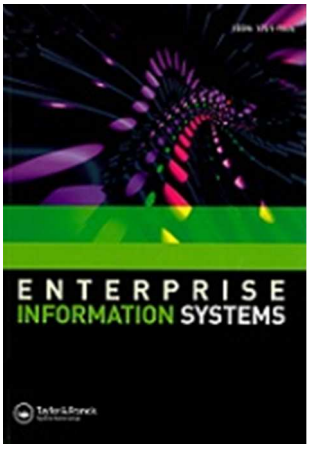


Policy Reconciliation for Access Control in Dynamic Cross-Enterprise Collaborations

Journal:	*Enterprise Information Systems*
Manuscript ID	TEIS-2016-0161
Manuscript Type:	Original Article
Keywords:	Security, Policy-based access control, Dynamic cross-enterprise collaboration, Authorization, Enterprise Computing Tools

For Peer Review Only

SCHOLARONE™
Manuscripts

To appear in *Enterprise Information Systems*
Vol. 00, No. 00, Month 20XX, 1–18

Policy Reconciliation for Access Control in Dynamic Cross-Enterprise Collaborations

Anonymous

(*Received 00 Month 20XX; final version received 00 Month 20XX*)

In dynamic cross-enterprise collaborations, different enterprises form a – possibly temporary – business relationship. To integrate their business processes, enterprises may need to grant each other limited access to their information systems. Authentication and authorization are key to secure information handling. However, access control policies often rely on non-standardized attributes to describe the roles and permissions of their employees which convolutes cross-organizational authorization when business relationships evolve quickly.

Our framework addresses the managerial overhead of continuous updates to access control policies for enterprise information systems to accommodate disparate attribute usage. By inferring attribute relationships, our framework facilitates attribute and policy reconciliation, and automatically aligns dynamic entitlements during the evaluation of authorization decisions. We validate our framework with a Industry 4.0 motivating scenario on networked production where such dynamic cross-enterprise collaborations are quintessential. The evaluation reveals the capabilities and performance of our framework, and illustrates the feasibility of liberating the security administrator from manually provisioning and aligning attributes, and verifying the consistency of access control policies for cross-enterprise collaborations.

Keywords: security; policy-based access control; dynamic cross-enterprise collaboration; authorization; enterprise computing tools

For Peer Review Only

1. Introduction

Enterprises are turning to attribute-based access control (ABAC) policies to enforce authorization to their information systems and business applications because ABAC policies give them the flexibility they need to express authorization rights and entitlements with a finer granularity of access control. The reason is twofold: (1) attributes can describe any property of an entity (e.g. the user, information asset, context) that must be considered for authorization purposes, and (2) a policy-based access control approach defines and evaluates security rules separate from the core business logic, which makes such security policies easier to adapt to changing access control demands.

Nowadays, state-of-practice identity and access management (IAM) systems already externalize *authentication* (i.e. ascertaining that somebody really is who he claims to be) and *authorization* (i.e. security rules that determine who is allowed to do what) from the main application. An IAM system relies on an identity provider (IdP) to describe subjects with different (types of) attributes (e.g. the profile, roles and credentials of a user within the organization). This way, security administrators can eliminate the need to maintain separate user credentials and access control policies for different services, hereby greatly simplifying the identity life cycle management of the users and their permissions within the trust boundaries of the enterprise.

For enterprise collaborations across these trust boundaries, the security administrator can add the IdP of a collaborating business partner to the circle of trust of its own IAM

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 1. Figure captions are saved separately, and listed at the end of the manuscript.

system, enabling federated single sign-on (SSO) to facilitate *cross-enterprise authentication* through delegation. However, *cross-enterprise authorization* remains a non-trivial challenge, as different enterprises may grant limited access to each other's business applications and data (see Figure 1) but use different attributes and impose other security rules regarding the subject requesting access, the actual resource to be accessed, and the nature of the temporary business partnership. By not having a centralized authority in place that standardizes all identity and attribute definitions, *Enterprise 1* may rely on possibly missing or inconsistent attributes of the IdP of *Enterprise 2* to evaluate and enforce its own access control policies.

1.1. *Problem statement*

The first challenge that we address in this work is that in dynamic cross-enterprise business ecosystems, there is no common definition for the attributes in the IdP of one enterprise and those used in the access control policies of the other enterprise. There are solutions to map attribute types from one backend system to another, but they target batch reconciliation of mainly static attributes across IdPs and databases within a single enterprise. However, *authorization* across enterprise boundaries that reacts immediately to dynamic business relationships requires disparate attribute types and values to be reconciled *during the evaluation of the access control decision* rather than in batch mode.

The second challenge that we address targets the recent trend of IAM moving towards identity relationship management (IRM) (Andresen (2014)). Whereas IAM systems were traditionally built for access control by on-premise employees, does IRM target managing trust relationships with parties inside and outside of the enterprise, including those of customers, devices and other things across intertwined networks. This adds to the managerial overhead for the security administrator in charge of access control for the enterprise information systems. Not only will the diversity of user and device attributes continue to proliferate, the inherent dynamic, transient and implicit identity relationships will further aggravate the complexity of freeing enterprises from manually establishing collaborations with one another and security administrators having to continuously update their authorization policies.

1.2. *Contribution*

To address these challenges, our access control framework provides the following key contributions for enterprise information systems:

(1) Facilitating policy reconciliation through inference of transient and implicit identity and attribute relationships during the evaluation of access control decisions
(2) Simplified administration of cross-organizational dynamic entitlements and identity relationship management
(3) A practical implementation on top of a state-of-practice IAM system with an acceptable performance overhead

We validate the framework in an Industry 4.0 use case where temporary cross-enterprise collaborations for manufacturing and logistics are quintessential. The scenario exhibits networked production across production factories where a customer's personal preferences can be swiftly fulfilled without any delay on the production process.

After discussing related work in section 2, we present a motivating use case on net-

For Peer Review Only

worked production and concerted manufacturing processes in section 3. We propose in section 4 our framework for reconciling attributes and policies for identity access and relationship management. In section 5, we aim at mapping these concepts onto a practical implementation on top of a state-of-practice IAM system. We discuss performance aspects in section 6. We conclude in section 7 summarizing our main insights.

2. Related work

In this section, we briefly review the state-of-the-art in the area of policy-based access control, discuss cyber-security challenges in Industry 4.0 collaboration scenarios, identify what is missing and highlight how we aim to bridge this gap.

2.1. *Policy-based access control*

Access control is a key information protection mechanism, with the most common, oldest, and most well-known identity-based access control models being Discretionary Access Control (DAC), Mandatory Access Control (MAC) and Role Based Access Control (RBAC) (Sandhu (1993); Sandhu and Samarati (1994); Sandhu et al. (1996)). Recently, there has been growing interest in Attribute Based Access Control (ABAC) (Jin, Krishnan, and Sandhu (2012)) to overcome the limitations of the aforementioned access control models. The ABAC model makes decisions on permitting or denying access by relying on attributes of subjects, resources, actions, and the environment. It allows for resource owners, such as enterprise information systems, to grant access to unanticipated users as long as they have attributes that meet certain criteria. In policy-based access control, such as Ponder (Damianou et al. (2001)), Rei and KAoS (Tonti et al. (2003)), and the eXtensible Access Control Markup Language (XACML) specification (XACML-V3.0 (2012)), regulation of access to protected resources is expressed external to the applications as high-level rules that define who has access to what resources under what conditions. In Ferraiolo et al. (2016), the authors offer a comparison of the XACML and NGAC attribute-based access control specifications and underlying architectures. The latest trend in access control models is Risk-Adaptive Access Control (RAdAC) (Kandala, Sandhu, and Bhamidipati (2011); Ni, Bertino, and Lobo (2010)) where access decisions depend on dynamic risk assessments.

