PREDYKOT deliverable D2.4

State of the art monitoring

Authority, author and revision history

Rev. N°.	Rev. Date	Prepared By	Reason
0.1	01 June 2013	IRIT – UPS	Initial draft
0.2	14 June 2013	IRIT – UPS	State of the Art – A Study of Monitoring to track impacts on the architecture.
1.0	25 June 2013	IRIT – UPS	Final version

Table of contents

1 POLICY-BASED MANAGEMENT

The main objective of this document is to illustrate the needs of evolution of the policy based management by the market, to address new needs. Replacements or adjustments are suggested to go beyond the obstacles.

1.1 OVERVIEW

Policy-Based Management (PBM) has been defined from different aspects and perspectives. Multiple definitions of this management approach introduce its adaptability towards all domains related to Network Management. Despite this variety, PBM has a core objective; on which all definition agree on, of achieving Business goals towards ensuring the management of networks.

In his paper of (Jude, 2001), Michael Jude qualifies PBM by its influence on Quality of Service and writes “When first conceived in the late 1990s, policy-based network management (PBNM) promised enterprise information technology shops the ability to control the quality of service (QoS) experienced by networked applications and users.” M. Jude keeps on writing: “In fact, the hype went further than that: Vendors promised that CIOs or CEOs would soon be able to control policies through a simple graphical interface on their desk. Behind the scenes, those instructions would translate into specific traffic management adjustments, bypassing traditional network operations.”

On the other hand, Boutaba has presented a more balanced definition in his paper (Boutaba & Aib, 2007). He sees PBM as “A management paradigm that separates the rules governing the behaviour of a system from its functionality. It promises to reduce maintenance costs of information and communication systems while improving flexibility and runtime adaptability. It is today present at the heart of a multitude of management architectures and paradigms including SLA-driven, Business-driven, autonomous, adaptive, and self-* management.”

1.2 BACKGROUND

1.2.1 POLICY

A policy could be defined as the outcome of a big effort that enterprises present on the definition of Business Goals and Requirements. This outcome is represented as set or a list of rules that controls the behaviour of the designed ecosystem. The application domain of an ecosystem is defined in four keywords: Subjects, Actions, Resources and Environments. Rules are currently expressed based on two paradigms:

- Event-Condition-Action (ECA): It is an important paradigm that captures the behaviour of the system and represents it as set of events. An event, or set of events, can trigger logical constraints or conditions to evaluate the behaviour of the

system. In case events met condition(s), there will be list of actions that need to be enforced or applied to readapt the behaviour to the ecosystems' goals.

- Condition-Action (CA): This paradigm is an exception case of the previous one. The need of such paradigm occurs when the rules are only triggered to one-and-only-one type of events. For instance, Access Control List (ACL) is an example of handling only one type of event defined as "Request for Access". The evaluation of rules written based on this paradigm gives usually an answer of Yes/No which reflects if the access is authorized or not.

1.2.2 AUTHORIZATION POLICIES

When authorization becomes an utmost priority, the ecosystem defines a bench of rules that could ensure that only authorized subjects can access the ecosystem's resources. For example, the following rules can be a part of the authorization policy:

- *"The CEO of a company can edit and modify any file on the ecosystem servers. No restriction bounded his rights."*
- *"The Accountable can edit and modify any file on the accounting server, and can edit and modify any file on the shared area of servers. He cannot have access to other servers and, nevertheless, he cannot edit or modify files on them. "*
- *"An employee may edit and modify any file on shared servers. He cannot have access to other servers and, nevertheless, he cannot edit or modify files on them."*

1.3 ACCESS CONTROL

One subtask a Policy-Based Ecosystem should have, among his tasks of controlling the behaviour of its resources, is to control the access to these resources. Access Control System uses authorization policies to ensure that management and usage of the ecosystem application domain are permitted only to the authorized entities. The authorization policy, expressed based on the high-level security policy document, is implemented and formally represented by security models.
From the history of Access Control Models, we distinguish two main categories: Discretional Access Control (DAC) and Mandatory Access Control (MAC).

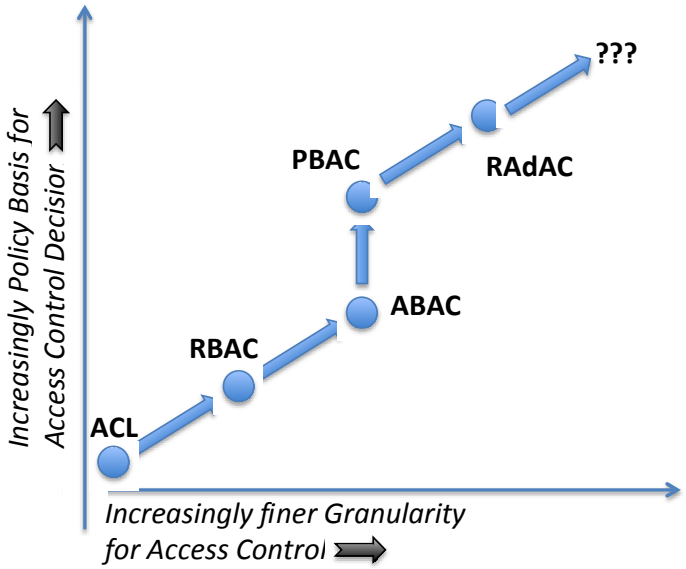
Discretional Access Control (DAC) aims to provide access control capabilities to the discretion of the resource owner or any other entity that should be authorized to control the resource accessing [14,32]. DAC focuses on fine-grained access control to objects through Access Control Matrices and object level permission modes. In DAC policies, access control lists are widely used for file systems access control mechanisms. For instance, in the matrix model state, access control characterizes the rights of each subject with respect to every object in the system [47]. In this model, the protection state of a system can be abstracted as a set of objects O, that is a set of entities that needs to be protected (e.g. resources, files) and a set of subjects S, that consists of all active entities (e.g. users, processes). Further, there is a set of rights R of the form r(s, o), where $s \in S$, $o \in O$ and $r(s,o) \subseteq R$. A right thereby specifies the type of access a subject is allowed to process with regard to an object. An access control matrix lists all processes and files in a matrix. Each row is a process ("subject"). Each column is a file (object). Each matrix entry represents access rights that a given subject has for a given object.

Conversely, MAC focuses on controlling disclosure of information by assigning security levels to objects and subjects, on limiting access across security levels, and consolidating of all classification and access controls into the system. This model is widely applied in the military field Access control where the access control decisions are made by a central authority. Mandatory Access Control is usually associated with the Bell-LaPadula Model of multi-level security (MLS) [47,48]. This model supports mandatory access control by determining the access rights from the security levels associated with subjects and objects. It specifies how information can flow within the system based on labels attached to each subject and object. The subjects and objects are often partitioned into different security levels. It also supports discretionary access control by checking access rights from an access matrix. In this model, processes can read the same or lower security levels, but can only write to their own or higher security levels. The majority of systems that needs to protect multi-level data use the MLS model.

The limitation that can be considered at the first level in these models is their strong attachment to specific properties, categories or metrics. For instance, Bell-LaPadula is concerned about confidentiality and presents a confidential policy to prevent unauthorized accesses. While Mandatory Access Control (MAC) Model is based on the regulations mandated by a central authority. (Boutaba & Aib, 2007)

September 2006, The National Institute of Standards and Technology (NIST) presented in its document [NISTIR7316] list of concepts that every Access Control would need. Inspired by this document, PREDYKOT is interested in definitions related to its implementation of Access Control. Objects definition as entities that store, receive information and implies access to this information (e.g. records, fields (in a database record), blocks, pages, files, directories, network nodes, electrical switches, relays, etc.). Subjects are seen as active entities that represent person, process, or device. They cause information to flow among objects or change the system state [NCSC88]. Operations are active process launched by a subject. Permission (privilege) is the right granted to subject to perform some authorized actions on the ecosystem [FKC03]. Access Control List (ACL) is a list planted inside an object that contains identification of all the subjects that are allowed to access this object. Entries of the list are formed in pairs (subject, set of rights). Each ACL represents a column in the access control matrix. Access Control Matrix is a table that its rows represent subjects' identification and its columns represent objects, and the cells of this table are access rights that links subject to an object. Separation of Duty (SOD) is a principle that implies that under no circumstance a user should have enough permission to abuse the ecosystem. Safety is a concept that measures that the leakage of permissions to an unauthorized principal will not be produced by the access control configuration. A safe configuration is to be true when no permission that leaks to an unauthorized or unintended principal [NISTIR7316].

Here is a historical view about access control models inspired from a workshop held by The National Institute of Standards and Technology (NiST):



To grant happiness to enterprises' business, Role-Based Access Control (RBAC) has been defined to group the access rights based on Roles inside organizations. For instance, a doctor can access patient's information. A trainee doctors, on the other hand, could not modify patient's information. This methodology of identifying or contribute in identifying a subject by its role has reduced the management headache as it groups the characterizations of individuals. But, what if enterprises needed to make the authorizations based on individuals themselves. The presentation of Attribute-Based Access Control (AC) model was a first step toward a generic model for Access Control Systems. The model generality comes from the notion of "Attributes" which aggregate all properties and categories that specifies old models. This abstract view made the expression of policies easier as the most important step became the definition of Attributes that an AC system will handle. ABAC represent a set of attributes that characterise a subject and based on an evaluation process, the subject will be authorized or not to access resources. However, ABAC basically is an extended idea of RBAC towards a generic representation of the policy. Afterwards, a proposition came up with new extension of ABAC, Policy-Based Access Control (PBAC) is said to be a harmonized and standardized image of ABAC that ensure supporting of specific governance objectives at the level of enterprises. PBAC works on combination of three aspects: 1) attributes from the resource, the environment, and the requester (subject) 2) information on the particular set of circumstances under which the access request is made 3) then uses rule sets that specify whether the access is allowed under organizational policy for those attributes under those circumstances. Although ABAC and PBAC are a standard, but the implementation of these models imply a specific representation of languages and architectures. Risk was not well respected in decisions of all previous access control models. Therefore, Risk-Adaptive Access Control (RAdAC) was invented to bring risk-aware access control to enterprises. Assessment of

business crises and the influence of them (risk level) on enterprises' sensitive domains (i.e. finance, economics), invites organizations to assess the risk to protect their IT infrastructure and data. Things ABAC, RBAC and PBAC are not capable to reach in such dynamicity and ability for changes on risk levels [Workshop of NIST project, Survey on Access Control Models].

RBAC and UCON for large scale environments

Several access control systems were built using RBAC for grid computing applications. For instance, PERMIS is a grid authorization system that makes use of RBAC policies where roles define the permissions granted to users, in order to perform actions on grid resources [17]. X.509 Attribute Certificates (ACs) are used to store authorization policies [50,51]. The analysts consider that RBAC is becoming the most commercially adopted access control system in the market [99]. RBAC model is based on the concept of role as a semantic construct on which access policy is formulated [41]. Concretely, the role is used to define subject rights. In the standard definition adopted by the NIST, an RBAC policy consists of a set of users, roles (i.e. a group of permissions associated with resources), permissions (i.e. access rights are composed of operations applied to resources) and sessions (i.e. linking a user to roles), see Fig 1. Therefore, the use of resources is restricted only to the individuals having the authorized roles in the organization. In this model, users are changed frequently but roles remain static. The roles can be assigned to users through the security management system and their activation can be both static and dynamic.

The important features of the RBAC model are related to the management of roles hierarchies, separation of duties, cardinality and dependency constraints [53]. For instance, the definition of SoD constraints on a policy ensures that no user could be able to create a conflict in the security policy through dual privileges like a role for authorizing all payment operations and another role for requesting a payment. However, the standard version of the RBAC model lacks additional concepts for management of access control policy in cloud computing such as the management of delegation from tenant to individual users, obligations management during the collaboration, etc. Sandhu et al, [54], proposed an extension to RBAC that can be useful to handle cloud policies administration for next-generation access control through the definition of the ASCAA concepts namely as abstraction, separation, containment, automation and accountability.

Ni et al, introduced in [55], the P-RBAC family of Privacy aware RBAC models that extends RBAC to support for privacy issues. The P-RBAC core includes a simple language for expressing conditions; they are expressed using context variables. As for the initial version of RBAC, the P-RBAC permissions are assigned to roles and users obtain such permissions by being assigned to these roles. Such privacy-relevant variables record information that has to be taken into account, when enforcing privacy permissions. The distinctive feature of Core PRBAC lies within the complex structure of privacy permissions, which reflects the highly structured ways of expressing privacy rules. While Core P-RBAC has limited expressive power, it remains sufficient for representing public privacy policies, privacy statements and notices of web applications that are compliant with privacy related

acts, such as HIPPA in the case of healthcare applications. On the other hand, conflicts detection in Core P-RBAC remains tractable. In [56], the authors propose adding an obligation model and two efficient decision algorithms that minimize invalid permissions and can compare the applicability scope of two obligations.

