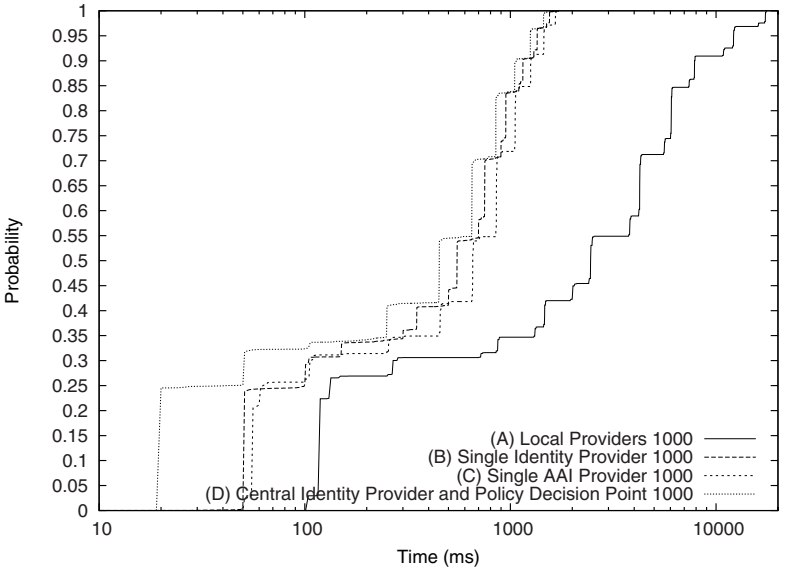

Fig. 7. All Scenarios: Probability of Response Time with Workload of 1000 Users in Parallel as Cumulative Distribution Functions (Logarithmic Scale)

Reviewing the CDFs of all simulated scenarios we can state that scenario (A), where all service providers act as enforcement, authentication, authorization and attribute providers for themselves, is not as efficient as the other simulated scenarios. Scenario (B) and (C) are reacting nearly the same under different user loads. If we take no other metric than response time into account, scenario (D) would fit our requirements the best.

7 Conclusions and Future Work

In this contribution we proposed a methodology and a system model for the evaluation of identity and access management architectures, which assure authorized access to services in distributed environments. This helps system architects of such systems in making the right design decisions, in particular the arrangement of the involved components. The position has significant impact on the properties like performance and scalability of the overall system. We extended existing qualitative evaluations by using the system model to derive criteria and metrics that show qualitative differences of IAM approaches in simulations.

Next steps include the simulation of the remaining metrics and the refinement of the IAM system model, e.g., detailed authentication protocols, fine-grained sync policies or authentication policies, varying user behavior, etc. to achieve more realistic results. This can be achieved by either adapting the Palladio Component Model to be able to determine more identity-specific metrics or by using a more generic simulator. Furthermore, we will further elaborate on the assumptions of the simulated scenarios that will also lead to more realistic results of the simulations. Another aspect that has to be addressed is the simulation of mixed scenarios. For example, some service providers sharing their providers, some using central providers, some only using their local providers, etc. altogether mixed in one simulated scenario.

References

1. Becker, S., Koziolek, H., Reussner, R.: Model-based performance prediction with the palladio component model. In: Proceedings of the 6th international workshop on Software and performance, pp. 54–65. ACM, New York (2007)
2. Benantar, M.: Access control systems: security, identity management and trust models. Springer, Heidelberg (2006)
3. Djordjevic, I., Dimitrakos, T.: A note on the anatomy of federation. BT Technology Journal 23(4), 89–106 (2005)
4. Höllrigl, T., Schell, F., Suelmann, S., Hartenstein, H.: Towards systematic engineering of Service-Oriented access control in federated environments. In: IEEE Congress on Services Part II, SERVICES-2., pp. 104–111 (2008)
5. Jøsang, A., Pope, S.: User centric identity management. In: Proceedings of AusCERT Asia Pacific Information Technology Security Conference, pp. 77–89 (2005)
6. Kormann, D., Rubin, A.: Risks of the passport single signon protocol. Computer Networks 33, 51–58 (2000)

7. Liberty alliance project (2009), http://www.projectliberty.org/
8. Lopez, J., Oppliger, R., Pernul, G.: Authentication and authorization infrastructures (AAIs): a comparative survey. Computers & Security 23(7), 578–590 (2004)
9. Maler, E., Reed, D.: The venn of identity: Options and issues in federated identity management. IEEE Security & Privacy 6(2), 16–23 (2008)
10. Mont, M., Baldwin, A., Griffin, J., Shiu, S.: Towards Identity Analytics in Enterprises. To Appear: Proceeding of the 24th IFIP International Information Security Conference (2009)
11. Passport (2009), https://accountservices.passport.net/ppnetworkhome.srf
12. Pfitzmann, B., Waidner, M.: Federated identity-management protocols. In: Christianson, B., Crispo, B., Malcolm, J.A., Roe, M. (eds.) Security Protocols 2003. LNCS, vol. 3364, pp. 153–174. Springer, Heidelberg (2005)
13. Ping Identity. Federated Provisioning: The Synergy of Identity Federation and User Provisioning, http://www.pingidentity.com/information-library/resource-details.cfm?customel_datapageid_1296=7587
14. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-based access control models. Computer 29(2), 38–47 (1996)
15. Schell, F., Höllrigl, T., Hartenstein, H.: Federated Identity Management as a Basis for Integrated Information Management. it-Information Technology 51(1), 14–23 (2009)
16. Schläger, C., Ganslmayer, M.: Effects of Architectural Decisions in Authentication and Authorisation Infrastructures. In: The Second International Conference on Availability, Reliability and Security, ARES 2007, pp. 230–237 (2007)
17. Schläger, C., Nowey, T., Montenegro, J.: A Reference Model for Authentication and Authorisation Infrastructures Respecting Privacy and Flexibility in b2c eCommerce. In: Proceedings of the First International Conference on Availability, Reliability and Security, pp. 709–716 (2006)
18. Schläger, C., Pernul, G.: Authentication and Authorisation Infrastructures in b2c e-Commerce. In: Bauknecht, K., Pröll, B., Werthner, H. (eds.) EC-Web 2005. LNCS, vol. 3590, pp. 306–315. Springer, Heidelberg (2005)
19. Shen, H., Hong, F.: An attribute-based access control model for web services. In: Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT 2006, pp. 74–79 (2006)
20. Shibboleth (2009), http://shibboleth.internet2.edu/
21. Smith, J.: Inside microsoft windows communication foundation. Microsoft Press, Redmond (2007)
22. Wilson, E.: An introduction to scientific research. Courier Dover Publications (1990)
23. OASIS eXtensible Access Control Markup Language, XACML (2009), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml
24. Yavatkar, R., Pendarakis, D., Guerin, R.: A Framework for Policy-based Admission Control. RFC 2753, Informational (2000)
25. Yuan, E., Tong, J., Inc, B., McLean, V.: Attributed based access control (ABAC) for Web services. In: 2005 IEEE International Conference on Web Services, ICWS 2005. Proceedings, pp. 561–569 (2005)