2.2. *Cyber-security in Industry 4.0*

A detailed discussion on the definition of Industry 4.0 (Lee, Bagheri, and Kao (2015)) is beyond the scope of this work. Instead, we refer to a quantitative text analysis and a qualitative literature review carried out by Herman et al. (Hermann, Pentek, and Otto (2016)), in which they identify the main design principles of Industry 4.0. They also present a case study to illustrate how these design principles can support practitioners in identifying Industry 4.0 scenarios. In Lee, Bagheri, and Kao (2015), the authors discuss the systematical deployment of Cyber-Physical Systems (CPS), and propose a unified 5C-level architecture (based on *connection, conversion, cyber, cognition* and *configure*) as a guideline for the implementation of CPS in the manufacturing industry. More recently, Wang et al. (2016) highlighted the need for horizontal integration of inter-corporation value networks, the end-to-end integration of engineering value chain, and the vertical integration of factory inside. They discuss the importance of emerging technologies such as IoT, big data and cloud computing in Industry 4.0.

Cyber-security remains a clear challenge for the rollout of the smart factories of the

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 2. Figure captions are saved separately, and listed at the end of the manuscript.

future. Many of the systems, technologies and protocols that exist today and that will become constituents of Industry 4.0 were never designed with networked production and large scale connectivity in mind. This can be witnessed from recent successful attacks on SCADA systems by dangerous malware like Stuxnet, Duqu, Flame, and Gauss (Langner (2011); Bencsáth et al. (2012)). Nicholson et al. (2012) survey ongoing research and present an overview of risks, threats and mitigation strategies in the area of SCADA security.

2.3. *Bridging the dynamic cross-enterprise collaboration security gap*

Networked and individualized mass production will communicate considerably more information about the manufacturing process itself across the network. A major threat are attackers involved in industrial espionage that steal and sell this information to competitors to produce counterfeit products. With unauthorized access to sensitive information becoming a key concern, identity and access management in the Internet of Things (IoT) will be critical in an overarching defense-in-depth strategy to the success of Industry 4.0. In a book (Mahalle and Railkar (2015)) and recent Gartner report (Perkins and Allan (2015)) on the same topic, the challenges of identity management and the identity of things in IoT are highlighted. As machines, products and services will be more abundant than users, the industrial IoT requires managing exponentially more identities and relationships between entities than contemporary Identity and Access Management (IAM) systems had to support up until today.

Our framework goes beyond the state-of-the-art by addressing authorization for enterprise information systems with a solution that specifically focuses on the enforcement of access control policies for dynamic cross-enterprise collaborations. To address the aforementioned challenges, our framework bridges this gap by automatically aligning disparate attribute usage across enterprise information and security systems.

3. Industry 4.0 motivating use case

A digital transformation is taking place in the manufacturing world, often referred to as the 4^{th} Generation Industrial Revolution (Industry 4.0) (Lee, Bagheri, and Kao (2015)) or the Factory of the Future (FoF) (Karnouskos et al. (2012)). These paradigm shifts envision smart factories where the Internet of Things (IoT) and Cyber-Physical Systems (CPS)-enabled manufacturing will provide the foundations for creating smart products through smart processes and procedures. Smart products will plan, control and optimize their own production process with minimal human intervention. The digital transformation with business applications moving to the cloud will enhance the transparency of the production process, even across the organizational boundaries of the manufacturing enterprise (see Figure 2).

Digital identities and relationships already play a major role in manufacturing and logistics to track and trace (Kemény, Ilie-Zudor, and Monostori (2009)) the current and past locations of products and the relationships with the customer. However, the paradigm shift also harbors severe cyber-security risks and threats (Nicholson et al. (2012)), ranging from viruses and malware (Langner (2011); Bencsáth et al. (2012)) that sabotage critical infrastructure, to unauthorized access to sensitive customer data and industrial espionage from within. Especially in production networks that operate across

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 3. Figure captions are saved separately, and listed at the end of the manuscript.

the organizational trust boundaries of the enterprise, the number of potential targets for attack increases. With unauthorized access to sensitive information being a major concern in this rapidly evolving and competitive business, adequate verification of digital identities, access privileges and relationships across the trust boundaries of several partnering enterprises is paramount. The key challenges that the stakeholders in such non-centralized cross-organizational collaboration scenarios are often faced with, include:

(1) There is an exponential growth of identities for employees, business partners, consumers and machines, adding to the managerial overhead.
(2) Production transparency needs in-network access control rather than just at the edge of the network.
(3) Cross-organizational authorization policies use inconsistent attributes for identities and relationships.

The first two challenges can be addressed with scalable identity management solutions that – specifically for networked production – enable integration with legacy manufacturing systems. This work focuses on the last challenge to ensure that dynamic entitlements are enforced correctly by all collaborating partners even without an upfront agreement on their definition and relationship with other attributes. There is a huge demand for such solutions, as due to the dynamic nature of business relationships, companies at one stage can collaborate as business partners only to act as competitors for future production orders.

4. Reconciling attributes and relationships

This section discusses relevant concepts in identity relationship management, key concerns with dynamic entitlements in cross-organizational policy-based access control, and limitations in the state-of-practice to effectively address these challenging concerns.

4.1. *Identities, attributes and relationships*

Identity management systems are based on the concept of a digital identity, which can represent an individual, a device or any other type of entity. These identities are annotated with attributes that are instantiated according to a data schema of an LDAP server or an entity relationship model of an underlying SQL database.

The identities and their attributes are used in a security model or policy to evaluate access control decisions. Role-based access control (RBAC) (Sandhu et al. (1996)) is one of the most widely adopted security models in many enterprises. Figure 3 illustrates the metamodel of RBAC where (1) users are granted roles, (2) permissions are assigned to roles, (3) permissions are expressed as combinations of operations on objects, and (4) users acquire permissions by activating a role in an application session. Most RBAC implementations also enable the definition of *role hierarchies* (i.e. a partial order between roles) to model the inheritance of permissions, and constraints for enforcing *separation of duties* (SOD) to spread the responsibility of a sensitive task over multiple individuals. A key challenge with networked production where manufacturing is driven by Cyber-Physical Systems is the explosion of roles. These systems are objects with their own identity, and access control supervises the execution of operations on other objects. With an IRM roadmap that unifies the identity of objects and users, objects will be assigned

For Peer Review Only

roles too to infer their permission set. Furthermore, enforcing dynamic separation of duties for cross-organizational tasks means a further duplication of certain roles.

The role explosion is the main reason why enterprises have recognized there is a need to go beyond groups of permissions and roles to manage access control to their shared resources. Attribute-based access control (ABAC) (Yuan and Tong (2005); Coyne and Weil (2013)) is a model where requests of a user (a.k.a. subject) to perform operations on an object (a.k.a. resource) are granted or denied based on rules. These rules express conditions on attributes assigned to the user and the object in question. The XACML (XACML-V3.0 (2012)) specification offers a standardized language for expressing such access control policies. The simplicity of ABAC lies in that attribute-based decisions are more flexible compared to RBAC's rigid structure of roles and permissions. Another advantage of ABAC is that it can accommodate more easily the use of dynamic environmental attributes in access control decisions, such as the time of day and location of a user or object. However, a disadvantage is that a large number of attributes has to be managed and maintained by security administrators as the meaning of an attribute is only substantiated when associated with a user and object in the frame of an access policy rule.

RBAC and ABAC can coexist by considering a role as another attribute of a user, though associating a role with a collection of permissions is not trivial. As such, auditing is more straightforward with RBAC as roles with permissions define what objects a user is allowed to access, whereas auditing with ABAC to understand a user's permission set requires an exhaustive enumeration of all the attributes of the user and those of all objects, as well as an evaluation of all access policy rules.

A key observation for role-centric and attribute-centric access control is that the domain knowledge or semantic meaning of attributes and relationships is directly encoded either in the database schema (RBAC) or in the policy rules that govern the access control decisions (ABAC). This is a challenge for cross-organizational collaboration scenarios where interoperability is key. The lack of common pre-existing schemas defining and externalizing all identity and attribute relationships upfront, is the main reason why federated authorization suffers from misalignments at different levels:

- **Attribute**: Missing, inconsistent naming or different semantic meaning of identity attribute types and values
- **Relationship**: Different role-permission assignments or mismatching partial orders in the role hierarchy

Such misalignments for users and objects will be harder to resolve in a world of the Internet of Things where identity relationships will be much more transient and implicit.

