On-Demand Dynamic Security for Risk-Based Secure Collaboration in Clouds

S. Bertram[1], M. Boniface[2], M. Surridge[2], N. Briscombe[1], M. Hall-May[2]
[1]*University of Southampton IT Innovation Centre,* [2]*QinetiQ Ltd*
mjb@it-innovation.soton.ac.uk

Abstract

Industrial adoption of cloud computing for collaborative business processes is limited by their ability to meet inter-enterprise security requirements. Although some clouds offerings comply with security standards, no solution today allows businesses to assess security compliance of applications at the business level and dynamically link to security countermeasures on-demand. In this paper, we present a Platform-as-a-Service infrastructure that combines semantic security risk management tools with dynamic web service policy frameworks to support the mitigation of security threats throughout the lifecycle of a service-oriented application deployed within the cloud. The platform address the need to model security requirements, dynamically provision and configure security services and link operational security events to vulnerabilities and impact assessments at the business level. The Platform has been evaluated using a collaborative engineering design scenario and a proof-of-concept deployed at a multi-tenant cloud as part of the UK CFMS project. The work is being further enhanced in the European Funded SERSCIS project.

1. Introduction

Cloud computing is the latest in a long line of attempts to support service-based outsourcing and collaborative hosting environments for the extended enterprise. Building on the principle of commoditized IT assets and on-demand usage patterns, clouds are significantly reducing the costs of operating software systems. Today, industrial adoption of public or collaborative clouds is limited by their ability to support governance requirements for QoS management and security. Only through the commoditization of on-demand governance services that adapt to individual business needs will clouds be accepted more widely by industry.

This paper describes a novel service-oriented infrastructure (SOI) being developed as part of the UK TSB CFMS Core Programme [1] and further developed in European Commission ICT SERSCIS project [2]. The SOI aims to provide a Platform-as-a-Service (PaaS) solution for on-demand management of security risks associated with digital assets shared in collaborative clouds. The SOI brings together data management, process management and trust management to allow business decision makers to plan service networks, to assess explicitly the consequences of security decisions and to mitigate emerging threats dynamically.

The paper reviews security requirements for Clouds and support in the marketplace today. The security risk management methodology, tools and services are described. Explicit consideration is made for the layers of governance, policy lifecycle and how risk assessment tools have been combined with dynamic web service policy management to support inter-enterprise security requirements. The results are evaluated in a proof-of-concept from the engineering sector where assurances for the protection of intellectual property rights are mandated for disclosed design data. When then describe how in SERSCIS the model is being enhanced to support collaborative decision making [3] through vulnerability management in interconnected service-based ICT systems.

2. Motivation

Maintaining the value of digital assets is critical for Enterprises participating in the knowledge economy. With the increasing use of the Internet to facilitate communication and collaboration between organizations, it is all too easy to lose ownership and control of knowledge as it is represented as digital information. Once disclosed the economic properties of digital information (non-rivalrous and diffusive) means that disclosure can have severe consequences for competitiveness, revenues and reputation [4]. Consequently, enterprises with a very low tolerance to risk will tend either not share information, often to the detriment of potential business exploitation, or maintain complicated and costly intellectual property arrangements with out-of-band information distribution channels [5].

For adoption of clouds by industry, tools and services are needed to analyze security requirements and status in the context of the following security questions:

a) Does this system satisfy my security requirements given its current state and security configuration?

b) Can the system evolve in ways that may compromise any of these requirements, and what policies could I use to prevent this?

c) How can I recognize if and when such changes do occur, and how can I manage the resulting threats?

To answer such questions the Cloud must be examined at both business and technical levels. Business issues of responsibility, liability, accountability, privacy, data protection and compliance need to be addressed. Cloud users need to devote time and resources to evaluating cloud providers in terms of their ability to support compliance requirements whether internal and/or regulatory in nature, and develop models and systems to support this evaluation both during system design and operations. SAS 70 Type II [6] and ISO 27001 information security standards [7], are emerging as ways to evaluate cloud provider security but they remain high level procedures and not linked to runtime tooling. Other standards such as Payment Card Industry and Health Insurance Portability and Accountability Act do not directly address requirements for cloud providers [8] although some providers do now supply information on how to achieve HIPPA using their clouds [9]. Vendor responses to compliance have seen the likes of Verizon (CaaS), Unisys (Stealth) and Microsoft's Azure support secure practices (e.g. ISO27001) and/or products for the cloud, but these are only scratching the surface. The Cloud Security Alliance and the Jericho Forum are only focused on developing best practices and consensus [10] and no supporting tools exist today to support them.