However, several privacy concerns remain as an open research issues, such as the interactions between obligations and the execution sequence of obligations; individual exceptions to default policies; compensation and reward mechanisms regarding the status of the fulfilment of obligations, etc. In fact, the standard RBAC model does not provide means to define exceptions to access control rules; these exceptions are very useful for the management of privacy. This implies that the exceptions to a default access control policy that is used to handle privacy should be stored with concerned information themselves. To deal with this issue, Reid et al proposed an RBAC extension that represents general consent with explicit denial, via a new authorization algorithm [57]. The denied role is associated with a negative permission and the allowed role is associated with a positive permission.

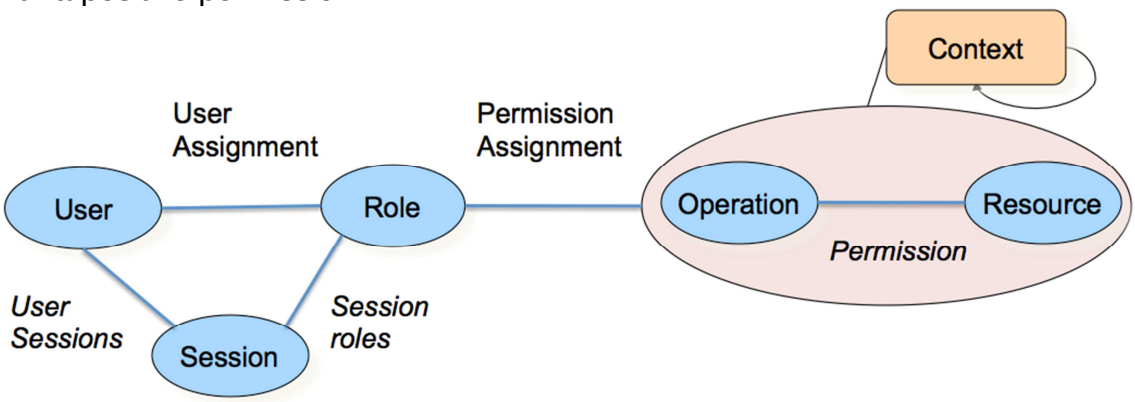


Fig. RBAC core model

RBAC core model matches a centralized management of security, in which a local resource, or a distributed system, implements an access control policy in order to control the access of users. The centralized access control requires from all sites, to agree in advance on the definition and naming of the roles and permissions that are applicable [14,41]. RBAC does not address resources ownership similarly to DAC, in which the creator of a resource determines who can access it. The RBAC model considers that all the resources are owned by the organization that has central control over the resources. This is an issue when we consider the creation of a resource that should be owned by several organizations. In the context of cross organizational sharing data and services in the cloud, even when a resource is created by a user, the organization wants to get a consensus on some level of control over how the resource is to be shared in the cloud. In the RBAC-based models, the composition of access control policies means the definition of a mapping between cross-domain roles. In [58], the authors propose a policy-merging algorithm that merges RBAC policies of n domains to produce a global multi-domain policy. The algorithm iteratively combines the RBAC policies in a pair-wise manner. After n−1 iterations, the access control policies of the n domains are composed to form a global multi-domain policy. For each iteration, the algorithm calls two procedures for the integration and the removal of the redundant roles.

Applying RBAC in cloud computing needs the extension of this model to handle multi tenants policy management issues such as: policy administration, roles mapping and roles delegation. The management of roles delegation is an important issue in the cloud, because it greatly enhances the flexibility and scalability of policies. It may also reduce the control overhead that the organization has to execute for accounting their access control. For this purpose, RBAC core model was extended in dRBAC (distributed Role-based Access Control for Dynamic Coalition Environments) [59]. In dRBAC, the activation of roles changes every time the context changes. Each user terminal has a context agent that detects context changes that trigger transitions between the roles. dRBAC roles represent classes of permissions controlled by their namespace. These permissions can be delegated to other roles (in the same or other namespaces), or entities or authorize a user to act in a given role. In addition, permissions can be delegated in a transitive fashion. A subject S that has been granted the permissions associated with a role R may be able to further delegate R to others, depending on how S was delegated those rights. A sequence of delegations from a subject to an object is referred to as a delegation chain. The access control decision can be based on some additional constraints, such as context constraints [60]. These constraints take as parameters a subject, an action or an object. In most cases, the context supports simultaneously any available and important information that can have an impact on the access control decision. Context information examples are: time, location, authentication factor, etc. A specific extension called GTRBAC (Generalized Temporal Role-Based Access Control) was proposed to support temporal constraints on access control rules [61].

Unlike RBAC, the Attribute Based Access Control (ABAC) model [62,76] defines policy subjects and objects using respectively the identity attributes of both the requester and the resource. It is scalable and flexible and is therefore more suitable for distributed open systems. The access decision is based on identity attributes that the user should prove to have such as social security number, address, birth date, etc. Moreover, ABAC can be considered as a good alternative to rights mismatches when mapping RBAC roles from different domains. However, ABAC requires that collaborating organizations must have an agreement on the meaning of the identity attributes of users that are stored in each domain.

Usage Control (UCON), [32,43], is new access control model that extends and goes beyond traditional trust management, digital rights management and access control models by integrating obligations and conditions, as well as authorizations continuity and the strategies of attributes mutability in covering security and privacy, see Fig 2. Therefore, usage control policy allows systems to enforce the security before the access request, during the use and after the services' use.

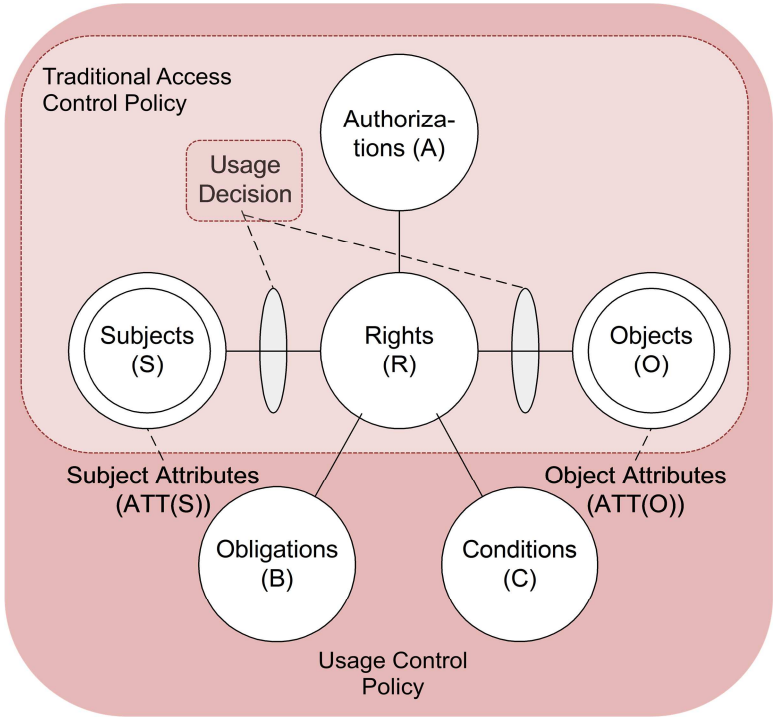


Fig. UCON Usage control model

Usage control allows, in fact, an efficient tractability of services usage in a multi-domain open environment such as the cloud, where domains are loosely coupled and managed independently. Indeed, UCON policies are defined using eight concepts namely: subjects, subject attributes, objects, object attributes, rights, authorizations, obligations and conditions [49]. The authorization, obligations and conditions are components of usage control decisions. The authorization consists of deciding whether to permit a particular form of service and data use. Normal authorization decision can be either permit or deny based on subject and object attributes and conditions. Conditions are system environment and context restrictions that are not explicitly related to subject or object attributes. Obligations are the actions that should be performed by subjects or by the access control system. Unlike traditional access control models such as RBAC or MAC that are applied only on service-side, UCON model is applied on both service provider and consumer sides, in order to guarantee a persistent control, during the usage time and even after. Consumer-side control requires the existence of trusted computing base and a reference monitor. Moreover, UCON can be applied for systems with centralized or decentralized access control such as peer-to-peer systems. Moreover, RBAC model requires that all domains users must be already known by the resource a priori (i.e., user should have accounts that are provisioned through roles), whereas the UCON model does not require from users to be known by the resource a priori. Therefore, the UCON model can cope better with highly distributed environments; this is also due to its support of attributes mutability. In [82], the authors propose to extend UCON in order to handle dynamic authorizations requirements in distributed environments. With respect to the issue of policy administration, RBAC and UCON models support rights delegation and trust relationships, but RBAC is more powerful for detecting and handling policy conflicts, hierarchies and temporal constraints along with revocation of user rights. Moreover, the use of obligations in UCON allows handling the interactive sharing of collaborative cloud services between

different users. For instance, videoconference services where people can share and delegate alternately the control of desktop and applications. UCON has been adopted in GridTrust [63] and Nego-UCON [64] for cloud computing. Unfortunately, unlike the platforms based on RBAC, these UCON platforms suffer from the complexity of the administration of users' identity, especially in highly heterogeneous cross organization environments.

1.3.1 THOUGHTS ON MODELS

The number of implementations and languages exist in order to represent available models annoyed Steve Barker – shortly before his death. He imagined, that one day, policy managers would stand in front of more than 700 access control models in order to control systems access. So that two questions asked in (Barker, 2009) about 1) the possibility of developing or implementing a unified access control meta-model based on the huge number of access control policies and 2) the benefits of having such meta-model.

However, creating a meta-model is an unfinished complicated task that needs effort and time. Moreover, the meta-model becomes more complex each time the integration or the adaptation of a new element (model) is required.

1.3.2 POLICY LANGUAGES & REPRESENTATIONS

Policy languages were independent of a model. To ensure Access Control, security administrators needed to define a model to apply and then use related policies that could respect the model. Then, the research advanced to a level was the models presents its own language and architecture to ease the management of access control.

1.3.2.1 Ponder

Ponder is an Event oriented language that keeps eye on the Quality of Service through preforming monitoring (surveillance) on network behaviours and activities based on predefined metrics (e.g. accuracy, performance, etc.). However, the security side is assured by a special representation of the authorization policy that adapt to Ponder and RBAC requirements (i.e. role-based limitation). Ponder gets a high score when it comes to management, but the lack in supporting issuer limits its capabilities to compete other Access Control implementations.

Four groups represent Ponder's authorization policy: positive authorization, negative authorization, information filtering, restrictions and delegation.

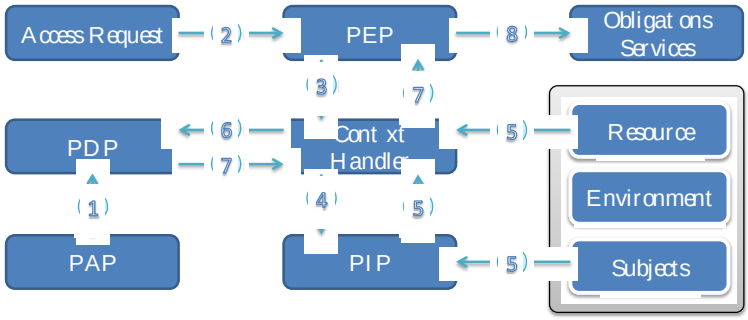
Major points of discussion about ponder could refer to the following: 1) the language does require users (i.e. Policy Writers) to have a level of Java Experience and Modeling, 2) As an RBAC implementation, it does not discuss or speak about the permissions of the object's creator (e.g. in MAC model, the user who created the object can modify its access control lists (Jin, Krishnan, & Sandhu, 2012)).

1.3.2.2 XACML

XACML (eXtensible Access Control Markup Language) is a security oriented policy language created by OASIS[1] in 2003 [Godik03]. The creation of this language based on two main goals: 1) standardize the access control management through using XML and 2) enhance the interoperability between heterogeneous systems. Therefore, the standard XACML provides two languages:

- A declarative language for specifying access control policies: these policies are expressed by a set of rules (could be only one) to be checked or evaluated, and optionally actions to be performed named obligations,
- A query language (request/response) that "allow" or "deny" the requester from performing access to system resources based on predefined policies.

Standardization of Access Control Management is proposed in the language definition through the notions of attributes. The Policy Language is based on applying a set of attributes: a subject (i.e. requester), a target (i.e. network resources or application domain), a role (i.e. policy administrator), and a task (i.e. action to do).