4.2. *Reconciliation of cross-organizational attributes and identity relationships*

To address the aforementioned concerns and challenges, we present a framework to externalize domain knowledge in order to dynamically reconcile attributes and identity relationships. This framework eliminates the need for common standardized attributes, hereby simplifying interoperability between business partners. The framework facilitates policy-based access control across organizations and the administration of dynamic entitlements through inference of transient and implicit attribute relationships.

For Peer Review Only

4.2.1. Partial ordered sets of users and objects

As identity relationship management is scaling out to support identities for both users and objects that interact with one another (possibly on behalf of their owner), we must revise the level of abstraction of users and objects to establish the kind of roles and relationships they have with one another and how that affects access control decisions.

The first construct we support in our framework is the definition of partial ordered sets U and O, respectively for sets of users u_i and sets of objects o_i:

$$u_1, u_2, u_3, ... \in U \qquad o_1, o_2, o_3, ... \in O \tag{1}$$

where each element u_i of the partial ordered set U represents a (labeled) set of users, and each element o_i represents a (labeled) set of objects. We define the binary relation $\leq$ between elements as *a subset of*, such that U is a partially ordered set if for each element $u_i \in U$ the following holds:

$$u_i \leq u_i \qquad \text{(reflexivity)} \tag{2}$$

$$u_i \leq u_j \wedge u_j \leq u_k \Rightarrow u_i \leq u_k \qquad \text{(transitivity)} \tag{3}$$

$$u_i \leq u_j \wedge u_j \leq u_i \Rightarrow u_i = u_j \qquad \text{(anti-symmetry)} \tag{4}$$

A set of users u_i can be defined in terms of its individual constituents, as well as in terms of other user sets u_j, as illustrated below:

$$u_1 = \{bob, alice\}$$
$$u_2 = \{u_1, eve\}$$
$$u_3 = \{alice, bob, mallory\}$$

For the above example, we can infer that $u_1 \leq u_2$ and $u_1 \leq u_3$. However, as there is no subset relationship between u_2 and u_3, we say that the pair of elements u_2 and u_3 is *incomparable*. Similar definitions and properties also hold for the partial ordered set O of object sets o_i.

The notion of partial ordered sets plays a vital role to infer inheritance and permission relationships across different sets of users and objects. Later on, we will illustrate how these partial ordered sets are used for reconciling identities based on their attributes.

4.2.2. Equivalence of attribute types and values

Users and objects are annotated with attributes of a particular type, each having a certain value. Consider an enterprise with an identity management system holding the following sets of LDAP attributes for a user *bob*:

bob.userName = 'bob'
bob.givenName = 'Bob'
bob.sn = 'Builder'
bob.mail = 'bob@enterprise1.com'
bob.postalAddress = '221B Baker Street'
bob.city = 'London'
bob.postalCode = 'NW1 6XE'
bob.country = 'UK'
bob.role = 'worker'

To align this attribute set with those of an identity management system of another

For Peer Review Only

organization, our framework is able to align both attribute types and values by declaratively defining equivalence relationships. This is illustrated in the examples below:

(1) Equivalent attribute types:

givenName ≡ *firstName*
sn ≡ *surName* ≡ *lastName* ≡ *familyName*
postalCode ≡ *zipCode*

(2) Equivalent attribute values:

'UK' ≡ *'United Kingdom'*
'worker' ≡ *'labourer'*

The definition of equivalence relationships simplifies federated authorization in cross-organizational collaborations between business partners where the identity provider of *Enterprise 2* provides attributes that need to be aligned with those used in the access control policies of *Enterprise 1*, and vice versa (as depicted in Figure 1).

For Peer Review Only

4.2.3. Implicit definition of user and object sets

The attributes types and values can also be used to define implicit user and object sets by constraining a particular attribute value, rather than explicitly enumerating all relevant users and objects. For example, our framework can declare the set of workers as a new user set as follows:

workers ≡ (?.role = 'worker')

This user set may initially represent a certain group of individuals, including the user *bob*. However, if other employees would later be added with a role *labourer*, then due the semantic equivalence of the role attribute, the implicit user set *workers* would automatically include these users for the computation of any access control decision. As a result, user sets as well as object sets can be defined through explicit membership *assignments* as well as through implicit attribute *associations*.

To express security concerns like dynamic separation of duty, we rely on the above construct to infer that two user sets with conflicting roles are disjoint.

developers ≡ (?.role = 'developer')
testers ≡ (?.role = 'tester')
developers ∩ testers ≡ ∅

The above example is a typical Chinese wall policy which states that individuals developing and testing software features should belong to different teams in order to avoid any conflict of interest. The implicit definition of user sets is used to assert that an individual should never activate both roles at the same time.

4.2.4. Permissions as user-action-object triples

Compared to ABAC specification languages like XACML that use conditions to find relevant policies and rules to evaluate access control decisions, our framework expresses permissions by means of associations between *users*, *objects* and *actions*. The following

example illustrates the definition of a permission as a triple:

$$u_1 = \{bob, alice\} \quad \text{(user set)}$$
$$a_1 = \{read, write\} \quad \text{(action set)}$$
$$o_1 = \{doc1, doc2\} \quad \text{(object set)}$$
$$p_1 =< u_1, a_1, o_1 > \quad \text{(permission)}$$

The advantage of this approach is that relevant permission definitions for users or objects can be easily retrieved by object references. The framework can further generalize the definition of permissions p_i to a set of objects o_j (rather than a set of users u_j) executing a set of actions a_k on another object set o_l, i.e. $p_i =< o_j, a_k, o_l >$.

4.2.5. Policies as partial ordered sets

Our framework defines a policy as a set of permission triples s_i, such that an action is granted if there is at least one permission $p_j \in s_i$ in the set that matches with the request:

$$s_i = \{p_1, p_2, ...\} \quad \text{(permission set)}$$
$$s_1, s_2, ... \in S \quad \text{(partial ordered set)}$$

Policies enable the logical grouping of permissions, and the security administrator decides which permission set s_i must be activated. When multiple permission sets $s_i, s_j, ...$ are activated for a particular object, then all these permission sets must grant the requested action for that particular user. Otherwise the action is denied.

Rather than overriding permissions with combination algorithms, our framework relies on the definition of permission sets as partial ordered sets to avoid sophisticated conflict resolution schemes:

$$s_0 = \{p_1, p_2\}$$
$$s_1 = \{p_3, p_4, p_5\}$$
$$s_2 = \{s_0, s_1\}$$

By activating the permission sets s_0 and s_1, our framework evaluates the following access control decision:

$$s_0 \wedge s_1 = (p_1 \vee p_2) \wedge (p_3 \vee p_4 \vee p_5)$$

whereas by activating s_2 it evaluates:

$$s_3 = s_0 \vee s_1 = (p_1 \vee p_2 \vee p_3 \vee p_4 \vee p_5)$$

Denying access or prohibiting the execution of certain operations can then be achieved by declaring an empty action set a_i for a combination of users u_j and objects o_k.

$$a_i = \{\}$$
$$u_j = \{mallory, eve\}$$
$$o_k = \{x\_files, trade\_secrets\}$$
$$p_l =< u_j, a_i, o_k >$$

URL: http://mc.manuscriptcentral.com/teis Email: eis@odu.edu

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 4. Figure captions are saved separately, and listed at the end of the manuscript.

This way the security administrator has the ability to logically compose permission sets as partial ordered sets with a clear definition on how the combination of policies can affect the outcome of an access control decision.