The technical issues associated with cloud trust and security include the technical requirements for implementing the consequences of trust decisions and trust relationships asserted at the business level. Typical mechanisms used in implantation include tight access controls, secure authentication, appropriate encryption of data, logging and playback, intrusion detection, anomaly detection, encrypted management mechanisms and more. Many technologies exist and to cover them all would go beyond the scope of this paper. Instead we focus on the essential need for policy dynamics across federations considering identity and access control, along with how to communicate requirements between different parties.

For access control, the most widely used distributed security policy architecture and representation is probably XACML, an XML language for expressing security policies, access requests, decisions and an architecture for (intra-domain) distributed policy administration, decision-making and enforcement [11]. XACML 2.0 provides profiles for RBAC and privacy as well as extensions for hierarchies (resources, roles), policy combining algorithms and integration for SAML assertions). However, the model for verifying and combining these policies assumes shared role/resource semantics, so it is difficult to communicate and use policies across domain boundaries or in a multi-stakeholder context. XACML V3.0 (still in development) adds administrative delegation and obligation notion to support decentralized policy administration but it does not offer a solution for the shared role/resource semantics issue, or allow dynamically changing rules for combining policies for evaluation. There are several research efforts to go beyond XACML and address cross-domain and especially dynamic security issues. PERMIS [12] is quite similar XACML supporting delegation and separation of duties, and multiple sources of authority (i.e. trusted sources of policies and user attributes), so it can be used in a cross-domain context but it doesn't support process dynamics as a core policy concept an essential element for compliance in dynamic service-based systems. Communicable policy representations are also an essential requirement. These have been addressed by service annotation models such as WS-Policy [13] which can be used to describe policies for using services (e.g. use of encrypted communications, or inclusion of particular SOAP headers or attribute tokens) and attached to service descriptions so they can be processed by client software when generating requests.

We can see that the combination of business/system compliance and technical level capabilities are needed to support data, process and trust management in Clouds. No cloud provider addresses this capability in a coherent and consistent way. The following section describes our PaaS solution including the application of the CORAS methodology and how semantic risk models can be used to plan, analyze and monitor WS-Trust services and Process-Based Access Control (PBAC) policies [14], specifically focusing on state-based model of processes in which a service can be used, and the definition of roles with respect to this model as policy targets so they can be mapped to externally defined roles of partners through additional mapping' policy rules.

3. Platform Description

3.1 Methodology

The methodology adopted by the Platform is based on the CORAS risk management process [15] and adapted for use with dynamic service based systems. Risk management activities include initial security planning, security service provisioning, active threat identification and threat assessment.

The high-level PaaS components for on-demand cloud security are shown in Figure 1. At the highest layer Operational Security of an Enterprise are responsible for planning the service network considering *Business Requirements* through an initial risk assessment. The task is completed using a Cloud security decision support tool (CS-DST) which provides the user interface for risk management throughout the lifecycle of the service-based applications. A model is produced of the known security

domains, communication paths, threats and vulnerabilities along with mitigations including trust establishment processes, user roles and access rights. The model is represented using semantic specification language so it can be interpreted during other risk management phases. The next task is to deploy the mitigation measures as services and associated policies during the application provisioning to provide *Service Governance.* This action instructs the PaaS provider to provision and configure the SOI from a palette of tools and services from the cloud provider in accordance with the defined security model. In our SOI this includes security token services for identity (X509 or Active Directory) and access control (WS Trust STS), policy enforcement interceptors, policy decision services and intrusion monitoring tools. All tools are connected to the CS-DST to provide security event monitoring and mitigating management trigger actions through WS-Notification. In additional, business processes used to manipulate data assets are also the source for security events. These events are used for security analysis, threat identification, assessment and reaction within the CS-DST with the aim of providing a "live" model of the security state for hosted data assets.