The XACML architecture is based on the association between PEP, PDP and the Policy Repository. The boxes in the figure above are the all components that identified the architecture:

- Context Handler is responsible of ensuring the communication (and translation if necessary) between the PEP and other boxes or entities to retrieve or complete required data for the evaluation process of the policy,
- PAP (Policy Administration Point) to creates and store security policies in the Policy Repository. Usually, it is a user-friendly interface that provides the policy writer (manager or administrator) to write his policy in a domain-specific language (DSL) that is closer to the human language from XACML.
- PIP (Policy Information Point) responsible of storing or retrieving complementary information about attributes' values. This information will be needed for the evaluation of the policy by the PDP,
- The Policy Decision Point (PDP), which takes management decisions. This management agent is independent from the application it manages. It receives requests using a standardized protocol such as COPS, SAML or XACML. These requests are analyzed according to a policy; the PDP may ask for more information

[1] Organization for the Advancement of Structured Information Standards. Its aim is to develop universal standards to facilitate communication within the boundary of e-business.

to take its decision. Finally, the PDP returns its decision, which can be “accept”, “refuse” or “I don’t know”. An obligation section can complete this decision, e.g. “send an email to the administrator after having enforced the authorization decision”.

- The Policy Enforcement Point (PEP) enforces PDP’s decisions. The main role of this agent is to be the interface between the management application and the PDP. Hence, it translates the requests that are expressed in the application specific language into the standardized protocol language understood by the PDP. The PEP can also get additional information that will help the PDP to take its decision. Finally it translates the PDP’s decisions into the application specific language.

Once the initialization phase □1□ terminated by the PAP (i.e. policies are written and stored by the PAP and made available to the PDP in a Policy Repository), policies will be ready for later evaluation. When a subject request for an access □2□, the PEP detects or receives the request and then forwards it to the Context Handler □3□. The latter asks (i.e. if necessary) the PIP to lookup for attributes’ values and necessary data that will be used to evaluate the policy □4□. Once the data is collected and received by the Context Handler □5□, it combines them and sends them to the PDP □6□ who evaluates policy and produce or make its decision. The decision then is transferred to PEP □7□. The PEP will make sure the enforcement of the policy (i.e. authorization or refusal of access request and implementation of appropriate actions specified by the policy named "obligations") □8□.

Today after that OSAIS launched it 3rd version of XACML specification, XACML as an implementation becomes more and more a best option in term of authorization policy. XACML V3 now supports obligations on rules level and adds abstract representation of attributes (i.e. subjects, resources and environment) by using “Any Of” and “All Of”. Moreover, combination algorithms is not anymore bounded with the policy set but with rules level as well. Algorithms supported by this version are Deny-overrides (Ordered and Unordered), Permit-overrides (Ordered and Unordered), First-Applicable and Only-one-applicable (only at the policy level) [eXtensible Access Control Markup Language (XACML) Version 3.0 Committee Specification 01].

XACML is a language for writing policies at a fine-grained level, but it is with highly verbosity. This makes its specification difficult. Despite the fact that XACML was initially dedicated to access control, XACML policies can be aligned to operate with a management orientation, thanks to its attribute-based model. In addition, events management became possible in XACML, as shown [Laborde08].