4.2.6. Dynamic entitlements

In our framework, dynamic entitlements are fine-grained access entitlements or privileges that are administered depending on the situation at hand (e.g. a continuously evolving temporary business relationship). These are particularly useful for networked production where the action permission is restricted by the location of the user or the manufacturing system within the production facility. By grounding users and machines to a particular location (v. Cleeff, Pieters, and Wieringa (2010)), attackers have more difficulty in gaining unauthorized access to sensitive information.

on_premise ≡ (?.location = 'enterprise1')

As such, dynamic entitlements are achieved through constraints on dynamic (environmental) attributes for users and objects, in a similar way as elaborated in the previous subsection on the implicit definition of user and object sets. The only distinction is that these attributes are provisioned externally, and not by any identity management system.

4.2.7. Administrative privileges

By design, our framework also supports dynamic provisioning and customization of the user and object sets by means of declaring administrative privileges that are enforced by the access control framework itself.

managers ≡ (?.role = 'manager')
a_1 = { add, remove }
u_1 = { bob, alice }
p_1 = < managers, a_1, u_1 >

This example states that managers have administrative privileges to add or remove entities to the given set of users. Compared to the XACML reference architecture where policy administration is a dedicated component (i.e. the *Policy Administration Point*), is policy administration directly integrated into and enforced by our framework.

5. Practical implementation

In this section, we present the practical implementation of our reconciliation framework for identity access and relationship management. We discuss details about the technical realization for cross-organizational manufacturing processes in which the authenticity of identity for users, machines and processes in networked production is verified.

Each production facility deploys its own identity management system, including their own data schema to store user and object attributes. Our framework builds on top of ForgeRock's OpenIDM framework (see Figure 4), a role provisioning and entitlement management platform part of ForgeRock's Open Identity Stack for high available and large-scale, mission-critical deployments[1] including the Internet of Things.

[1] https://www.forgerock.com/resources/high-availability-internet-things/

URL: http://mc.manuscriptcentral.com/teis Email: eis@odu.edu

For Peer Review Only

5.1. *Identity provisioning for users and objects*

Our proof-of-concept framework integrates with OpenIDM and its CRUDPAQ-based REST interfaces to provision and retrieve managed objects, such as users and devices. In practice, all entities (user, devices and services) involved in networked production can initiate such requests. Provisioning a new user as a JSON object is done with an HTTP POST request using the administrator's credentials of OpenIDM, as shown with the *curl* command-line utility in Listing 1:

```
admin@work:~$ curl \
  --header "Content-Type: application/json" \
  --header "X-OpenIDM-Username: openidm-admin" \
  --header "X-OpenIDM-Password: openidm-admin" \
  --request POST \
  --data '{
  "userName":"bob",
  "givenName":"Bob",
  "sn":"Builder",
  "mail":"bob@enterprise1.com",
  "postalAddress":"221B Baker Street",
  "city":"London",
  "postalCode":"NW1 6XE",
  "country":"UK",
  "role":["worker"],
  "password":"MyS3cret!",
  "description":"A senior employee with enterprise 1",
  "_id":"bob" }' \
  https://host/openidm/managed/user/?_action=create
```

Listing 1 Creating a new user in OpenIDM

To create such a user, the OpenIDM administrator must first configure a data schema that defines the type and validation patterns for all the attributes. To retrieve all the users, one can issue an HTTP GET with the following *curl* command:

```
admin@work:~$ curl \
  --header "X-OpenIDM-Username: openidm-admin" \
  --header "X-OpenIDM-Password: openidm-admin" \
  --request GET \
  https://host/openidm/managed/user/?_queryId=query-all-ids

{"result":[{"_id":"bob","_rev":"1"}],"resultCount":1,
"pagedResultsCookie":null,"totalPagedResultsPolicy":"NONE",
"totalPagedResults":-1,"remainingPagedResults":-1}
```

Listing 2 Retrieving all users from OpenIDM

One can query the details of a particular user with a REST request like this:

```
admin@work:~$ curl \
  --header "X-OpenIDM-Username: openidm-admin" \
  --header "X-OpenIDM-Password: openidm-admin" \
  --request GET \
  https://host/openidm/managed/user/bob?_prettyPrint=true

{
  "_id" : "bob",
  "_rev" : "1",
  "userName" : "bob",
  "givenName" : "Bob",
  "sn" : "Builder",
  "mail" : "bob@enterprise1.com",
  "postalAddress" : "221B Baker Street",
  "city" : "London",
  "postalCode" : "NW1 6XE",
  "country" : "UK",
  "role": ["worker"],
  "description" : "A senior employee with enterprise 1",
  "accountStatus" : "active",
  "effectiveRoles" : [ ],
  "effectiveAssignments" : [ ]
}
```

Listing 3 Retrieving user 'bob' from OpenIDM

The above examples merely demonstrate standard functionality and capabilities offered out-of-the-box by ForgeRock's OpenIDM platform.

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 5. Figure captions are saved separately, and listed at the end of the manuscript.

5.2. *Provisioning user and device to manufacturing facility relationships*

Within Industry 4.0 supported by Cyber-Physical Production Systems, there will be long-standing relationships between many types of devices (machinery, sensors and actuators) and the manufacturing facility. We register these devices (including any globally unique identifiers, shared secrets, certificates and PKI details) through the dashboard of OpenIDM (as depicted in Figure 4) as a new type of managed objects.

For each type of managed object (e.g. user, device, service, facility), we declare a schema with various properties and validation patterns regarding the content of these properties, as well as many-to-many relationships to other managed objects. This way, we can establish trust relationships when a new device – such as a sensor, actuator, or other machinery – joins or leaves the facility. Similarly, we can model which user and services have a relationship with a device. Note that this is not to enforce access control, but rather to establish which security material is needed for authentication and secure communication between these managed objects.

```
admin@work:~$ curl \
  --header "X-OpenIDM-Username: openidm-admin" \
  --header "X-OpenIDM-Password: openidm-admin" \
  --request GET \
  https://host/openidm/managed/device/?_queryId=query-all-ids
```

Listing 4 Retrieving all devices from OpenIDM

Last but not least, each of these managed objects can be queried using REST requests (see Listing 4) in a similar way as in Listing 2 and 3.

5.3. *Identities for legacy production systems*

The whole identity and access management suite for networked production is built on top of ForgeRock's Open Identity Stack that makes use of the following components:

- **OpenIDM**: identity provisioning
- **OpenAM**: authentication and federated single sign on
- **OpenDJ**: high performance and secure directory server
- **OpenIG**: identity gateway acting as a reverse proxy

OpenAM uses OpenDJ as an identity repository and the latter is provisioned through OpenIDM. All the authentication requests to and responses from OpenAM may pass through OpenIG (as depicted in Figure 5). OpenIG is not a mandatory component but it simplifies secure access to remote cloud services for legacy production systems and applications without having to modify them to support state-of-practice authentication and authorization protocols such as OAuth 2.0, OpenID Connect and SAML 2.0. For large scale deployments with many IoT or CPS devices, we use HAProxy[2], a high performance TCP/HTTP load-balancing proxy server.

5.4. *Reconciling attributes and relationships*

The strength of our framework lies in the externalization of relationships between attributes and identities into a domain knowledge model that can be separately managed and maintained by each organization, and adapted by the security administrators towards

[2] http://www.haproxy.org

URL: http://mc.manuscriptcentral.com/teis Email: eis@odu.edu

For Peer Review Only

the attribute definitions used by their identity management systems. To automatically infer implicit relationships, our access control framework leverages the OWL 2 RL[3] ontology language (Grau et al. (2008); Krötzsch (2012)) to formally and semantically represent user and object sets, as well as equivalence, inheritance, transitive, reflexive and symmetric relationships for any of the attributes with those attributes used by the other party.