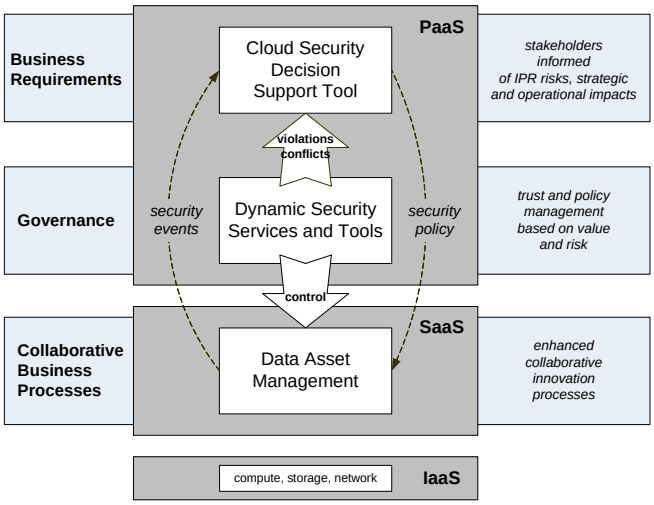

Figure 1. On-demand cloud security services

3.1 Dynamic Security Services

The PaaS solution includes a palette of security services that can be configured and adapt to changing security policies during runtime with minimal loss of functionality and with little or no manual assistance. The services take into account the ability to reconfigure security policy at any time to address events such as shifts in alliances, changes in personnel, changes in the execution environment (e.g. transition from trusted execution environment to untrusted execution environment), and crisis situations. The services support both dynamic identity federation and dynamic authorization policy for fine-grained control of services in the Cloud. The dynamic authorization system takes into consideration different events (e.g. intrusion) when deciding if a certain request should be granted or revoked. This allows exposing different service/component functions according to the context evaluation.

This contextual information is available for the protected security-aware component to allow its extension with adaptive functionality that reacts positively to the environment events (e.g. modifying component behavior) rather than failing in some manner. Contextual policies, as opposed to static ones, offer a dynamic interpretation of authorization rights (i.e. in terms of authorized resource/component operations). The security policy requires that a certain function, that returns sensitive digital assets, be unavailable when an intrusion or attack is detected. Our approach is based on three observations:

1. The meaning of 'security attributes' in Clouds is actually not defined by the issuer, but by service provider(s) when they refer to the attributes in an access control policy. The issuer defines the process for deciding what attributes should be issued to each user, but how this is interpreted is entirely up to the service provider(s).
2. Since the service provider typically doesn't know the rules for issuance, there has to be some way for the issuer to communicate with the service provider to ensure that the policy is consistent with this.
3. Once a service provider understands how to interpret an attribute in their own access policy, they can transfer this understanding to other providers, either directly or by issuing their own 'derived' attributes to the user and communicating their meaning in turn.

Note that the process of making connections between attributes and policy (steps 2-3) is inherently dynamic. New attributes can be defined at any time, and policies can (in some cases must) be updated to handle them as soon as their existence is communicated to the service provider. These basic principles are further elaborated in [16]. Given this, it is possible to use WS-Trust and WS-Federation design patterns to provide a service-oriented architecture for dynamic attribute federation. The key elements in the architecture are: the use of WS-Trust as a standard mechanism for obtaining and validating security tokens encoding identity or other attributes; the use of WS-Federation patterns to orchestrate these token exchanges when a service is accessed; the use of simple attributes in tokens that cross the boundary between domains; the use of simple messages to communicate the meaning of these tokens (through 'policy mappings') between token issuer and token validator.

This last feature establishes (and gives meaning to) the relationship between the two security token services (STS), thus filling the gap in WS-Trust and WS-Federation by explicitly addressing the question of how

'trust' is established in the first place. It provides a flexible way for attributes from one domain to be used in other domains, even if the original identity is local to its provider (e.g. based on Active Directory usernames). It allows extended value chains to be established by passing 'policy mapping' messages up and down the chain as required. The use of 'policy mappings' to and from a simple, cross-domain security attribute token makes it possible to obscure the original identity and attributes as rights are established down the chain of service providers. It is also possible to define as many simplified attributes as one wants to represent arbitrary bundles of access rights as required by the user. Finally, it is still possible to audit service access, by tracing security tokens back up the chain through the accountable providers of policy mappings used across domain boundaries.