Finally, XACML stay standing at the front line of Authorization Policies queue with its support for the issuers. XACML issuer is a set of attributes describing the source of a policy. Policy Issuer is defined in a separated administration profile [XACMLAdmin].

```
<xs:element name="PolicyIssuer" type="xacml:PolicyIssuerType"/>
<xs:complexType name="PolicyIssuerType">
  <xs:sequence>
    <xs:element ref="xacml:Content" minOccurs="0"/>
<xs:element ref="xacml:Attribute" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

Digital signatures ensure integrity of XACML statements and not as a method of selecting or evaluating policy. That is, the PDP should not request who signed the policy or whither it's signed or not. However, "***the PDP must verify that the key used to sign the policy is one controlled by the purported issuer of the policy***". [XACML V3.0 Co. Spec. 01]

1.3.3 GOING BEYOND XACML

According to Bonatti et al analysis in [67], we consider the following characteristics in policy mangement:

Support of heterogeneous control models and unknown policies: The policies those are partially described or unknown, but specified and enforced by specific AC systems, such as black boxes. These policies should be queried at run-time to be updated without handling specific features of underlying AC systems.

Controlled interference and update: Avoid undesired side effects resulting in some policy updates decisions that are not correctly reflecting the business collaboration objectives.

High-level expressiveness: the framework should provide a high level of expressiveness that concerns both the policy representation language and the policy administration.

Support of different abstraction levels: The framework should offer different levels of abstraction of the policy rules of each part of the information system and each domain. Sharing several levels of abstraction through an incremental approach should facilitate an analysis, specification, administration and an agreement on the new updates of the policy.

Policy semantics formalization and reasoning: Need formalism to be used in proving and implementing reasoning algorithms on policies and guarantee their correct update.

Extensibility: it concerns how the framework can dynamically support the evolution of policies and their control models over the time.

Context awareness: the framework should be able to take into account contextual information. Several context information types are relevant to adapt the decision of a control system [38]. Examples of contextual information are the user's specific situation, location, time, identity, preferences, and the resources available in the surrounding environment of users. This information can be captured from different physical or virtual data sources to update the policy [48].

Even if the characteristics given above are of high importance for policy management, other characteristics are needed to the multi domain policy management lifecycle and guarantee a better interoperability.

Table 1 gives a comparative analysis of the properties of policy management and languages proposed in the state of the art.

	Ontology based policy languages						XML based policy languages			
	KAOS	Rei	XACML OWL	RBAC OWL	OWL-POLAR	MULTI POL	S4P/SE CPAL	XACML + RBAC +Privacy	XRML	P3P/APPEL
High-level expressiveness	++	++	++	+	++	++	+	+	-	-
Multiple abstraction levels (multi domain)	+	+	+	+	+	++	+-	-	-	-
Support of heterogeneity	+	+	+	+	+	+	+-	+-	-	-
Extensibility	++	++	++	++	++	+	+-	+-	+-	+-
Composition with partially defined local policies	+-	+-	+	+	+-	++	-	-	-	-
Semantics formalization	++	++	-+	+	++	++	+-	+-	-	-
Conflicts and interference	-	-+	+	-+	+	++	-	-	-	-
Trust support	-	-	+-	-	-	+	-	+-	+	+
Attributes mutability	-	-	+-	-	-	+-	-	-	-	-
Delegation support	+-	+	-	-	+-	+	-	-	-	-
Context Awareness	+-	+	+-	-	-	+	-	+-	-	-

Table 1. Comparison of the most representative approaches

2 COMPLEX EVENTS AND SITUATIONS PROCESSING

2.1 COMPLEX EVENT PROCESSING

Complex Event Processing could be applied to a wide range of domains such as: Business Process, Artificial Intelligence, Network Administration, Security of Information Systems, etc. Therefore commercial, academic and free tools are available for different aims and powerful features.

CEP is a process of “Conclusion” driven by events (set of phenomenon)…
It is a Rule-Based Approach, where rules could take many forms: SQL-Like, When-Then, Logical Predicates, …

People call them the nerves, the blood of a network. Events are an important terminology in IT ecosystems. They simply cause an ecosystem to behave!

Keywords: CEP Rules, Business Rules as Situations describe the systems behaviour, Events are no more enough, Complex Events a representation for situations, adaptive and dynamic rules enough to be compatible with other languages.

2.1.1 EVENTS

Oxford Dictionary defines an event as

"Something that happens or is thought of as happening"

Events as defined by following French Dictionaries: LAROUSSE & REVERS:

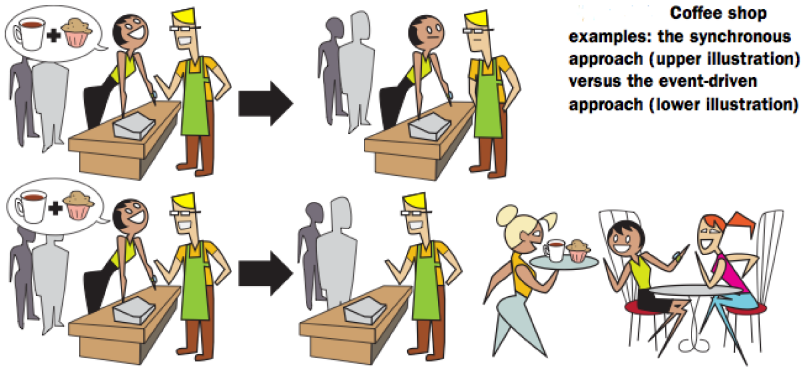
« Tout ce qui se produit, arrive ou apparaît: Fait auquel a abouti une situation, Fait d'une importance toute particulière, Fait marquant de l'actualité ou l'historique »

The following study extracted and made based on information from the Centre of National Resource Textual and Lexical of France (CNRTL). Event is:

- **Fact that leads to a situation**. The emperor has knew, by an event, about all faults of the armistice and he would surely done better by pushing stubbornly forward [Las Cases, Mémor. Ste-Hélène,t. 2, 1823, p. 16] I see people who accompany the army on the map and do not lose more than that if they respond to the event [Courier, Lettres Fr. et It.,1825, p. 862].
- **Everything that happens, any fact that is a part of the time**. In such weather, I did not had any visit, and if an event, however small it may be, does not appear in my flat there, could cause with little distractions [Flaub., Corresp.,1879, p. 182]
    - The Photographic Art, in the eyes of most people, is primarily based on capturing the fleeting event, accident maybe, or unpredictable disaster, to displace the minute or the second so that it could create an apparition of the fantastic moments of our daily habits. [Lhote, Peint. d'abord,1942, p. 49].
- **Fact that made a significant importance for a community.** Think and act in the light of an event; measure the impact of an event, consider and mark an event.
    - Yes, my parents moved. This is not an extraordinary event. This is "an event" indeed, and to my mother, it is a big concern [Duhamel, Nuit St-Jean,1935, p. 57].
- **Facts on what we give an excessive importance**.
- **Fact that attracts attention by its uniqueness and exceptionalness.** Make event, an event of the day, the century, of my life. Make historical events, literary, politically, scientifically and any landmark event (i.e. with date). The major crimes such a great virtues surprises us.
- **All the facts that make more or less important news.** The events we are witnessing, be aware of the events, being rushed, and overwhelmed.
    - If Metz had surrendered a day later, if the second army had arrived a day later to the front of the Orleans' forest, there will be no would to renounce the siege of Paris (...). The turns that events have taken affected my chances rather than my merit [Barrès, Cahiers,t. 9, 1911-12, p. 5].

"***The event concept is simple yet powerful***" this is how Opher Etzion sees events in his book "Event Processing in Action". A coffee shop example that M. Etzion has mentioned in his books looks explicit enough to PREDYKOT. A man is setting in a coffee shop with his laptop trying to finish a story. Couple of things happen around him, i.e. people moving, waitresses bring orders, etc. Exciting events are not those, but the ones that leave a significant mark. The story gets more exciting when a robber breaking in and starts asking people money. The peace of this atmosphere of the shop has been affected by this phenomenon. The man interrupted his writing while there are no more thoughts coming out. During this event, a client surrendered his wallet to the robber, waited the robber to disappear and called his credit card company to cancel the card usage. So what is happening is that each event introduces the triggering of other events or maybe activities.

The coffee shops works in two ways: 1) A synchronous fashion (i.e. example of traditional information systems as Request/Response): a client reaches the desk to order a coffee and a brownie, the waitress serves his order by heating the brownie, preparing the coffee, takes the brownie out of the microwave, gets paid. She is then turns to serve the next customer. 2) Asynchronously take orders then deliver clients while working in parallel (i.e. example of Event-Based Systems). This way needs to team one that takes orders and money and others deal with the order by the same previous way. When order is ready waitress calls clients, or brings them orders directly to the table. The last approach allows clients to relax at the table, maybe get out a laptop and start working while waiting. The following figure taken from the same book by M. Etzion illustrates these two approaches. Whereas both approaches see the robbery as an unexpected event (exceptional) that results in reactions from those involved.



Coffee shop examples: the synchronous approach (upper illustration) versus the event-driven approach (lower illustration)

It is important to notice that unexpected or exceptional events are not always negative (e.g. winning best project awards for PRDYKOT or finding a significant amount of petrol in France).

Events have sources! If we know the source of events or it is included in our daily activity, then we go directly to the source or the stream (e.g. like reading the news paper). Otherwise, we need to do an effort to get those events because they enter in our interests (e.g. Call For Paper). Therefore, we might be obligated to subscribe, for example, to those events. Once we discovered the source, we start detecting events, but real events that actually happening. In fact, in reality what we can only observe are indications, warnings or signs (e.g. a man noticed that his family's started to drink juice more, so he needed to add additional carton of juice the grocery list. So, what happened is that the man observed the

consumption of his family after the first time the family was out of juice. After the third time that this phenomenon occurred, he reached his conclusion). The example shows that there are two types of events: low-level events or indications and others that are an outcome of observing these events. We call the latter a Complex Event. The conclusion the man reached is a meaningful and a semantic description that we call Situation (i.e. “Out of Juice”). The reaction on this situation is when he decided to add item (i.e. a new configuration) to his weekly list.

Events may happen across the various layers of an organization as sales leads, orders or customer service calls. Or, they may be news items, text messages, social media posts, stock market feeds, traffic reports, weather reports, or other kinds of data. An event may also be defined as a "change of state," when a measurement exceeds a predefined threshold of time, temperature, or other value.

Event Processing is a methodology that processes (tracks and analyses) collections of information (data) about things that happen (events), and deriving a conclusion from their meaning. Collections might be structured and organized (datacentres), grouped (streams) or vast (clouds). So logically, an Event Processing System should have monitoring system, reasoning engine (semantics analyser), decision support system, enforcement system (to react and apply actions) and report system (logs, historization and audit).

It is important to state that events could be located in two forms: Time-Based and Storage-Based. Storage-Based is when the events are partially ordered (poset) or disordered inside a pool, i.e. aka Event Cloud. On the other hand, events could be fully ordered by Time. This creates a sequence of events known as an Event Stream. A special case of the Event Stream is when one applies Windows of Time on the stream of events (i.e. normally this window is moveable with time and it is called “Sliding-Window”).

It was in 18th of August 1998, that David C. Luckham and Brian Frasca first introduced the concept of Complex Event Processing in Distributed Systems at Stanford University. He defined this technology as:

> “*Complex event processing is a new technology for extracting information from distributed message-based systems. This technology allows users of a system to specify the information that is of interest to them. It can be low level network processing data or high level enterprise management intelligence, depending upon the role and viewpoint of individual users. And it can be changed from moment to moment while the target system is in operation. This paper presents an overview of Complex Event Processing applied to a particular example of a distributed message-based system, a fabrication process management system. The concepts of causal event histories, event patterns, event filtering, and event aggregation are introduced and their application to the process management system is illustrated by simple examples. This paper gives the reader an overview of Complex Event Processing concepts and illustrates how they can be applied using the RAPIDE toolset to one specific kind of system*”.

However, it is only after 2000 when CEP began to popup in IT market. In fact the number of CEP applications and products in the marketplace has grown rapidly in 2006.