Defining implicit user or object sets based on attribute restrictions (and not by enumerating its constituents) is fairly straightforward, as illustrated for the implicit *detectives* user set in Listing 5 (using the *OWL Manchester* syntax).

```
/* Semantic reconciliation of attributes and identity relationships */

<http://host/ontologies/2016/5/abac> rdf:type owl:Ontology .

:familyName rdf:type owl:DatatypeProperty ;
    owl:equivalentProperty :sn .

:firstName rdf:type owl:DatatypeProperty ;
    owl:equivalentProperty :givenName .

:givenName rdf:type owl:DatatypeProperty .

:lastName rdf:type owl:DatatypeProperty ;
    owl:equivalentProperty :sn .

:postalCode rdf:type owl:DatatypeProperty ;
    owl:equivalentProperty :zipCode .

:role rdf:type owl:DatatypeProperty ;
    rdfs:domain :UserSet .

:sn rdf:type owl:DatatypeProperty ;
    owl:equivalentProperty :surName .

:surName rdf:type owl:DatatypeProperty .

:zipCode rdf:type owl:DatatypeProperty .

:location rdf:type owl:DatatypeProperty .

:Permission rdf:type owl:Class .

:Set rdf:type owl:Class .

:UserSet rdf:type owl:Class ;
    rdfs:subClassOf :Set .

:ObjectSet rdf:type owl:Class ;
    rdfs:subClassOf :Set .

:Workers rdf:type owl:Class ;
    rdfs:subClassOf :UserSet ,
        [ rdf:type owl:Restriction ;
          owl:onProperty :role ;
          owl:someValuesFrom [ rdf:type rdfs:Datatype ;
                               owl:oneOf [ rdf:type rdf:List ;
                                           rdf:first "worker" ;
                                           rdf:rest rdf:nil
                                         ]
                             ]
        ] .
```

Listing 5 Reconciling attributes and relationships

The datatype properties in the above example (such as *sn*, *givenName*, *surName*, ...) correspond with the attribute definitions of the OpenIDM platform with which our framework integrates. OpenIDM saves the attribute declarations for that particular configuration in a JSON file (usually in the `<openidm>/conf/managed.json` file). Our framework imports this configuration file and translates all attribute definitions into equivalent ones in the above ontology specification language such that reconciliation mappings can be edited with a tool like Protégé (see Figure 6).

The inference engine loads the OWL domain knowledge model with the reconciliation

[3] http://www.w3.org/TR/owl2-profiles/#OWL_2_RL

URL: http://mc.manuscriptcentral.com/teis Email: eis@odu.edu

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 6. Figure captions are saved separately, and listed at the end of the manuscript.

mappings, and leverages the Drools 6.5 rule engine[4] from JBoss to implement the OWL 2 RL ontology profile and semantic reasoning backend. Declarative rules are used to trigger environmental attribute updates for users and objects (e.g. location).

6. Evaluation

This section offers qualitative and quantitative evaluation of our attribute-based access control framework.

6.1. *Qualitative comparison with XACML*

A key advantage of our framework is that it separates the domain knowledge from the rule-based policies of XACML, which makes the reconciliation of attributes and relationships in cross-organizational authorization policies much easier. Subsets of users and/or objects that are otherwise part of the access rule definition are now declared separately, simplifying the definition of a policy towards a logical composition of permission triples. This is a key benefit compared to XACML-based ABAC policies for which it is not trivial to ascertain which policies apply for a given access request. The XACML specification incorporates dedicated language constructs (i.e. *targets*) to identify which policies and rules are relevant to compute an access control decision.

With XACML, the decision of multiple relevant policies may conflict with one another, and special rule and policy combination algorithms must be used by the security administrator to define which access control decision overrides the others. This makes conflict resolution with XACML-based access control policies unnecessary complicated compared to the logical composition of permissions in our framework.

6.2. *Performance and scalability evaluation*

In this subsection, we carry out a systematic scalability assessment using the Universal Scalability Law (USL) (Gunther (2007)) demonstrating the performance and scalability of our framework. The USL combines (a) the initial linear scalability of a system under increasing load, (b) the cost of sharing resources, (c) the diminishing returns due to contention, and (d) the negative returns from incoherency into a model that defines the relative capacity $C(N)$:

$$C(N) = \frac{N}{1 + \alpha(N-1) + \beta N(N-1)} \tag{5}$$

where N represents the scalability of the software system in terms of the number of concurrent access control requests, α represents the contention penalty, and β defines the coherency penalty, with $0 \leq \alpha, \beta < 1$. To benchmark the scalability, N is incremented on a fixed configuration.

In our experimental setup, we deployed OpenIDM and our framework on a Dell PowerEdge R620 server with 32 Intel Xeon E5-2650 CPU cores running at 2.00GHz, and

[4]http://www.drools.org

URL: http://mc.manuscriptcentral.com/teis Email: eis@odu.edu

For Peer Review Only

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 7. Figure captions are saved separately, and listed at the end of the manuscript.

Figures are not embedded in the manuscript (cfr. instructions for authors).

Figure 8. Figure captions are saved separately, and listed at the end of the manuscript.

64GB of memory connected to a 1 Gigabit network. This server evaluated access control decisions against a set of 100 artificial permission definitions based on random subsets from collections of 1000 users, 1000 objects and 10 actions. The access control policies used user and object attributes that required reconciliation against the attributes sent by the client, as outlined in section 3.2.2.

We simulated an increasing number of concurrent access requests from users, devices, services, etc. against the REST interfaces of our framework with a mix of read/write operations. The *read* operations represent access requests without any update in the user attributes. The *write* operations represent access requests where a user attribute was changed such that the implicit user sets and other relationships require reconciliation, hereby triggering the reasoning engine in our framework to infer any relevant attribute and identity relationships.

Figures 7 and 8 show the scalability of the system with a growing number of concurrent entities that each submit access requests to the server hosting our framework at a rate of 1 per second, with respectively a 95/5 and 50/50 read-write ratio. For the first deployment and configuration, we reach near linear scalability up to about 6000 requests per second, whereas for the second configuration we start to deviate from linear scalability at around 1600 requests per second. The performance drop can be explained by the fact that attribute updates by the client trigger the reasoning engine to reconcile the attributes again. Nonetheless, this experiment validates the technical feasibility for deploying our solution, especially when one considers that *write* requests are less likely compared to *read* requests, and that our solution can be scaled out horizontally through sharding on the object identifiers for which access is requested.

We also carried out a preliminary performance comparison with a XACML3-based policy engine[5]. We noticed a 30% performance improvement for our framework in benchmark scenarios with large and complex policy rules. While we could not generalize the performance gain overall, we believe that this positive effect was due to the fact that the XACML3 engine had to re-evaluate with each access requests which policies were applicable (i.e. the target clause) and the complete evaluation of the rules as well. By externalizing the domain knowledge and defining implicit user and object sets, much of the equivalent computations in our framework may have been cached across different access requests. A more in-depth performance analysis and comparison is required to validate these observations.

7. Conclusion

Linking production facilities to the Internet and connecting them to the cloud for remote monitoring and data analysis opens them up to severe security threats, ranging from sabotaging critical infrastructure from the outside, to unauthorized access to sensitive customer data and industrial espionage from within. Especially in production networks that operate across the organizational boundaries of the enterprise, the number of potential targets for attack increases.

[5]https://github.com/wso2/balana

For Peer Review Only

We presented an attribute-based access control proof-of-concept framework that supports policy reconciliation in authorization scenarios that cross the organizational trust boundaries of the enterprise. Our framework targets applications where different collaborating enterprises grant each other limited access to shared business services and data depending on the contextual circumstances. The main challenge that we address is the managerial overhead for security architects when non-standardized attributes are used across identity management systems and attribute-based access control policies that convolute cross-organizational authorization. Our solution facilitates attribute and policy reconciliation during the evaluation of access control decisions. By inferring attribute relationships, our framework aligns dynamic entitlements and identity relationship management to accommodate disparate attribute usage across enterprise security systems. The prototype operates on top of a state-of-practice identity and access management platform, and is validated for a motivating scenario on networked production where such cross-organizational collaborations are quintessential. By systematically benchmarking our framework against the Universal Scalability Law, we were able to demonstrate the practical performance of our solution.