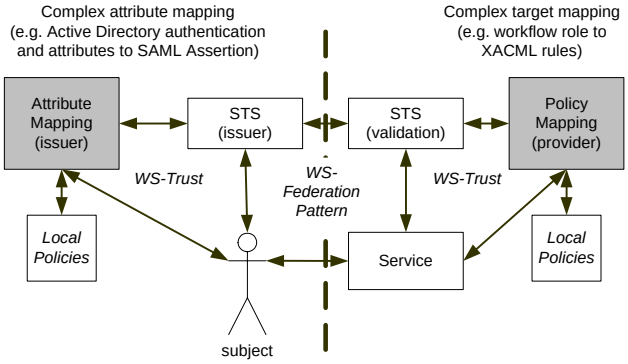

Figure 2. Dynamic policy architecture

The implementation of this model is based on Policy-Based Access Control (PBAC), a dynamic access control mechanism used in GRIA. PBAC provides a uniform approach to trust management and access control. PBAC facilitates simple orchestration and adaptive composition through the delegation and dissolution of trust relationships by changing access control policies. PBAC is based on the XACML architecture including a PEP (Policy Enforcement Point), a PDP (Policy Decision Point) and one or more a PIPs (Policy Information Points).

Conceptually, a service controls a number of resources. Each resource is identified by a unique resource ID, which is a globally unique string (often a URI). Each resource has a workflow state and a resource type. Different operations on the resource are possible in different states. For example, it is not possible to perform the "read" operation on resources of "data stager" type which is in the "empty" state. Each possible operation of a given state has a set of workflow roles (or process role) which may invoke it. The definition of the workflow state model and workflow roles results in what is known as the static policy.

To access an operational a subject requires a security attribute that is mapped to a workflow role. A subject can only invoke the operation if the intersection of these two sets is non-empty (i.e., if they have one of the permitted roles). The rules for determining a subject's workflow roles can be changed dynamically, allowing users to delegate access to a resource to others. Delegation is performed by invoking a delegation service operation, and therefore delegation is also controlled by the policy.

The dynamic policy consists of a set of policy rules. For a user to have a particular workflow role, they must match at least one *Sufficient* rule, all *Necessary* rules, and no *Deny* rules. There are five types of match rules:

Subject DN is ...: This rule will only match someone with a distinguished name (DN) as given by a provided X.509 certificate, which is signed by a specified CA.

Certificate is signed by …: This rule will match anyone whose identity is vouched for by a particular CA.

Has SAML attribute …:. This rule will match anyone with a SAML assertion signed by a specified SAML issuer asserting that they have the specified (name, value) attribute.

Member of group …: This rule will match anyone who is a member of the specified group. Each group is itself a resource, and has its own set of match rules to determine who is a member.

Anyone: This rule matches for everyone

The static policy is defined by a service provider when the service is deployed within the cloud and includes policy mappings from the provider' perspective, for example, mapping a workflow role to a database login. The initial deployment also includes an STS that is used to validate a subjects security attributes against workflow roles even though the specific attributes or issuers have not be defined yet. It is then possible for service consumers, collaborators or resource owners to use high-level system modeling tools to define security attributes and roots of trust for dynamic policies at runtime without the service provider needing redeploy the service or to know the semantics of security attribute in advance. This is described in the next section.

3.2 Security Modelling and Decision Support

Modeling security requirements within the Cloud is planned and monitored using purpose built information security (InfoSec) tools [17]. Underpinning these is an ontology that is used to reason about security risks associated with underlying web service policies. The ontology supports the specification of administration domains, communication links and associated, threats, vulnerabilities and countermeasures associated with information exchanges. Figure 5 shows a graphical representation of a section of the InfoSec ontology and how it relates to web service capabilities and policy.