Tim Bass from TIBCO Software Inc., in April 23, 2007 has defined Complex event processing (CEP) as *an emerging network technology that creates actionable, situational knowledge from distributed message-based systems, databases and applications in real time or near real time. CEP can provide an organization with the capability to define, manage and predict events, situations, exceptional conditions, opportunities and threats in complex, heterogeneous networks. Many have said that advancements in CEP will help advance the state-of-the-art in end-to-end visibility for operational situational awareness in many business scenarios. These scenarios range from network management to business optimization, resulting in enhanced situational knowledge, increased business agility, and the ability to more accurately (and rapidly) sense, detect and respond to business events and situations*.

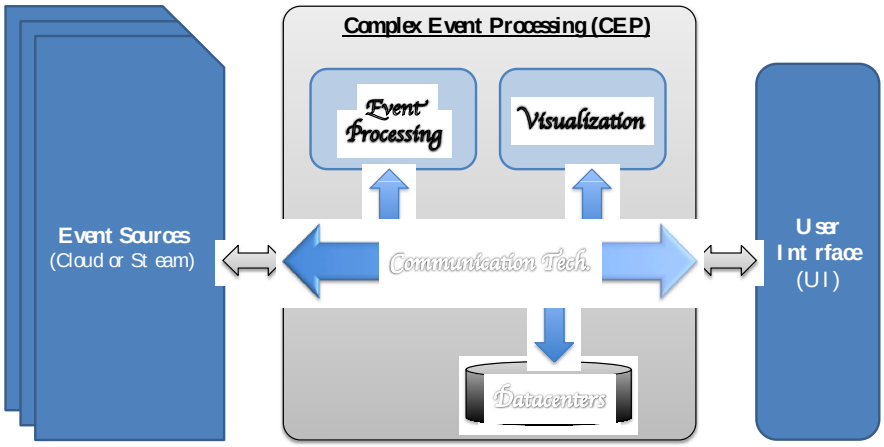
For PREDYKOT, Complex Event Processing (CEP) is a complete loop of processing on events. That is starts with combining information (data) from multiple sources (i.e. heterogeneous sources that could contain raw or pre-processed data) to infer events or patterns (i.e. complex events) that suggest more complicated circumstances, i.e. situations that have meaning (such as opportunities or threats) or maybe another complex events. The goal of complex event processing is to identify meaningful events and respond to them as quickly as possible. So logically, a CEP-Based System should have monitoring system, operational system (processing complex events i.e. filtering and aggregation), situation manager (i.e. could be reasoning engine or semantics analyser) (Adi & Etzion, 2003), decision support system, enforcement system (to react and apply actions) and reporting system (reports, logs, historization and audit). At last, a representation language to express CEP rules that govern or control all these systems.

Tim Bass wrote in “The CEP Blog” that is possible for a person to understand CEP by examine the way we as human, and in particular our minds, interoperate within world around us. Here is an example representing the analogues between the human mind and CEP in a table of “Human Cognitive Functions and CEP Functionality”:

Human Body	Complex Event Processing	Functionality
Senses	Transactions, log files, edge processing, edge detection algorithms, sensors	Direct interaction with environment, provides information about environment
Nervous System	Enterprise service bus (ESB), information bus, digital nervous system	Transmits information between sensors and processors
Brain	Rules engines, neural networks, Bayesian networks, analytics, data and semantic rules	Processes sensory information, “makes sense” of environment, formulates situational context, relates current situation to historical information and past experiences, formulates responses

and actions

Solving complex processing requires a functional architecture to help organizations understand and organize the system requirements based on analysing the behaviour. The following architecture is a common architecture in which usually each CEP engine respects:



Visualization at all levels of the processing mechanism is extremely important especially in complex scenarios like PREDYKOT. A human body mind posses an impact of visualization that helps him to understand and solve many complex problems. The internet as a complex environment full of heart beating, visualizing the internet traffic is one of the most important technologies that an enterprise can provide, e.g. HP OpenView that could give visualization of the internet traffic at the major backbone routers.

Complex Event Processing requires visualization as well and at every level of the event-processing model. Possibly, visualizations in a CEP could take these forms:

Visualization Example	Description
Event Pre-processing	Map raw sensor input data to a special message format;
Event Refinement	Tracking and Graphing an event object.
Situational Refinement	Providing a visual list in a network management centre of the top X number detected of threat-related situations. Maybe as well giving an estimated values and conditional probability.
Impact Assessment	Providing a visual list of the top X number detected of (e.g. equity trading opportunities) with estimated profits, along with a risk KPI.
Process Refinement	Providing a visual graphic of alternative solutions (e.g. alternative routes for commercial aircraft during a snowstorm)
User Interfaces (UIs)	Tools to model and design event processing scenarios, rules and other analytics.

CEP relies on a number of techniques, including:

- Event-Pattern Detection
- Event Abstraction
- Modelling Event Hierarchies
- Detecting relationships between events (i.e. such as causality, membership or timing).
- Abstracting Event-Driven Processes
- Commercial Applications of CEP include security monitoring, algorithmic stock trading, the detection of credit-card fraud, and business activity monitoring.
- Event correlation is a technique for making sense of a large number of events and pinpointing the few events that are really important in that mass of information.

One of the main interfaces of PREDYKOT is the SIEM (security, information and event management) product. Red Lambda has introduced it to the market recently with a feature of having a CEP (complex event processing) engine. The engine is deployed at the front-end because, in the company's own words, "log and security data is a big data problem". One of SIEM vendors who use CEP is Tier-3, though that SAS has been doing work in this area (for example, for real-time identification of 'low and slow' attacks, which is notoriously difficult) [Philip Howard, 2011 Bloor]. However, not all SIEM products supports CEP as the front-end interface, OSSIM (The Open Source SIEM) that provides a SIEM solution, and a framework that allows tight control over widely distributed enterprise networks from a single location. Nevertheless, a new research project presented in 2012 has introduced a cooperation work between OSSIEM and CEP. MASSIF: A Highly Scalable SIEM, Ricardo Jimenez-Peris explained the scalability of OSSIM SIEM by translating directives directly into CEP queries.

Event Processing Applications are usually associated to following functionalities or categories: 1) Observation: CEP works as a monitor system or process looking for exceptional behaviours and generating alerts when such behaviour detected. The reaction in category is private to the consumer of the alerts. 2) Information dissemination: CEP delivers the right information to the right consumer at the right granularity at the right time. 3) Dynamic Operational Behaviour: CEP drives the actions performed by a system dynamically so as to react to incoming events. 4) Active diagnostics: CEP diagnoses a problem, based on observed symptoms. 5) Predictive processing CEP identifies events before they have happened, so that they can be eliminated or their effects mitigated.

2.1.2 INTEGRATION WITH BUSINESS PROCESS MANAGEMENT

CEP goes in solving complex problems: Rule-Based Access Control, Risk Management, Fraud Detection, Intrusion Detection, Exception Management and others. Business Process Management (BPM) focuses on end-to-end business processes, but only, in order to continuously optimize and align for its operational environment. One use of CEP is to link separated processes.

"*In the aerospace industry, monitoring process on breakdowns of vehicles to look for trends is a good practice (i.e. determine potential weaknesses in manufacturing processes, material, etc.). Another parallel process monitors current operational vehicles' life cycles and decommissions them when appropriate. Now CEP could link these*

separated processes when the initial process (breakdown monitoring) discovers a malfunction based on metal fatigue (a significant event), CEP creates an action to exploit the second process (life cycle) in order to issue a recall on vehicles using the same batch of metal discovered as faulty in the initial process." --***Wikipedia***

CEP and BPM must be integrated at two levels: 1) Business Awareness Level (End Users must be aware about the added value of their individual processes), and 2) at Technological Level (the definition of a methodology BPM implementation).

2.1.2.1 Off-The-Shelf Tools

- ESPER: Better in processing Complex Events
- DROOLS: Better in Processing Actions based on Rules, Business and Complex Events (i.e. Knowledge Base).
- Other Tools: Storm, etc.

2.2 SITUATIONS

Abstract and high-level terminologies stand like a mirror, with their semantics, between the middleware technologies and business plans. They are reflected, or maybe described, in a meaningful manner the business objectives and goals within the stream of ecosystem's behaviour (i.e. this is true when understanding, then assigning, meanings/semantics to events and complex events).

2.2.1 DEFINITION

Situations as defined by the L'internaute dictionary:

"Ensemble des conditions, circonstances dans lesquelles une personne se trouve"

Collins Dictionary also has the same definition for situations as:

"A set of conditions and circumstances in which one finds oneself"

The clarification of the situation's notion is very important because of the image it reflects to the entity interested in it (i.e. Security and Management). Therefore, it is necessary to discuss and clarify what the notion of situation in certain definition could be useful to PREDYKOT. First, we discuss the different faces of situation that could be interesting to use in the project. Here is a study extracted and made based on information from the Centre of National Resource Textual and Lexical of France (CNRTL). Situation is a set of conditions and circumstances in which one finds oneself:

- **At a given time**, situations could be described as awful, advantageous, critical, cruel, terrible, exceptional, favourable, bad, painful, emergent, fake in which they can examined, flipped, reversed the situation back into the same initial situation. One must consider goals for the production of his business, e.g. the cost, to state and expose the moral and physical situations of Workers [Gobineau, Pleiades, 1874, p 166].
- **Being in an interesting situation**: A woman being pregnant, her wavy black silk dress that follows the fashion of a certain time, in which this allows her to reflect this

phenomenon, was not in what the English call an interesting situation [Nerval, Fayolle, 1855, p. 15]. As for her good friends, they are all in an interesting situation [France, Life literature 1891, p. 111].

- **At a given point of view**, administrative situation, financial situation, legal situation, physical situation, military situation, financial situation, property status, family status, irregular situation, marital status. A person inquired straight away on our business situation, and though he was very young, and limited in his means, he paid everything! The look that marked the greatest affection on my sister and me [Restif La Bret., Nicolas, 1796, p. 113].
- **Being the Man of the Situation**, being the person who arrives at the right time to solve the problems usually present. If you have objected to me that you do not know what it means to be the man of the situation, and that, without any reason, it is no longer even worth a comparison, I will push up the condescension excessively and assuring you that Storch does not meet the aspirations and needs of the time [Gobineau, Pleiades, 1874, p. 106].
- **Being at the height of the situation**, to be qualified to overcome the current difficulties. [Dumas, Demois. St. Cyr, 1843, i, 5, p. 106].

Saint Thomas d'Aquin (Italian Theologian) has described situations in a quote: "*Every step, every situation reflects your state of mind and, by the same token, carry a spiritual meaning*". If we only make a little words match between Your and Ecosystem, between Mind & Business Goals and between Spiritual & Behaviour, a new brief definition could make PREDYKOT happy: "*Every step, every situation reflects ecosystem's state against its business goals and, by the same token, carry a meaning to its behaviour*".

As an outcome of this study, we can try to define in details a situation in the eyes of PREDYKOT to be:
"*A set of events, conditions, rules, policies, actions, alarms and context participating all together to deliver an evaluation messages, i.e. point of views about the behaviour that are in any form of meaningfully or semantically described, to the business dashboard, the reasoning engine or any entity in PREDYKOT that have the power of delivering a decision, i.e. the right entity in the right time to react on situation. Based on these messages, decision-makers of PREDYKOT are qualified to find or figure out which situation the ecosystem is in*".

SITUATION A situation is an event occurrence that might require a reaction.

One of the main themes in event processing is the detection and reporting of situations so that they can be reacted to. The reaction might be as simple as picking up the phone or changing the weekly shopping list, or it might be more complicated. If we miss a flight connection there may be several alternative reactions depending on the time of the day, the airport where we are stranded, the airline policies, and the number of other passengers in the same situation. Several people including Asaf Adi and Opher Etzion have used the term situation in this way: "AMIT - The Situation Manager" VLDB J. (VLDB) 13(2): 177-203 (2004).

Until the moment, Situations are has been introduced only from human beings perspectives. Let's now move from the world of people to the world of information systems.

2.2.2 SITUATIONS IN PERVASIVE SYSTEMS

In Pervasive Systems: A situation is a set of contexts in the application over a period of time that affects future system behaviour. A context is any instantaneous, detectable, and relevant property of the environment, system, or users, such as location, available bandwidth and a user's schedule.

2.3 CONTEXT AWARENESS IN EVENTS AND SITUATIONS PROCESSING

Dey and Abowd defined the "Context" as following: "*Context is any information that can be used to characterize the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves*". Context is typically the location, identity and state of people, groups and computational and physical objects.