8. Acknowledgments

Anonymized

References

Andresen, Lasse. 2014. "Open Sourcing the Future of IAM." *Netw. Secur.* 2014 (9): 18–20. http://dx.doi.org/10.1016/S1353-4858(14)70094-7.

Bencsáth, Boldizsár, Gábor Pék, Levente Buttyán, and Márk Félegyházi. 2012. "The Cousins of Stuxnet: Duqu, Flame, and Gauss." *Future Internet* 4 (4): 971. http://www.mdpi.com/1999-5903/4/4/971.

Coyne, Ed, and Timothy R. Weil. 2013. "ABAC and RBAC: Scalable, Flexible, and Auditable Access Management." *IT Professional* 15 (3): 14–16.

Damianou, Nicodemos, Naranker Dulay, Emil Lupu, and Morris Sloman. 2001. "The Ponder Policy Specification Language." In *Proceedings of the International Workshop on Policies for Distributed Systems and Networks,* POLICY '01. 18–38. London, UK, UK: Springer-Verlag.

Ferraiolo, David, Ramaswamy Chandramouli, Rick Kuhn, and Vincent Hu. 2016. "Extensible Access Control Markup Language (XACML) and Next Generation Access Control (NGAC)." In *Proceedings of the 2016 ACM International Workshop on Attribute Based Access Control,* New Orleans, Louisiana, USA. ABAC '16. 13–24. New York, NY, USA: ACM. http://doi.acm.org/10.1145/2875491.2875496.

Grau, Bernardo Cuenca, Ian Horrocks, Boris Motik, Bijan Parsia, Peter Patel-Schneider, and Ulrike Sattler. 2008. "OWL 2: The Next Step for OWL." *Web Semant.* 6 (4): 309–322. http://dx.doi.org/10.1016/j.websem.2008.05.001.

Gunther, Neil J. 2007. *Guerrilla capacity planning - a tactical approach to planning for highly scalable applications and services.*. Springer.

Hermann, Mario, Tobias Pentek, and Boris Otto. 2016. "Design Principles for Industrie 4.0 Scenarios." In *HICSS,* 3928–3937. IEEE Computer Society.

Jin, Xin, Ram Krishnan, and Ravi Sandhu. 2012. "A Unified Attribute-based Access Control Model Covering DAC, MAC and RBAC." In *Proceedings of the 26th Annual IFIP WG 11.3 Conference on Data and Applications Security and Privacy,* Paris, France. DBSec'12. 41–55. Berlin, Heidelberg: Springer-Verlag.

Kandala, S., R. Sandhu, and V. Bhamidipati. 2011. "An Attribute Based Framework for Risk-Adaptive Access Control Models." In *Availability, Reliability and Security (ARES), 2011 Sixth International Conference on,* 236–241. Aug.

For Peer Review Only

Karnouskos, S., A. W. Colombo, T. Bangemann, K. Manninen, R. Camp, M. Tilly, P. Stluka, F. Jammes, J. Delsing, and J. Eliasson. 2012. "A SOA-based architecture for empowering future collaborative cloud-based industrial automation." In *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society,* 5766–5772. Oct.

Kemény, Zsolt, Elisabeth Ilie-Zudor, and László Monostori. 2009. "From Tracking Operations to IOT: The Small Business Perspective." In *Proceedings of the 14th IEEE International Conference on Emerging Technologies & Factory Automation,* Palma de Mallorca, Spain. ETFA'09. 1342–1349. Piscataway, NJ, USA: IEEE Press. http://dl.acm.org/citation.cfm?id=1740954.1741138.

Krötzsch, Markus. 2012. "OWL 2 Profiles: An Introduction to Lightweight Ontology Languages." In *Reasoning Web. Semantic Technologies for Advanced Query Answering - 8th International Summer School 2012, Vienna, Austria, September 3-8, 2012. Proceedings,* Vol. 7487 of *Lecture Notes in Computer Science* edited by Thomas Eiter and Thomas Krennwallner. 112–183. Springer. http://dx.doi.org/10.1007/978-3-642-33158-9_4.

Langner, Ralph. 2011. "Stuxnet: Dissecting a Cyberwarfare Weapon." *IEEE Security and Privacy* 9 (3): 49–51. http://dx.doi.org/10.1109/MSP.2011.67.

Lee, Jay, Behrad Bagheri, and Hung-An Kao. 2015. "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems." *Manufacturing Letters* 3: 18 – 23. http://www.sciencedirect.com/science/article/pii/S221384631400025X.

Mahalle, Parikshit N., and Poonam N. Railkar. 2015. *Identity Management for Internet of Things.* Wharton, TX, USA: River Publishers.

Ni, Qun, Elisa Bertino, and Jorge Lobo. 2010. "Risk-based Access Control Systems Built on Fuzzy Inferences." In *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security,* Beijing, China. ASIACCS '10. 250–260. New York, NY, USA: ACM.

Nicholson, A., S. Webber, S. Dyer, T. Patel, and H. Janicke. 2012. "SCADA Security in the Light of Cyber-Warfare." *Comput. Secur.* 31 (4): 418–436. http://dx.doi.org/10.1016/j.cose.2012.02.009.

Perkins, Earl, and Ant Allan. 2015. "The Identity of Things for the Internet of Things." Wharton, TX, USA. http://dx.doi.org/10.1109/MSP.2011.67.

Sandhu, R.S., and P. Samarati. 1994. "Access control: principle and practice." *Communications Magazine, IEEE* 32 (9): 40–48.

Sandhu, Ravi S. 1993. "Lattice-Based Access Control Models." *Computer* 26 (11): 9–19.

Sandhu, Ravi S., Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. 1996. "Role-Based Access Control Models." *Computer* 29 (2): 38–47. http://dx.doi.org/10.1109/2.485845.

Tonti, Gianluca, Jeffrey M. Bradshaw, Renia Jeffers, Rebecca Montanari, Niranjan Suri, and Andrzej Uszok. 2003. "Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder." In *International Semantic Web Conference,* Vol. 2870 of *Lecture Notes in Computer Science* edited by Dieter Fensel, Katia P. Sycara, and John Mylopoulos. 419–437. Springer.

v. Cleeff, A., W. Pieters, and R. Wieringa. 2010. "Benefits of Location-Based Access Control: A Literature Study." In *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom),* 739–746. Dec.

Wang, Shiyong, Jiafu Wan, Di Li, and Chunhua Zhang. 2016. "Implementing Smart Factory of Industrie 4.0: An Outlook." *IJDSN* 2016: 3159805:1–3159805:10. http://dx.doi.org/10.1155/2016/3159805.

XACML-V3.0. 2012. "eXtensible Access Control Markup Language (XACML) Version 3.0. Candidate OASIS Standard 01." http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cos01-en.html.

Yuan, Eric, and Jin Tong. 2005. "Attributed Based Access Control (ABAC) for Web Services." In *Proceedings of the IEEE International Conference on Web Services,* ICWS '05. 561–569. Washington, DC, USA: IEEE Computer Society. http://dx.doi.org/10.1109/ICWS.2005.25.

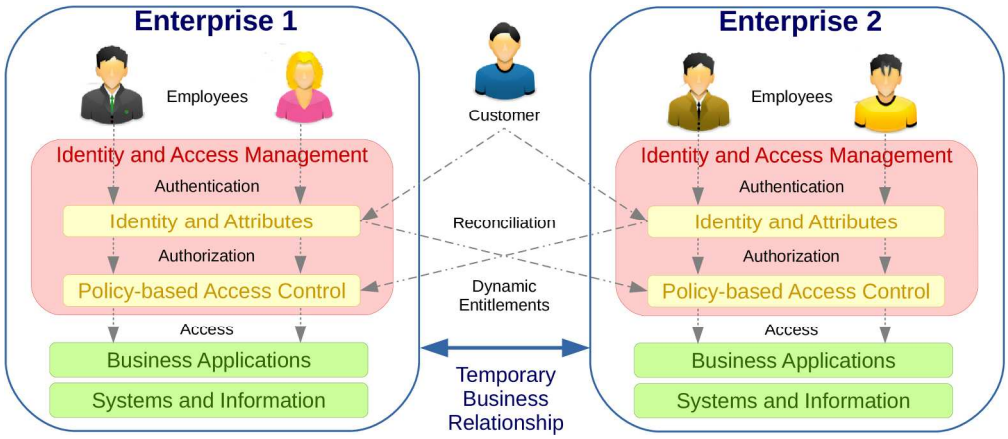
For Peer Review Only

Figure captions

Following the instructions for the authors for this journal (`http://www.tandfonline.com/action/authorSubmission?journalCode=teis20&page=instructions`), we include below the figure captions as a list.