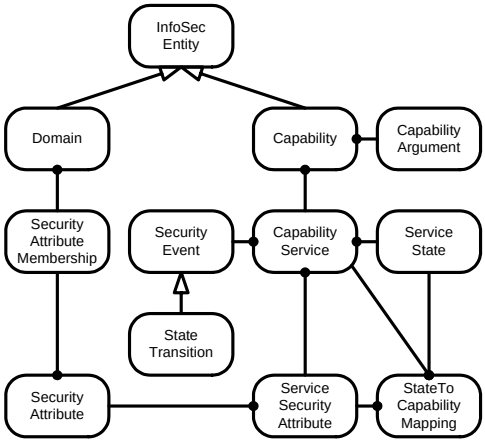

Figure 3. key classes used in policy generation

The InfoSec_Entity is the parent class of all entities in the ontology. It is a core entity of the security model and related to all other entities which are further described in [17]. A Domain is a specific type of InfoSec_Entity and it represents a group of people performing similar operations on a common set of data. A Capability is an abstract concept of a function that can be performed, e.g. "get data from repository" or "add new document to repository". Each Capability has one or more Capability_Arguments, which are abstract definitions of the types of data required to perform the function of the Capability. These arguments are kept abstract as each capability can be implemented in different ways with different services naming their parameters differently or adding additional parameters while still implementing the same Capability. A Capability_Service is a service (e.g. web service) that can deliver a capability. The Capability_Service supports concrete implementations of one or more implementations. Each Capability_Service can have multiple named Service_States. Each state supports a potentially complete subset of the capabilities that are supported by the Capability_Service. One Service_State can also be defined as the initial state for a Capability_Service. Service_States can also have a number of transitions and may allow for transitioning to the final “destroyed” state. State_Transition is a type of security event and is modeled as a simple pairing of two states, defining the states that the transition moves from and to when a given named event occurs. The modeling of states allows the consideration of process context in the security model. Each Capability_Service implements zero or more Capabilities and each Capability can be implemented by multiple Capability_Services, providing alternate implementation or sources for a given capability. The Capability_Service in our implementation has direct bindings to web service operations.

Security_Attribute is a token that can be issued to a subject, for example this could be a workflow Role PBAC’s static policy or SAML attribute in the dynamic policy.. Each Domain can be a member of many Security_Attributes and each Security_Attribute can have many members. Security_Attribute_Membership is a mapping object that associates Domains as members of a SecurityAttributes is the first of two further mapping objects that allow accurate modeling of the mapping of attributes, capabilities, services and states. It specifies which attributes maps are applicable to a given Service. State_To_Capability_Mapping is a second mapping object. It maps Service_Security_Attributes to the capabilities that permission is granted to and the states in which the service must be for those permissions to be granted.

4. Results

In this section we describe the Platform evaluation results based on a scenario of collaborative design of complex engineering products. The scenario includes two companies AerosCo (Prime Contractor) and StoresCo (Supplier) working together for a military customer. AeroCo designs aircraft and StoresCo designs missiles (stores) with each company operating independently lifecycle management for their products. The engineering challenge is for the Prime Contractor and Supplier to ensure safe compatibility between the aircraft and stores during launch using computation fluid dynamics (CFD) simulation techniques. The business challenge is for each company to protect intellectual property in their engineering designs and manage security threats in 3rd party clouds (See Figure 4)

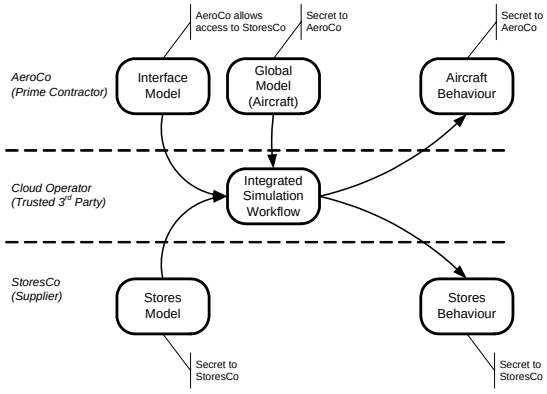

Figure 4. Integrated simulation data model

The Prime Contractor (PC) supplies the Global model of the aircraft and a Pylon interface model used to attach the stores to the aircraft. The Global model needed to remain secret to the PC and the pylon was shared with the supplier. The stores model was provider by the Supplier

and needed to remain secret to the Supplier. The simulation process (meshing, assembly, simulation and post processing) was orchestrated by the PC even though some data within the process could not be accessed by their engineers. The resultant behavior of the Aircraft and Stores needed to be kept confidential to each party. The aircraft behavior could only be accessed by the PC and the Stores behavior could only be access by the Supplier.