Context is a University, a School or an Institute where one applies and gets admission. Once an admission obtained, people will see him/her in a different way, a different look, where (s) he becomes a bachelor student or a researcher. It is a new identity, location, position, status and many other meaning related to the environment.

Non-Contextual information is information with no sense (e.g. Light is turned off) or maybe useless (e.g. men in black suits)

Therefore an event could be fired or triggered by contextual and non-contextual information. However, to express this event in order to be used inside a complex event and produce or define a situation, we need the event to get admitted by a context.

Originally, following definition has been placed in 1375–1425; late Middle English < Latin contextus a joining together, scheme, structure, equivalent to contexere to join by weaving, Dictionary.com:

"The set of circumstances or facts that surround a particular event, situation, etc."

Wikitionary defines the term "context" as:

"The surroundings, circumstances, environment, background or settings that determine, specify, or clarify the meaning of an event, situations or other occurrence."

Wikitionary defines the term "contextual" as comparative or superlative:

"Of, pertaining to, or depending on the context of information; relating to the situation or location in which the information was found."

Every event, situation or fact that is not surrounded with this definition is a not/non contextual one (e.g. a non contextual event could be "a person heard a noise"). Sometimes, people inside the computer science domain refer to this kind of events as "Raw Event". On the other hand, F. MELGANI used another definition of non-contextual events in his paper "Fusion of multitemporal contextual information by neural networks for multisensor image classification". He supposes that the inability of producing a schema

that could classify or contains images produces a non-contextual classifications of. This is usually caused by single-time images; which are images that appear once in a sequence. **This could mean that an event that we cannot classify it and that appears only once is considered to be non-contextual.**

An interesting article is talking about "The Dangers of Non-Contextual Pattern Matching" contributed by Rafal Los at INFOSEC Island, 15 Feb 2012. A story published by "*The Chronicle Herald - News, Canada"* mentioned in his article invited him to produce this analysis on non-contextuelity: "*Information Security or homeland security is simple, simple pattern matching (non-contextual) is dangerous, and produces more issues than it solves. Technology isn't just an answer it's the answer here, but that's only part of the solution. Context is critical, and an analysis system built on solid technology, implemented effectively and maintained fiercely is the only way you'll even have the slightest chance to defend your organization against the daily threats.*" So, context (complex or non-simple) pattern matching, i.e. complex event processing and SIEM, is a very essential issue in order to defend against threats.

The way we view things in our daily life is affected by their context, as M. O. Etzion described. Context could be related to three main elements: Time, Location and Conditions (circumstances):

- *At night one can open his car with the remote control from quite a long distance, but he may have to come quite close on a sunny day.*
- *One might feel safe enough to carry money in his wallet in his own city, whereas in countries that have a reputation for muggings, he hides his money.*
- External conditions could be like the state of the traffic. *The route one chooses to drive to the airport might depend on his knowledge of likely traffic conditions, or on congestion reports that he has picked up from the radio.*

Like in real life, Context makes the same effects in event processing. *A particular event can be processed differently depending on the context in which it occurs, and it may be ignored entirely in some contexts.* Some of the Cupcake Delivery agents, e.g the Assignment Manager agent deals with a driver's Delivery Order event differently depending on that event's location relative to an earlier Order Request event.

In Context-Aware computing, we take a cloud of event instances and classify them into one or more sets, i.e. context partitions. An event processing operation, application or agent is associated with a context that links it to each of these context partitions.

Event processing applications or agents may use context in three ways: 1) **Temporal context:** *A stream can comprise an open-ended set of event objects*. The processing operations could not be executed until having the stream filled with events. Then, the stream will be divided to possible divisions that contain set of events: sequence of context partitions or windows. It's the rules that determine which event instances are admitted into which window. 2) **Spatial contexts** allow agents to aggregate location-related events into separate context partitions or windows. 3) **State-oriented context** an agent could active in some contexts and inactive in others. For sure, there are other ways in order to use contexts.

In 2003, Kellerer, W., and Tarlano, A. have presented in the paper “Context Aware Wireless Ubiquitous Computing” the following scenario about Context-Aware Computing:

> “*Cheryl has invited her boyfriend and his parents for dinner tonight. The dinner will take place at her house at 8pm and she is currently sitting in an important meeting with her manager. Fortunately, Cheryl has her Reminder Buddy, RB, running on her mobile phone, to take care of notifying her, if she forgets something. Knowing calendar of Cheryl, RB assumes she has forgotten the dinner, therefore RB decides to notify Cheryl about the dinner menu preparation. However, RB realizes, after verifying with the manager's personal assistant, that Cheryl is currently having a meeting, and the notification had better be given after the meeting is over. The meeting is over, and Cheryl walks to her office. RB informs Cheryl about the dinner tonight. When she enters car at the parking lot, she asks RB for menu preparation. RB communicates with Cheryl’ s electronic Household Buddy, HB, which registers any discovered item via RFIDs that come with each product. HB indicates that there are not enough ingredients in her refrigerator, thus he suggests a menu and a related shopping list to RB. The shopping list is not only based on the contents of the refrigerator, but it also takes into account the Cheryl’ s and the parent’s favourite recipes. Based on responses from HB, RB checks the availability and price of products in the supermarkets along the route to Cheryl’s home. RB selects one or two markets, which offer all the products in order to avoid several stops, and displays the info to Cheryl. Related recipes are automatically transferred to HB and HB will display them on LCD in the kitchen once Cheryl enters the house.*”

3 BEHAVIOURS TACKLED BY THE POLICY

Observing failures and other – desired or undesired – behaviour patterns in large-scale software systems of specific domains (telecommunication systems, information systems, online web applications, etc.) is difficult. Very often, it is only possible by examining the runtime behaviour of these systems through operational logs or traces. However, these systems can generate data in order of gigabytes every day, which makes a challenge to process in the course of predicting upcoming critical problems or identifying relevant behaviour patterns (situations). The one can say that there is a gap between the amounts of information a system has and the amount of information a system needs to make-a-decision (Fülöp et al., 2010).

In practice, CEP is obvious in the healthcare industry where it continually tracks healthcare workers’ behaviour for hygiene compliance (e.g. sanitizing hands and wearing masks), reminding them to perform hygiene when appropriate to prevent the spread of infectious disease. Each worker wears an RFID badge that displays a green (safe), yellow (warning) or red (violation) light, depending on what behaviour the RFID chip has observed.(Etzion & NIBLETT, 2011)

3.1 INTRUSION DETECTION

Intrusion Detection: A Brief History and Overview

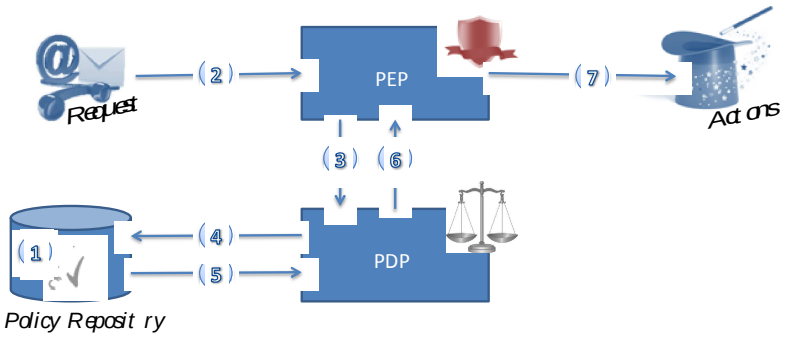
3.1.1 EXAMPLES ON CEP

Complex event processing is a technology, which allows correlating basic events to complex events closely aligned with the semantics of business processes related to the events. (Hamerling, n.d.)

Using events in information systems is not new. Events considered as exceptions in order to interrupt the regular flow of execution and enforce another process to happen (e.g. a small computational program divided by zero, which raises an exception event in order to enable the programmer to end the program with an error message then maybe to add correct value that will let the program continue the computation). Events also exist in graphical user interface (GUI) systems where components, i.e. buttons or menus, are designed to react to events like mouse or keyboard events. Nowadays, following examples are different ways of automated event processing: A patient is connected to multiple sensors that monitor and calculate medical readings. The readings are events that will be analysed by an event processing system (CEP). A physician can configure this system, for patient-by-patient. Once a certain combinations of readings are detected within a predefined time period, a nurse will be notified. Then the physician will be informed in case other combinations occurred. This example demonstrates the use of event processing to allow timely response to emergency situations, or as part of a personalized diagnosis program. (Etzion & NIBLETT, 2011)

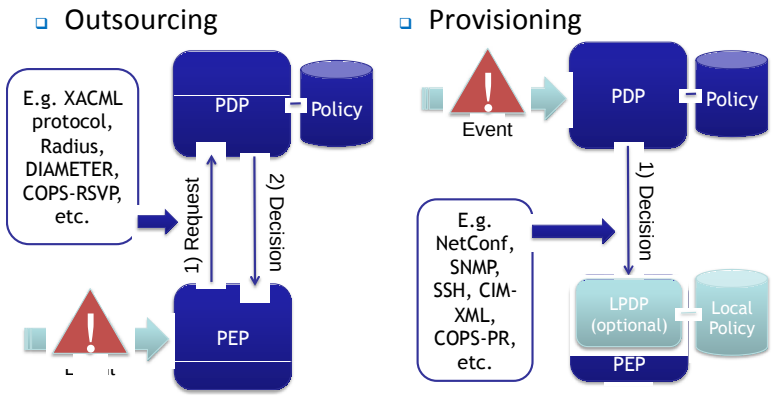
3.1.2 TECHNICAL ARCHITECTURE & APPROACH

Besides the policy languages, architectures and their implementation approach are important criteria for choosing an access control model. The absence or poor functioning (or any difficulty of use, configuration, parameterization) can be a real obstacle of the language usage as an implementation. Some policy languages do not have architecture and others are stored in a repository as part of the architecture. However, the more the applied architecture is structured, modulated, the more the policy management will be easier.



Most languages use a particular framework perfectly adapted to them and generally inspired by the policy-based management architecture. The Network Working Group of the Internet Engineering Task Force (IETF) proposed this widespread, standard and well-known architecture [Yavatkar00]. Several Elements are currently observed: Policy Repository (PR) that stores policies, Policy Decision Point (PDP) that allows to evaluate the relevant policies (set of rules) and shares a decision based on them, Policy Enforcement Point (PEP) is responsible of enforcing the decisions taken by the PDP. In order, to ease the process of writing the policy on the manager, tools such as Policy Administration Point (PAP) was proposed by XACML architecture. PAP gives an interface to the policy writer to write his policy in a natural language.

The Architecture was proposed in an outsourcing mode (i.e. synchronized, blocking or request/response mode), where the flow of processes takes the following manner: 1) the policies are written (e.g. by the PAP), and then stored in the Policy Repository for a later access and assess by the PDP. 2) When receiving a requesting, 3) the PEP sends the request to the PDP 4) who verifies or match in the Policy Repository to which policy it should refer in its decision. 5) Once the PDP becomes aware about the decision, 6) the PDP transmits it to the PEP 7) by its role, the PEP responsible for enforcing the corresponding actions.



This Outsourcing approach defined and standardized by Network Working Group, Common Open Policy Service (COPS) [RC3084] is the only and default mode of the IETF architecture. "*The Outsourcing model addresses the kind of events at the PEP that require an instantaneous policy decision (authorization). In the outsourcing scenario, the PEP delegates responsibility to an external policy server (PDP) to make decisions on its behalf. For example, in COPS Usage for RSVP [COPS-RSVP] when a RSVP reservation message arrives, the PEP must decide whether to admit or reject the request. It can outsource this decision by sending a specific query to its PDP, waiting for its decision before admitting the outstanding reservation.*"

On the other hand, the IETF Architecture at the PDP level only supports the "Provisioning" approach (i.e. using the COPS protocol for support of policy provisioning (COPS-PR)). An extraction from the [RC3084] could best describe the adaption of IETF architecture to the Provisioning mode: "*The COPS Configuration model (herein described as the Provisioning model), on the other hand, makes no assumptions of such direct 1:1 correlation between PEP events and PDP decisions. The PDP may proactively provision the PEP reacting to*

external events (such as user input), PEP events, and any combination thereof (N:M correlation). Provisioning may be performed in bulk (e.g., entire router QoS configuration) or in portions (e.g., updating a DiffServ marking filter)."

4 STUDY ON RELEVANT PROJECTS

This section aims to monitor related projects to PREDYKOT and the concurrent achievements presented by these projects.

4.1 INDUSTRIAL PRODUCTS – SUPERVISION POLICY

PREDYKOT uses a sort of administration or supervision policy that refines the deployed authorization policy. Part of this refinement process is based on the observation and filtering of network events and activities. The main objective is to adapt the policy at the end to multi-objectives of business enterprises (i.e. use cases presented by industrial partners).