- Figure 1: Identities and relationships in dynamic cross-enterprise collaborations
- Figure 2: Trust boundaries for cross-organizational networked production in Industry 4.0
- Figure 3: Role-based access control metamodel
- Figure 4: Registering devices as managed objects
- Figure 5: Identity management and access control with legacy production systems
- Figure 6: Attribute reconciliations in Protégé
- Figure 7: Systematic scalability analysis for a read-heavy workload (95% read - 5% write)
- Figure 8: Systematic scalability analysis for a write-heavy workload (50% read - 50% write)

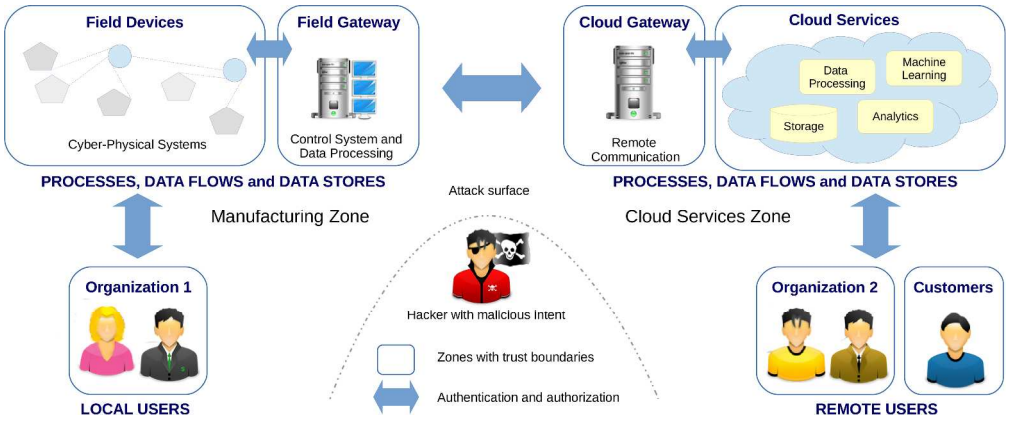
For Peer Review Only



Identities and relationships in dynamic cross-enterprise collaborations

299x128mm (200 x 200 DPI)

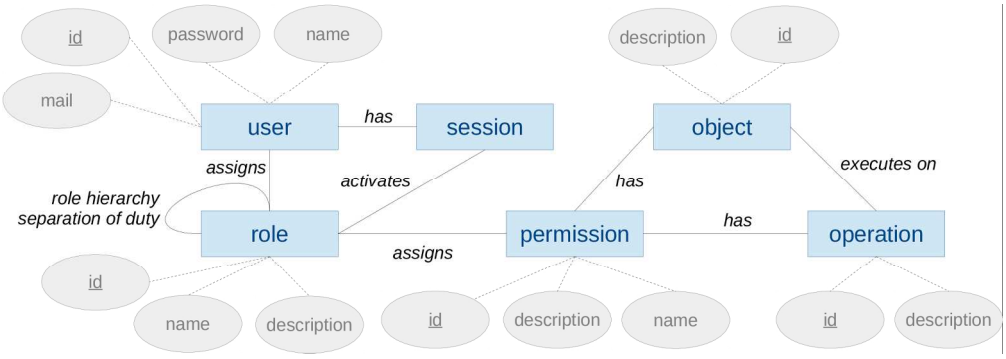
-er Review Only



Trust boundaries for cross-organizational networked production in Industry 4.0

553x226mm (200 x 200 DPI)

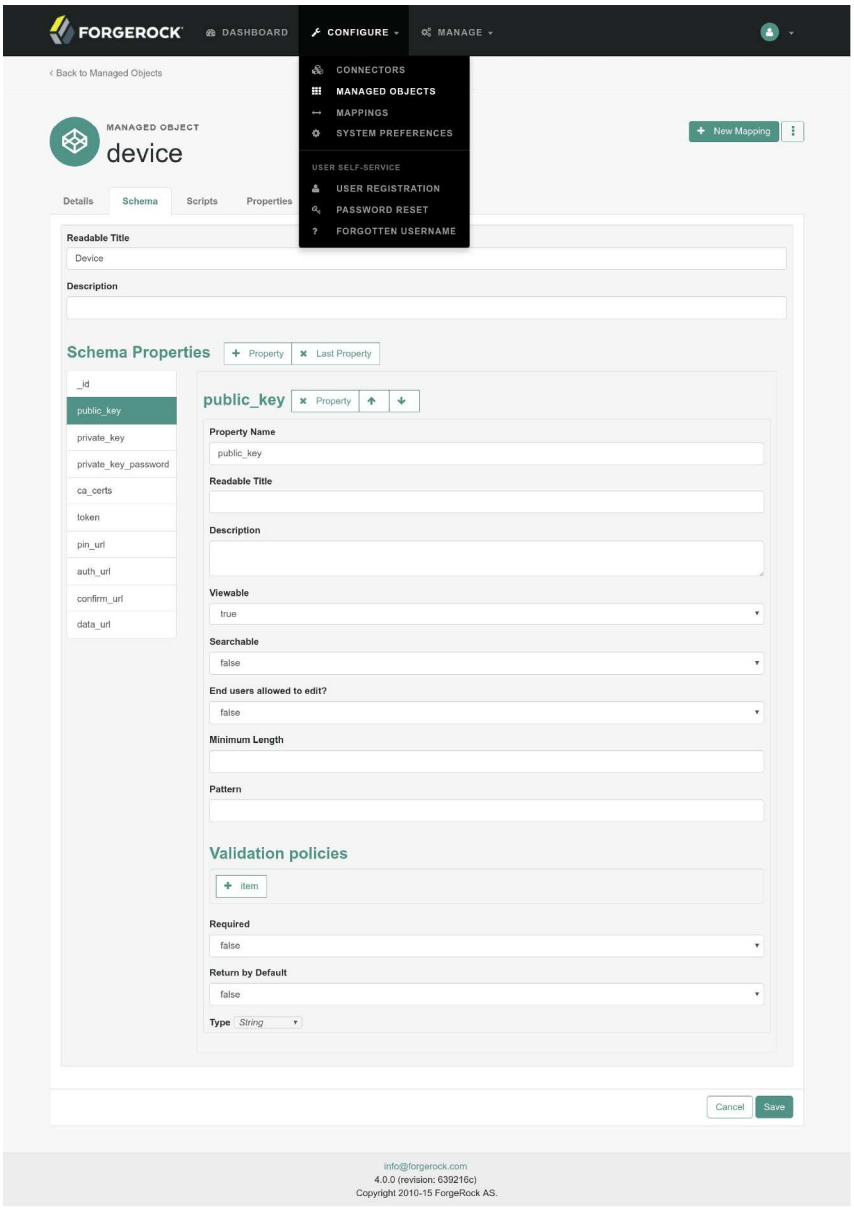
For Peer Review Only



Role-based access control metamodel

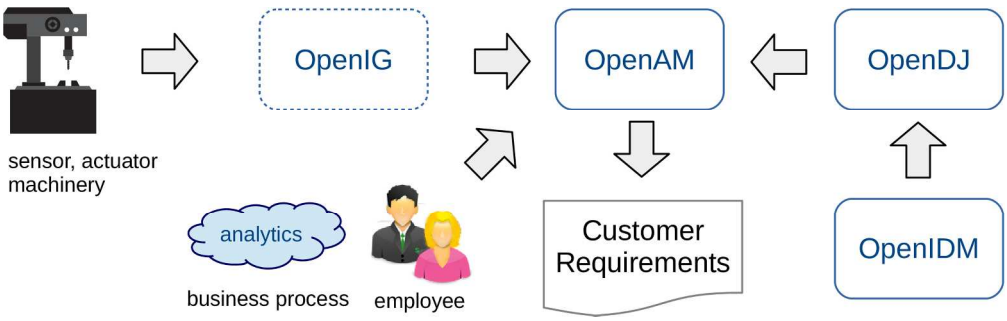
279x97mm (200 x 200 DPI)