The Platform was deployed at the CFMS virtual capability laboratory which provides a multi-tenant Cloud infrastructure supporting collaborative engineering design. The Cloud operator acts as a secure trusted 3rd-party for the integrated simulation. The Cloud consists of computation and storage infrastructure hardware, platform services based on GRIA which is used for the dynamic provisioning of engineering applications, workflows and associated security services [18] and CS-DST based on the InfoSec tools. The SaaS includes Share-A-Space product data management (PDM) system [19] and various CFD analysis codes wrapped as services for model assembly, simulation and post processing. All of the collaborative design processes where orchestrated using Windows Workflow [20].

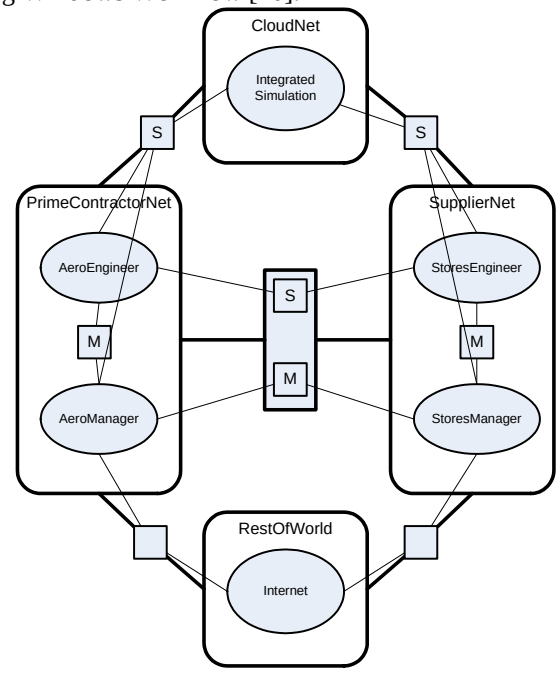

Figure 5. InfoSec tools cloud security domain model

In the scenario, the PC performed an initial security risk assessment for collaboration by describing information security islands, domains and capabilities (e.g ability to produce data assets) along with mitigation measures such as security policies and intrusion monitoring tools). Islands included PrimeContractorNet, SupplierNet, CloudNet and RoW (see Figure 5). Each island had domains defined, for example, PrimeContractorNet has two domains AeroEngineer and AeroManager which mapped directly onto security attributes issued to subjects. For each interaction between domains and islands a connection was defined. Each connection was assigned properties, for example “Web Service”, and a risk assessment performed to identify vulnerabilities, threats and mitigation measures for the information channel. For example, the Supplier needed to deliver the stores model to the PDM system hosted in the IntegratedSimulation domain of CloudNet. A connection was defined between the IntegratedSimulation domain and the StoresManager domain with the capability uploadCAD. Vulnerabilities where defined for the uploadCAD capability in terms of unauthorized access, integrity, and confidentiality. Security countermeasures were selected linking directly to GRIA’s policies and monitoring components. The resultant GRIA static policy defining the capabilities as service operations including uploadCAD is shown in Table 1.

```
<?xml version="1.0" encoding="UTF-8"?>
<state-model description="http://aeroco/Share-A-space"
  xmlns="…./grid/pbac2/staticpolicy/types">
 <state name="UNINITIALISED-STATE">
   <transition><event name="init"/><to-state
name="default"/></transition>
</state>
 <state name="default">
  <operation name="logout">
   <process-role name="Reader"/>
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <operation name="attachAnalysis">
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <operation name="createBillOfMaterials">
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <operation name="downloadCAD">
   <process-role name="Reader"/>
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <operation name="uploadCAD">
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <operation name="login">
   <process-role name="Reader"/>
   <process-role name="Writer"/>
   <process-role name="Owner"/>
  </operation>
  <transition><event name="destroy"/><to-state
name="DESTROYED-STATE"/></transition>
 </state>
 <state name="DESTROYED-STATE"/>
```

```
</state-model>
```