Thanks to a research presented in the Université Paul Sabatier, this section is presenting mainly two tools similar to PREDYKOT in details. Nevertheless, the following tools are mostly the important ones in terms of Audit and Supervision:

- TUFIN, the name of the company founded in 2005 by a former member of "Check Point".
- FIREMON, the name of the American company founded in 2004 by Gary Fish.
- Secure Works the name of the company that was founded in 1998. Since 2011, it becomes a part of Dell Inc. This company presents facilities to manage security through: Network Intrusion Detection and Prevention (IDS/IPS), Security Information Management (SIM), Firewall Management (i.e. Protection of data using Certified Security Analyser that correlate events (e.g. user activities) across users environments and then analyse it to asses and detect threats. Then human experts respond or react on this threat to protect the concerned organisations) and there are many more functionalities related to logs and security monitoring.
- Skybox Security, the name of the American company, which was established in 2002. The main objective of this company is providing Proactive Security Risk Management. The company has three main activities: Firewall Assurance (i.e. Real-Time deliverance of PCI (Payment Card Industry from the Security Standards Council) compliance audits, automated rules and configuration checks and optimizing the change workflow process as "Skybox Change Manager"), Network Assurance (i.e. visibility and monitoring of outages and network topology) and Risk Control (i.e. simulation for attacks to identify and avoid potential cyber threats then inform the workflow as "Skybox Threat Manager" and define IT risks).

4.1.1 TUFIN

The TUFIN Security suite (TSS) solution is a solution of 3 products:

- **Secure-Track** is the core of all TUFIN's products. It's responsibilities to know rules and objects of Firewalls, as well as all routing tables of firewalls and routers. This module analyse the configuration of Firewall (i.e. the module ensure the collection of both: configuration information and real-time events, in order to ensure tracking changes. Then, it provides visibility, improve deployed policy and ensure the policy conformity. The data collection is made through real-time regular polling).
- **Secure-Change** is a powerful workflow to manage changes and simulate the rule in the network. This tool managing changes in the software suite. Secure-Change is primarily a tool for Secure Exchange Workflow (WMS). It allows the setup of a structured cycle for Change Management. Based on the internal organization, it adapts the process of modifications/changes of the security policy. Each workflow will be reserved only for involved or concerned persons.
- **Secure-App** that allows simple application through your network publication. This tool is an approach for the policy management of network security. Users can easily define, update, monitor and remove applications without analysing long list of rules (ACLs) on multiple Firewalls. A user, whose job has nothing in common with the network administrator, can communicate his needs about the firewall and have a vision on the functionality of his applications.

These modules are to be installed on a Linux operating system. TUFIN provides his own OS suitable for its tools called "TUFIN-OS." This OS is originally based on Red-Hat Linux with some modifications.

TUFIN offers its solution for an appliance or a server. Here are the main points that could interest PREDYKOT:

- **Conformity**: TUFIN retrieves firewalls' configuration automatically and regularly for modification detection. It keeps all the configurations of each version in order to be able to compare different versions together. However, only two versions could be compared at the same time. It is able also to compare two configurations from two different firewalls. TUFIN has the following characteristics as well:
  - Changes Management: Change Control Panel, publishing a report in real-time to prevent changes that are unplanned or executed by mistake.
  - Optimization of safety: search for not used rules to remove them, thus reducing security vulnerabilities.
  - Disabled, redundant or hidden rules can potentially be removed from the filtering policy. "*A rule is defined as **Hidden** when another rule above it is applied to the query. Thus, a rule that turns out to be always hidden by another will never be used.*"
  - When a rule detected to be too permissive, an Automatic Policy Generator (APG) will propose a division of this rule. Then, the user will choose the granularity he wants and he will choose as well which flow to make less permissive.
  - To improve the performance of the filtering policy, TUFIN allows, via an audit, to verify the priority of rules inside the following checklist: "Rule Order

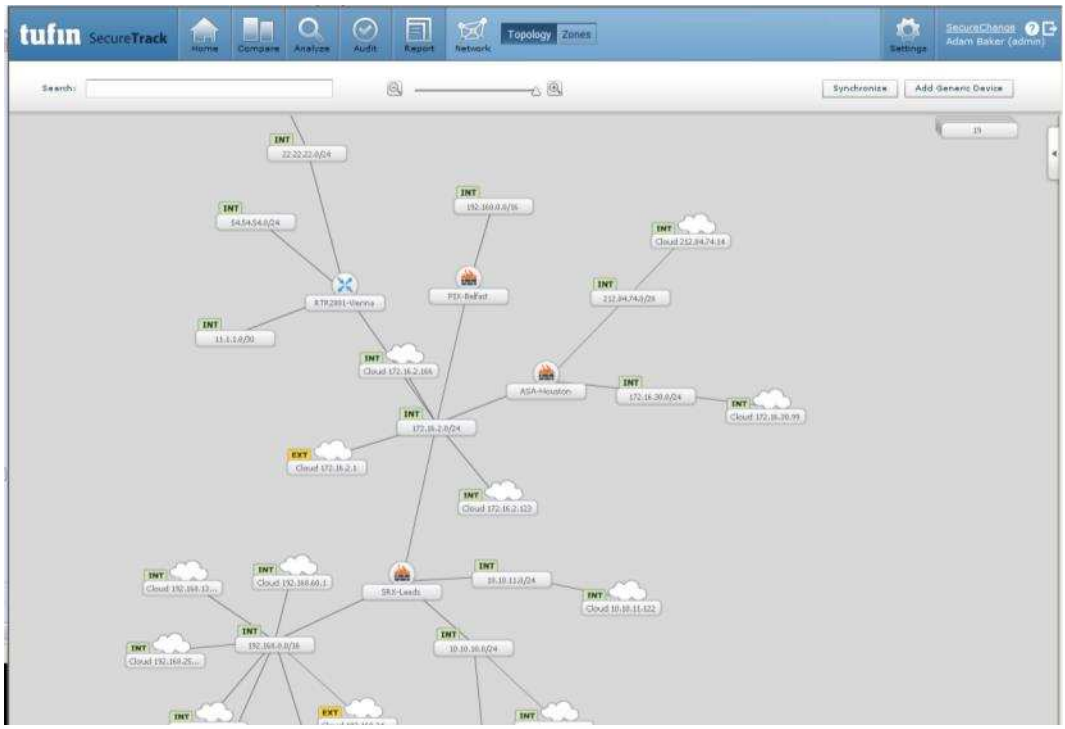
Optimization". The latter will allow users to discover which rules are misplaced in the flow matrix.

- TUFIN provides in its reports a list of X% of rules that are used; with X is the desired value. By default, the value is set to 50% of the used rules and it sorts them by ascending order. Each listed rule is assigned a utilization rate in %.
- TUFIN does not allow any manual changes on the configuration. It is only happy to show and illustrate the configuration files and logs.

- TUFIN can generate a report on the use rate of rules, of the policy.

- TUFIN is functional, but sometimes uses resource intensively. Although it respects manufacturer recommendations, but the interface undergoes slowly. Sometimes, the end-user is obligated to respect the order and wait for the current results of a requested report before applying for a second one. In case of having the machine overloaded with tasks, TUFIN is quickly outdated and it is necessary to wait several minutes or maybe an hour to regain control.
- **Decision-Support**: TUFIN highlights to the user some suggestions and warnings about the tasks it is currently trying to achieve. It does not hesitate to bring up messages and suggestions to users while he is at the dashboard level.
- **Audit results** are presented in a tabular form. However, outside the dashboard there are no charts and graphs that could give a quick overview on the results. It lacks a development of important elements when results are displayed. Moreover, it is not possible to export in all cases the information (often in large numbers) in a handy format to rework on them later (e.g. CSV, XML).
- **Risk Assessment:** Upon the injection of the firewall among TUFIN elements, TUFIN checks the filtering policy and assigns a criticality index. When building the index, TUFIN compares each filtering rule with a base of "Risk". Therefore, TUFIN include within the configuration all the potential risks that should be monitored and which should be remedied. Risks assessment is based on the type of service used and the formation of the rule (source / destination). Depending on the context or business problems, it is possible to modify the severity of the risks. Nevertheless, it is not possible to remove or add new one. In addition to this initial audit, TUFIN plans to create fully customizable analyses based on need. Thus it will be possible to check the opening or blocking flow. The completion of this analysis requires the definition of "Queries" that will define a particular test. This test will be attached to an equipment, then TUFIN generates a report based on the test's outcome.
- TUFIN match all rules to a given query (i.e. the verification of authorization service). This functionality is useful in case the firewall administrator wants to troubleshoot and find the cause of a problem. So, TUFIN prepares a report and defines a list with the corresponding rules that forms the query.

- **Ticket Management**: This module, named "Secure Change", is fully customizable according to the Operational Mode of the company. It allows customizing steps of ticket's opening and chooses persons authorized to valid and manage tickets. Secure Change also has a Priority Time Management and Service Level Agreement (SLA). When a ticket is opened, the user must manage this ticket. So, the "Secure Change" helps the user (ticket manager) by assessing the criticality of the application and providing decision-support.
- TUFIN can display a map showing the network topology. Different firewalls do not always have the exact object name of the same equipment. This is why the user needs to manually setup, on the application, all the objects' names that will be subsequently displayed on the map. A search function that can retrieve objects from different firewalls' configurations. TUFIN simply displays the network topology, but it could display more in-depth by providing more information and interaction.



4.1.2 FIREMON

The FIREMON solution also consists of three products:

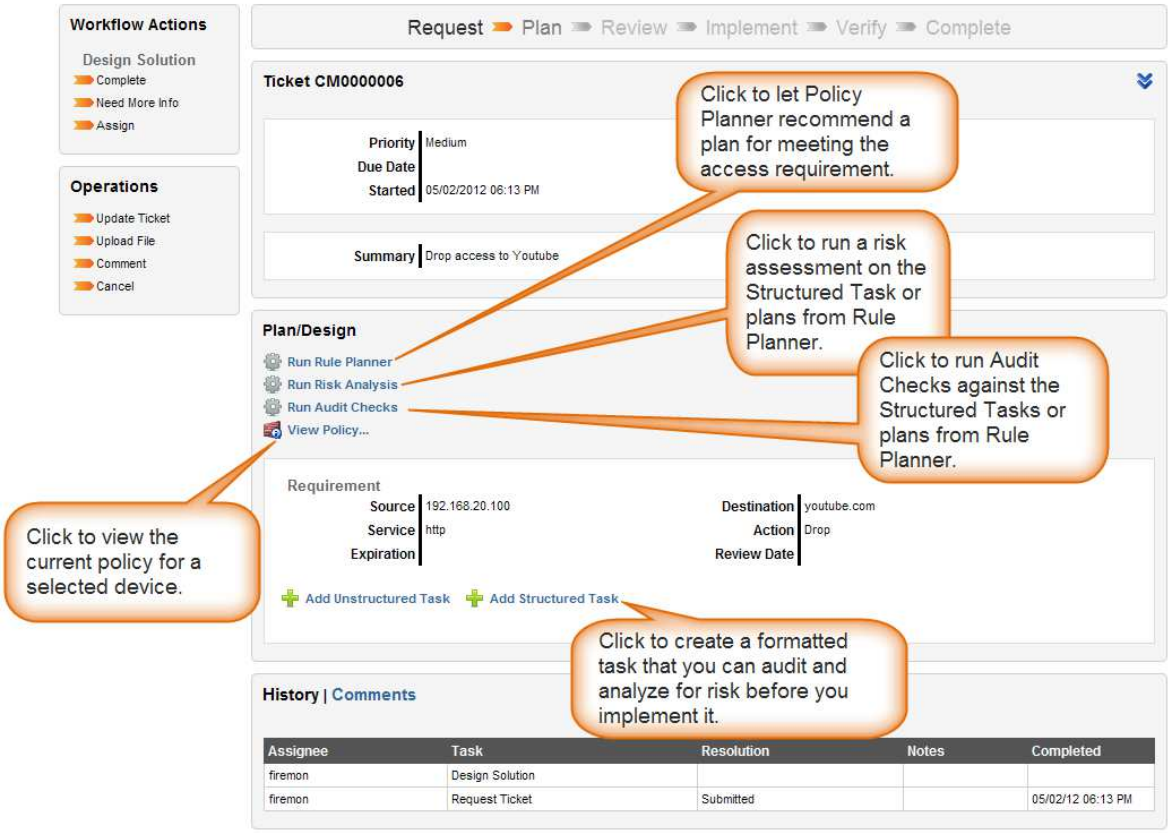
- **Security Manager**, which retrieves or collects all the information from the firewall and generates, based on this information, audits of compliance to defined requirements. The collection of information is achieved when a change is detected at each firewall. The Security Manager controls the configuration changes and provides features (i.e. such as optimization of rules through either searching for unused rules or by highlighting misplaced rules inside the filtering policy). Security Manager proposes to conduct audits of compliance defined by the administrator. So, the Manager can test its policy on certain flow wherein the Manager deems that it contains risk.

- **Risk Analysis**, which calculates the risks that cross through firewalls by making assumptions and attack scenarios. Once an attack scenario is defined, this module will assess the severity of vulnerabilities that can be achieved for each group of flow that crosses through the firewall. Risk Analyser searches for security risks by running scenarios built on vulnerability assumptions and applied on computer equipment that crosses through the firewall. It gives the possibility as well to visualize the risk and the consequences for each machine (i.e. this happens when a machine is appeared to be infected or to be with the hands of unauthorized persons). With such scenarios, the administrator will be able to track how far the attacker will reach. Thus, it gives the option to define the scope and several possibilities of infiltration into the computer network. By this, it allows the user to validate the correct bulkhead/partitioning made between networks, as well as indicate the most critical machines.
- **Policy Planner**, which provides recommendations of filtering rules, at the moment when a request for opening tickets is placed by the administration group of the firewall. This tool structures the process of validation and verification the openings of the flow that cross through the firewall. Policy Planner is concerned about processing requests of opening flows on firewalls. This module provides a decision-support for administrators, who must judge/evaluate the risk accompanied with each rule. Administrators should make a rule that meets the needs of users, but at the same time restricting at maximum the open access towards the firewall. Policy Manager will therefore be able to simulate the insertion of this rule in the filtering policy and inform the administrator about a potential risk in the network infrastructure, if any. It is a validation system of third-party that prevents or avoids even more human errors.

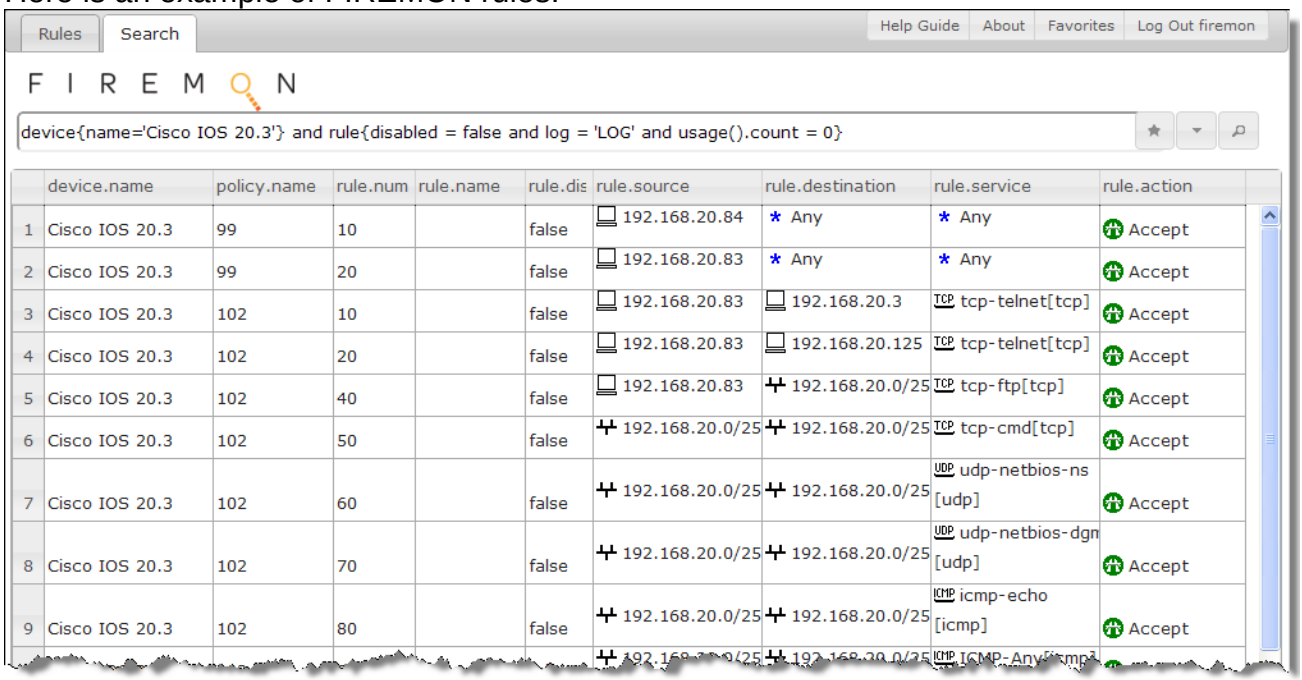
Here are the main points that could interest PREDYKOT:

- **Conformity**: FIREMON automatically retrieves device configurations and proposes possibility to export and download. So that, they can be reinstalled, manually if needed, anytime on the firewall. FIREMON has the following characteristics as well:
  - It provides a usage report on the most used rules (TOP 10) in a given time range. It uses the logs, generated by the execution of each rule, to define/measure their utilization. TUFIN uses the same method as well. The use of counter, that is exists on every rule on most of firewalls, could also be used to raise the utilization rate. However, it turns out that using logs seem more appropriate for the simple reason that the firewall counters can be reset. This scenario could occur in some clients that their Rule Management tool had to reset the usage counter for rules that has been modified.
  - The tool should provide an optimization of filtering rules implemented nowadays. It will identify what rules are useless in the current configuration of the Firewall (e.g. rules covered by a broader rule, poor prioritization of rules etc.). To do so, the tool generates two reports. The first one “Unused Rules Report” is the list of disabled rules. The second one "Hidden Rules Report" shows list of hidden rules by another ones recently assigned at the top of the filtering policy. These reports are available in HTML and XML and

PDF version. XML is useful in order to retrieve and to process the data via scripts.
    - FIREMON does not allow any manual changes on the configuration. It is only happy to show and illustrate the configuration files and logs.
- **Audit Management**: FIREMON proposes to create an "Audit". The audit will consist of "Audit Check", which is a list of points that FIREMON cover when checking the filtering policy.
- **Risk Assessment**: FIREMON analyses the risks of rules based on analysing the firewall environment. For that, it needs to retrieve data from probes that already analysed all the surrounding networks. FIREMON setup a value of criticality for each machine present on the network. Finally, it is possible to make a vulnerabilities report using “Metasploit” tool. Using all this information, plus those that already exist (i.e. routing table, filtering rules), FIREMON will be able to identify the machines that are potentially critical and thereby propose amendments on filtering rule. It has the advantage not to be limited to the analysis of the rule. This analysis provides a real added value. It has also a positive note for the reports that can be generated in the form of tables and histograms. This kind of presentation is highly appreciated by customers. Nevertheless, the large amount of preparation that this analysis needs is cumbersome to implement. FIREMON is more focused on the analysis of the environment to present it on a paper sheet!