Peer Review Only



Registering devices as managed objects

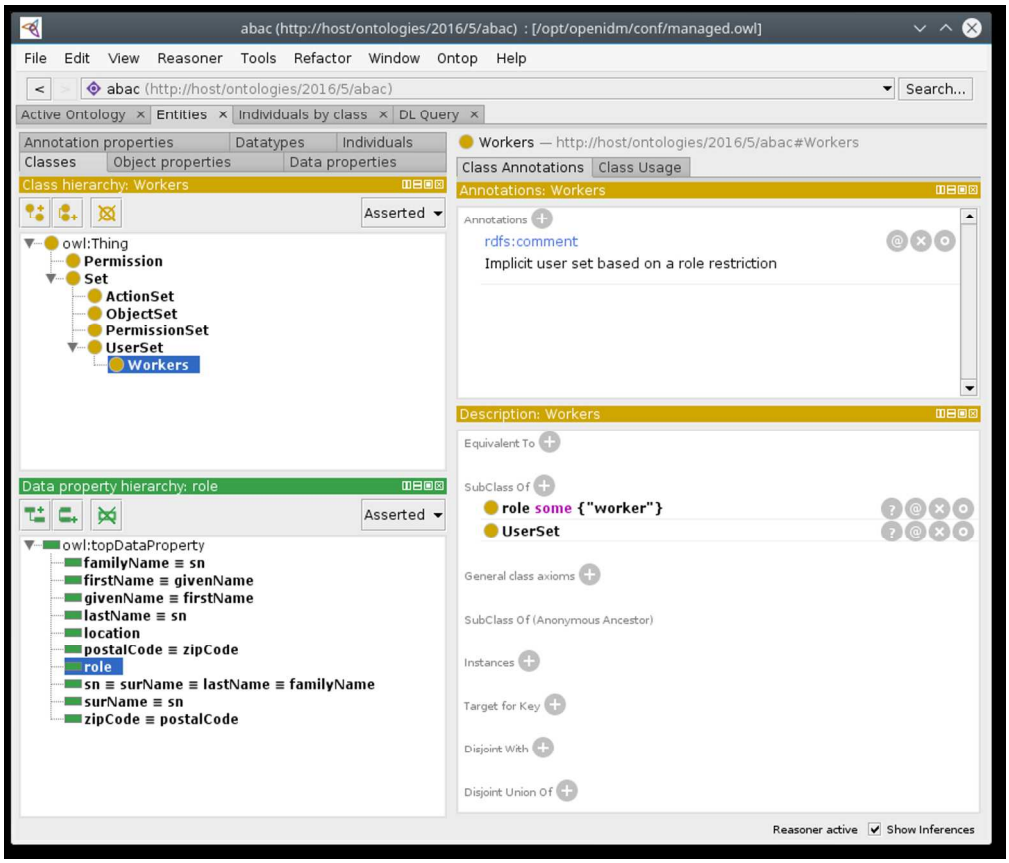
508x719mm (200 x 200 DPI)



Identity management and access control with legacy production systems

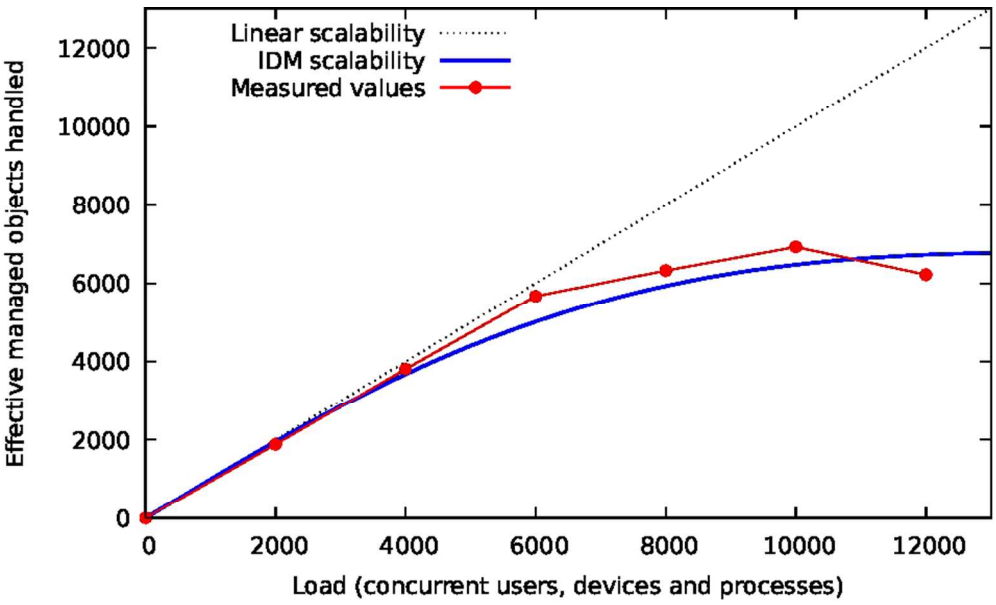
301x93mm (200 x 200 DPI)

Peer Review Only



Attribute reconciliations in Protege

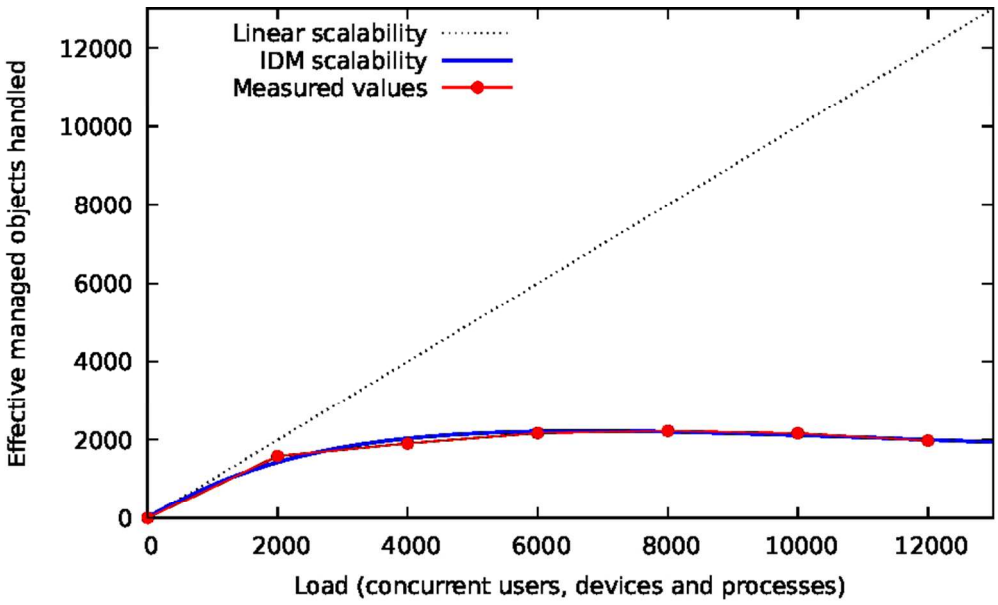
For Review Only



Systematic scalability analysis for a read-heavy workload (95% read - 5% write)

122x72mm (200 x 200 DPI)

For Review Only



Systematic scalability analysis for a write-heavy workload (50% read - 50% write)

122x72mm (200 x 200 DPI)

For Review Only