Table 1. Generated PBAC policy

PrimeContractorNet was the root of trust for GRIA WS-Trust security token services issuing AeroEngineer and AeroManager SAML assertions as a mitigation measure for unauthorized access and confidentiality. A similar approach was taken for StoresManager domain within SupplierNet. The resultant roots of trust and SAML attributes were used to generate match patterns associated with GRIA's dynamic policies. For example, the SAML attribute required by the Supplier to access the PDM "uploadCAD" operation is defined in Table 2.

```
<saml:Attribute AttributeName="member-of-group"
AttributeNamespace="http://www.storesco.com/storesco">
<saml:AttributeValue
xsi:type="xsd:string">StoresCoManager</saml:AttributeValue
>
</saml:Attribute>
```

Table 2. Generated SAML attribute

The business requirements asserted that continuous monitoring of vulnerabilities during the collaboration was required. The evaluation focused on monitoring security events associated with dynamic policies and data integrity vulnerabilities. Each monitor acted as a security event source to which the InfoSec tools subscribe using WS-Notification [21]. Each security event triggers an impact analysis that can result in warnings to cloud users and recommendations for further investigation. For example, once uploaded CAD models needed to remain immutable to prove integrity of derived analysis results. Intrusion was identified as a threat for the data integrity vulnerability associated with the Stores model and an integrity monitoring tool was deployed alongside the PDM repository. During the evaluation an intrusion attack on the IntegratedSimulation domain was simulated resulting in corruption of the StoresModel. The attack triggered a security warning within the InfoSec tools, identifying integrity errors with the Stores model. An impact analysis report was generated using data relationships defined within the Bill of Materials of the PDM system to understand which data assets were affected by the corrupted model. The report was then used by the PC to determine if simulation results needed to be recalculated.

5. Future Work

All policy representations at some stage encode trust assumptions that cannot be checked by policy decision processes (however intelligent), but reflect the beliefs of a stakeholder about the other stakeholders in their system. In simple models, these trust assumptions may be limited to a set of trusted sources of identity or other assertions provided by users. Existing web service security standards allow these sources to lie beyond domain boundaries. In more complex cases, trust assumptions are not limited to who can make assertions about external actors, but may cover agreements about the behavior of components. These may be specified in (sometimes machine readable) Service Level Agreements, which form part of the wider frameworks used to describe, access, use and govern services (policies and contracts). Several languages have been developed to represent SLAs, and the most widely adopted so far is WS-Agreement [19], though none of these languages have gained much traction in the distributed systems community. Although not standardised, the SLA specifications used in the GRIA middleware are especially interesting, as they are designed to act as 'trust anchors' for security as well as a basis for management in a cross-domain network of services.

This idea is being further developed in FP7 SERSCIS where the key to the approach is to use semantic models and reasoning to analyze critical infrastructure requirements and vulnerabilities, including ICT failures and the cascading of failures through interconnected ICT systems. These models will then be used to describe, develop and operate service-oriented systems with a high level of security and dependability, capable of automatic adaptation to compensate for faults or changing requirements, or to prevent the spread of disruptive effects between interconnected ICT systems. Governance services are being developed to manage trust relationships via service level agreements that encode dependability commitments between information suppliers and consumers.

6. Conclusions

This paper has shown how novel PaaS software can be used to model, analyze, plan and monitor system security requirements in cloud environments. The techniques applied incorporate security risk assessment tooling, dynamic service provisioning and dynamic web service security infrastructures.

We have shown that to improve adoption of Clouds by industry, business requirements for securing assets in collaborative processes must be closely coupled to technical countermeasures in the platform. To be successful such integration must be facilitated through decision support tools and dynamic security services deployed on-demand and operated actively throughout the execution of distributed business processes. The Platform has been evaluated through a specific application scenario from collaborative engineering sector

which has provided valuable business requirements associated with the protection of intellectual property. The capabilities of the Platform support security concerns throughout the full lifecycle of a service-oriented application deployed within the cloud, this includes: security requirements modeling for distributed cloud business processes; threat mitigation by linking security models to dynamic web service security policies and services; business-level vulnerably monitoring by processing operational security events and using them to trigger impact analysis within data assets.

Overall the Platform demonstrated that for adoption of clouds by industry security cannot be considered just at design time but needs to form part of an overall operator strategy. Operational security experts from all stakeholders, including cloud providers and customers, need to be supported by collaborative decision making tools to assess cloud providers and address business level concerns at runtime. These requirements will be further addressed by linking trust decisions to service behaviors encoded in service level agreements.

7. References

[1] Centre for Fluid Mechanical Simulation Core Programme (CFMS), [Online]. Available: http://www.cfms.org.uk [Accessed: Feb. 11, 2010].

[2] M. Hall-May and M. Surridge, “Resilient Critical Infrastructure Management Using Service Oriented Architecture”, CISIS 2010, The Fourth International Conference on Complex, Intelligent and Software Intensive Systems

[3] EuroControl, “Airport Collaborative Decision Making Implementation Manual,”, December 2008. [Online]. Available: http://www.eurocontrol.int. [Accessed: Jan. 10, 2010].

[4] D. Nasaw, “Upcoming X-Men movie leaked online ahead of release” The Guardian, April 2, 2009. [Online]. Available: http://www.guardian.co.uk. [Accessed Jan. 10, 2010].

[5] M.C. Lee, T. Chang, " Linking knowledge management and innovation management in e-business" International Journal of Innovation and Learning, vol 4, no 2, pp. 145-159, 2007.

[6] The American Institute of Certified Public Accountants," Service Organizations: Applying SAS No. 70, as Amended – AICPA Audit Guide ".

[7] International Organisation for Standardization, “ISO/IEC 27001:2005, Information technology -- Security techniques -- Information security management systems – Requirements”.

[8] Health Insurance Portability and Accountability Act, “45 CFR Parts 160, 162, and 164 Health Insurance Reform: Security Standards; Final Rule”, [Online]. Available: http://www.hhs.gov. [Accessed: Jan. 10, 2010].

[9] Amazon, “Creating HIPAA-Compliant Medical Data Applications with Amazon Web Services”, April 2009, [Online]. Available: http://www.amazonaws.com. [Accessed: Dec. 09, 2009].

[10] Cloud Security Alliance, “Security Guidance for Critical Areas of Focus in Cloud Computing V2.1”, December 2009, [Online]. Available: http://www.cloudsecurityalliance.org. [Accessed: Feb. 10, 2009].

[11] OASIS, “eXtensible Access Control Markup Language (XACML)” V2.0, Feb 1 2005

[12] D. W. Chadwick and A. Otenko, " The PERMIS X.509 role based privilege management infrastructure" in Future Generation Computer Systems Vol 19, Issue 2, Feb 2003, pp 277-289

[13] OASIS, “WS-Policy”, V1.2, Apr 25 2006

[14] Surridge, M., Payne, T. R., Taylor, S. J., Watkins, E. R., Leonard, T., Jacyno, M. and Ashri, R. (2006) Semantic Security in Service Oriented Environments. In: UK e-Science Programme All Hands Meeting 2006 (AHM2006).

[15] F. den Braber, I. Hogganvik, M. S. Lund, K. Stølen and F. Vraalsen, “Model-based security analysis in seven steps — a guided tour to the CORAS method “, BT Technology Journal Vol 25, No 1, Jan 2007

[16] Ahsant, M., Surridge, M., Leonard, T. A., Krishna, A. and Mulmo, O. (2006) Dynamic Trust Federation in Grids. In: The 4th International Conference on Trust Management, 16 - 19 May 2006, Pisa, Tuscany, Italy, published as Trust Management, Lecture Notes in Computer Science Volume 3986/2006, Springer Berlin / Heidelberg, 2006. ISBN 978-3-540-34295-3.

[17] Briscombe, N.J., GoiIlau, P., Sheppard, T, and Watson, G., D3C: A Coherent, Socio-Technical Framework for Identifying, Modelling and Managing Risks in Coalition C2, International Command and Control Research and Technology Symposium, 2007

[18] Dynamic Service Provisioning Using GRIA SLAs, Boniface, M.J.; Phillips, S.C.; Sanchez-Macian, A.; Surridge, M., in proceedings of Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'07), September 17 2007, Vienna, Austria.

[19] Eurostep, Share-A-Space, [Online] http://www.eurostep.com [Accessed: Feb. 11, 2010]

[20] Microsoft, Windows Workflow Foundation, [Online]. Available: http://msdn.microsoft.com [Accessed: Feb. 11, 2010]

[21] OASIS, “WS Notification”, V1.3, Oct 1 2006