    - FIREMON uses a map to display the network topology, especially during the risk analysis function. On this display, the user will be able to simulate an attack from a network and a machine from it. This will help the user of FIREMON to visualize the different possibilities of having attacks and to conclude that it is possible to achieve this attack bypassing through the filtering equipment.
- **Ticket Management**: The "Workflow" that helps to schematize the flow of tickets cannot be modified. FIREMON includes decision-support features such as recommendations and risk analysis reports. The manager of the ticket will be able to simulate the addition of new rule and verify the impact that this rule may have on the entire policy.

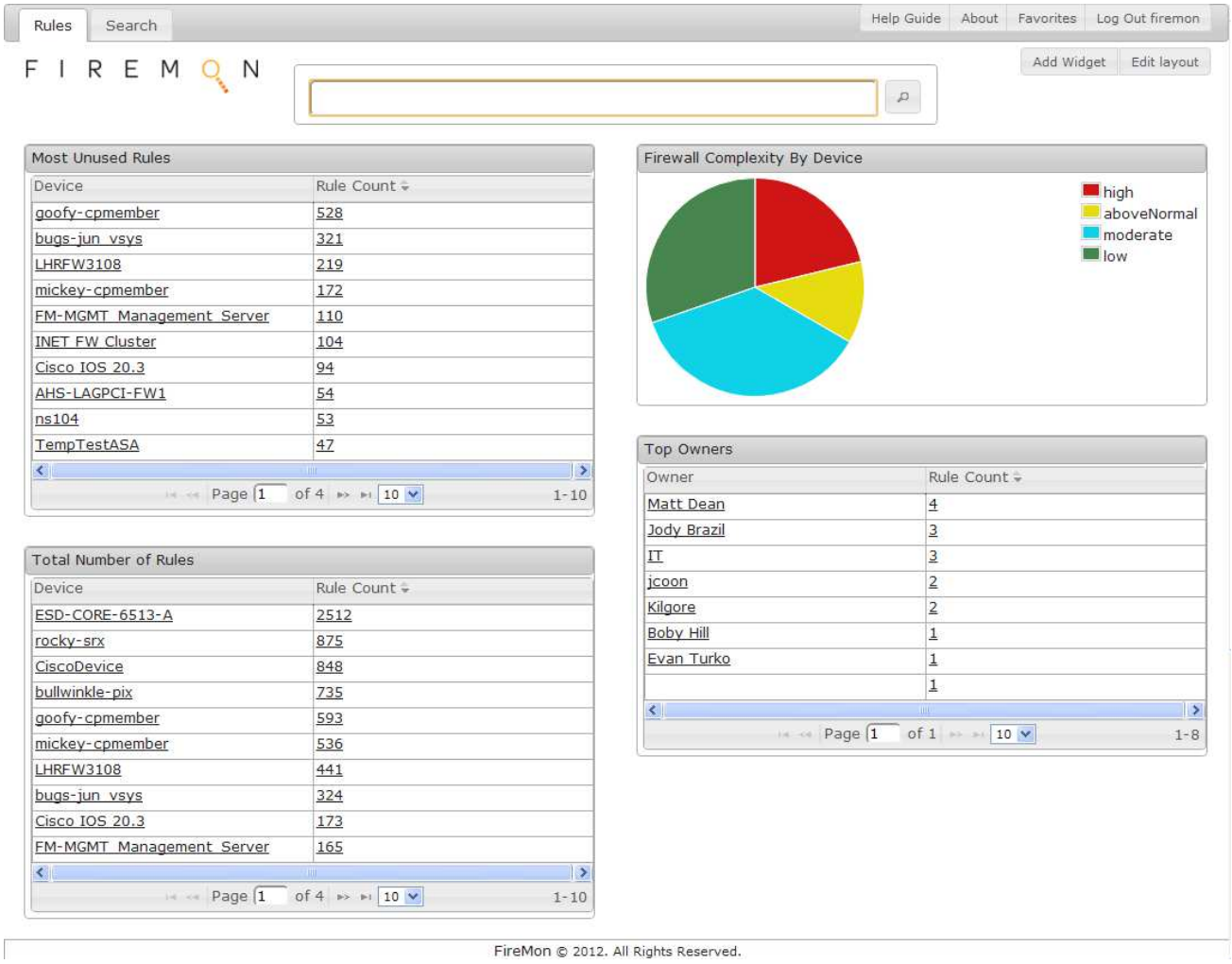


Here is an example of FIREMON rules:



	device.name	policy.name	rule.num	rule.name	rule.dis	rule.source	rule.destination	rule.service	rule.action
1	Cisco IOS 20.3	99	10		false	192.168.20.84	* Any	* Any	Accept
2	Cisco IOS 20.3	99	20		false	192.168.20.83	* Any	* Any	Accept
3	Cisco IOS 20.3	102	10		false	192.168.20.83	192.168.20.3	tcp-telnet[tcp]	Accept
4	Cisco IOS 20.3	102	20		false	192.168.20.83	192.168.20.125	tcp-telnet[tcp]	Accept
5	Cisco IOS 20.3	102	40		false	192.168.20.83	192.168.20.0/25	tcp-ftp[tcp]	Accept
6	Cisco IOS 20.3	102	50		false	192.168.20.0/25	192.168.20.0/25	tcp-cmd[tcp]	Accept
7	Cisco IOS 20.3	102	60		false	192.168.20.0/25	192.168.20.0/25	udp-netbios-ns [udp]	Accept
8	Cisco IOS 20.3	102	70		false	192.168.20.0/25	192.168.20.0/25	udp-netbios-dgm [udp]	Accept
9	Cisco IOS 20.3	102	80		false	192.168.20.0/25	192.168.20.0/25	icmp-echo [icmp]	Accept

Once the rule defined, this rule will be inserted in the dashboard to generate the following report:



4.1.3 COMPATIBILITY IN TUFIN & FIRMON

Both products are compatible with following Firewalls' equipment: CheckPoint, Cisco ASA, Juniper, Palo Alto and Fortinet. However, they are both not compatible with "Netasq".

4.2 RESEARCH PROJECTS

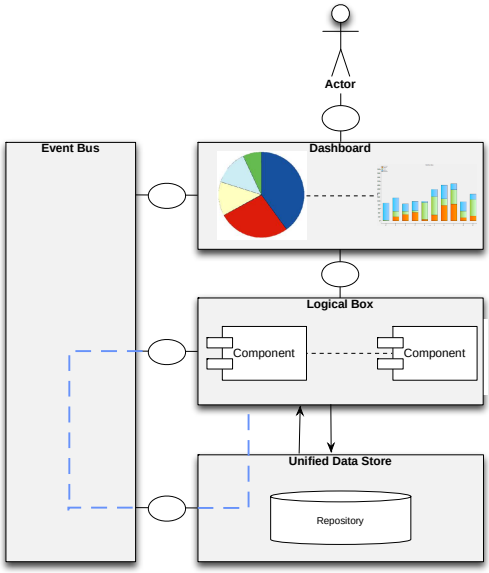
PREDYKOT and the three following projects are based on Access Control to deliver the State of the Art. Projects are related in terminologies like: Configuration, Refinement,

4.2.1 NIST PROJECT – POLICY MACHINE

The National Institute of Standards and Technology (NIST) is involved in many projects with variety of divisions (e.g. Information Security, Computer Security, etc.), and domains (e.g. Healthcare, Energy, Information Technology, etc.). In December 2012, NIST has launched, within the division of Computer Security, a new Access Control Framework. Policy Machine is a fundamental change in the methodology of expressing and enforcing a Policy. PM is a general-purpose access control mechanism that can express and enforce, through the usage of policy configuration settings, arbitrary, organization-specific, attribute-based access control policies [NIST IR-xxxx].

4.2.2 PROSECCO

PREDYKOT can retain from monitoring the state of the art, the innovation presented by ProSecCo in 2013 to build a model of models (meta-model) because of the count of models number currently in the Access Control field. Their main goal summarized by realizing an environment supporting the top-down definition of security policies and its efficient maintenance (i.e. based on requirements of future information systems). Top-Down approach depends on series of models with relation to Business, IT and Infrastructure levels. ProSecCo proposed policy chain represents the abstraction level that connects security requirements representation (i.e. Business level) to the configuration (i.e. Infrastructure level).
The IT Policy is their security policy representation as declarative specification (i.e. it specifies only the set of admissible configurations). Its definition is based on the use of a meta-model. The meta-mode is a class diagram modelled with UML language. It is composed of six sub-models representing the following: security principals, security associations, resources, privileges, authentication properties, and security domains.



ProSecCo architecture contains a model repository that enables storing unified functional and Security models. The View is a dashboard for the end-user and have a user interfaces (UI). The main goal of having the business logic box is to be decoupled from the User Interface (UI) (i.e. to support the harmonization for different prototypes). The Model + The View are what construct or ensure the Controller Accesses. The latter represents a “Logical Decision Point”. When actions start to be triggered, the Controller retrieves, builds, or modifies a model based on them and then decides which view to illustrate. HTTP-based Protocol is used for interactions between components. The communication in the system is divided to three categories: 1) Direct communication between components, 2) Indirect communication passing by the event bus till the model repository 3) Event bus is producer/consumer mechanism that allows components to exchange events about updates on the model.

Policy Refinement:

The policy refinement takes as input a conflict-free set of IT Policies as well as security capabilities described in the Infrastructure model, and transforms the declarative IT Policies into a consistent set of desired Configurations that serves as input for the generation and deployment of new Configuration settings. The refinement of policies towards a deployable Configuration relies on the description of the current landscape, which supports the decision process by eliminating unfeasible configuration alternatives and as such reducing the overall space of all possible configuration options. ProSecCo should be able to support the detection of conflicts for a given set of Security Requirements. Then it performs consistency checks between policies and provides different levels of support to the user in order to solve the detected conflicts.

4.2.3 ANIKETOS

One of Aniketos's main objectives is presenting a solution for "Overriding Access Control":

- Allows the user to temporarily extend his permissions
- Also known as Break-Glass or Break the Glass (BTG)

Aniketos relies on a post-hoc audit to evaluate the override:

- Effort for auditing overrides increases costs
- Support auditor to reduce time and effort.

"Aniketos is about establishing and maintaining trustworthiness and secure behaviour in a constantly changing service environment. The project aligns existing and develops new technology, methods, tools and security services that support the design-time creation and run-time dynamic behaviour of composite services, addressing service developers, service providers and service end users. Socio-technical viewpoint of the project needs a new way to model socio-technical requirements, which must be integrated in the Aniketos platform. Existing approaches that address security early in the requirements phases are suitable to model organizations, but not appropriate for cross-organizational settings as in the case of Aniketos environment. Therefore, to be able to capture such dynamic, service- oriented setting and analyse security and trust properties, we need a new organizational modelling language, namely the socio-technical security modelling language. Aniketos provides methods for analysing, solving, and sharing information on how new threats and vulnerabilities can be mitigated. The project constructs a platform for creating and maintaining secure and trusted composite services. Specifications, best practices, standards and certification work related to security and trust of composite services is promoted for inclusion in European reference architectures. Our approach to achieving trustworthiness and security of adaptive services takes account of socio-technical aspects as well as basic technical issues." -- (M. B. UKL, UKL, Mouzakitis, & Szkuta, n.d.)

5 CONCLUSION

CEP rules are powerful enough to detect and describe security events with a meaningful way close enough to the business rules. And it is adaptable enough to interact with XACML architecture.
This combination could be a good proposition that keep the powerful of issuer in XACML and gain the effectiveness of complex events.

This proposal is consistent with the PREDYKOT global architecture: thanks to the flexibility of the PREDYKOT architecture, any translation for a new proposed approach based on the study we presented can be adapted and integrated with PREDYKOT Architecture.

6 REFERENCES

1. S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, A. Ghalsasi, “Cloud computing - The business perspective”. Decision Support Systems Journal, Volume 51, Issue 1, pages: 176-189, (2011)

2. Y. Chen, V. Paxson, R.H. Katz, “What’s New About Cloud Computing Security?”, Technical report number UCB/EECS-2010-5, Univ. of California, Berkeley, (2010). Available from: www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-5.html.

3. Q. Zhang, L. Cheng, R. Boutaba, “Cloud computing: state-of-the-art and research challenges”, The Brazilian Computer Society 2010, Journal of Internet Services and Applications, Volume 1, Issue 1, pages: 7–18, (2010)

4. P Mell, T Grance, “A NIST Definition of cloud computing” Technical report, National Institute of Standards and Technology publications, (2009). Web link: www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf

5. H. Takabi, J. D. Joshi, G. J. Ahn, “Security and Privacy Challenges in Cloud Computing Environments”, Co-published By The IEEE Computer And Reliability Societies, 1540-7993/10, Volume 8, Issue 6, pages: 25 -3, (2010).

6. Cloud Security Alliance (CSA). “Security Guidance for Critical Areas of Focus in Cloud Computing V3”; (2011). Available from: https://cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf

7. S. Subashini and V. Kavitha. 2011. Review: “A survey on security issues in service delivery models of cloud computing”. Journal of Network and Computer Applications, Volume 34, Issue (11), pages: 1-11, (2011)

8. W. Jansen, T. Grance, “Guidelines on Security and Privacy in Public Cloud Computing”, NIST Special Publication. Number 800-144 (2011), available from: http://csrc.nist.gov/publications/nistpubs/800-144/SP800-144.pdf

9. D. Bernstein, E. Ludvigson, K. Sankar, S. Diamond, and M. Morrow, “Blueprint for the Intercloud - Protocols and Formats for Cloud Computing Interoperability”. In Proceedings of the 2009 Fourth International Conference on Internet and Web Applications and Services (ICIW '09). IEEE Computer Society, Washington, DC, USA, 328-336, (2009).

10. E. Bertino, F. Paci, R. Ferrini, “Privacy-Preserving Digital Identity Management for Cloud Computing”, IEEE Computer Society Data Engineering Bulletin, pages: 1–4, (2009).

11. S. Dawson, S. Qian, P. Samarati, “Providing security and interoperation of heterogeneous systems, Distributed and Parallel Databases”, 8:119–145, (2000).

12. Y. Zhang, J. Joshi, “Access Control and Trust Management for Emerging Multi domain Environments”, Annals of Emerging Research in Information Assurance, Security and Privacy Services, S. Upadhyaya and R.O. Rao, eds., Emerald Group Publishing, pages: 421–452, (2009).

13. E. Bertino, C. Brodie, S. B. Calo, L. F. Cranor, C. Karat, J. Karat, N. Li, D. Lin, J. Lobo, Q. Ni, P. R. Rao, X. Wang, “Analysis of privacy and security policies”, IBM Journal of Research and Development, Volume 53, Issue 2, Pages: 225-241, (2009).

14. V. Hu, F. Ferraiolo, R. Kuhn, “Assessment of Access Control Systems,” National Institute of Standards and Technology Inter agency report number 7316, (2006). Available from: http://csrc.nist.gov/publications/nistir/7316/NISTIR-7316.pdf

15. T. Chen, Y. Chen, H. Chu, C. Wang, “Development of an access control model, system architecture and approaches for resource sharing in virtual enterprise” Journal Computers Industry Volume 58, Issue 1, pages 57–73, (2007).

16. R. H. Khan, “Applicability Analysis of Grid Security Mechanisms on Cloud Networking”, In proceedings of the International Conference of cloud and service computing (CSC), pages : 65-70, (2011)

17. D.W. Chadwick , A. Otenko, E. Ball., “Role-based access control with X.509 attribute certificates”. IEEE Internet Computing Journal. Volume 7, Issue 2, pages 62-69 (2003)

18. M. Thompson, W. Johnston, S. Mudumbai, G. Hoo, K. Jackson, A. Essiari, "Certificate-based Access Control for Widely Distributed Resources", Proceedings of the Eight Usenix Security Symposium, pages : 23-36, (1999).

19. R. Alfieri , R. Cecchini , V. Ciaschini , L. dell Agnello,. A. Frohner, A. Gianoli, K. Lorentey, and F. Spataro, “VOMS, an Authorization System for Virtual Organizations”. Proceedings of European Across Grids Conference, pages: 33-40, (2003).

20. Y. Demchenko,"Virtual organisations in computer grids and identity management", Information Security Technical Report, Volume 9 Issue 1, pages:59–76, (2004).

21. J. G. Chen, R. C. Wang, H. Y. Wang, "The extended RBAC model based on grid computing", The Journal of China Universities of Posts and Telecommunications, Volume 13, Issue 3, pages:93 – 97, (2006).

22. L. Popa, M. Yu, S Y. Ko, S. Ratnasamy, I. Stoica, "CloudPolice: Taking Access Control out of the Network", In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks (Hotnets-IX). ACM, New York, NY, USA, pages:1-7, (2010).

23. XACML Standard, eXtensible Access Control Markup Language, Web link: http://www.oasisopen.org/committees/xacml/

24. ContentGuard, Inc.," eXtensible Rights Markup Language", XrML 2.0, (2001). Web link: http://www.xrml.org

25. G. Karjoth, M. Schunter, "A privacy policy model for enterprises". In 15th IEEE Computer Security Foundations Workshop (CSFW), pages: 271-281, (2002).

26. T. Yu, A. Li , I. Anton, "A formal semantics for P3P". In the proceedings of the ACM Workshop on Secure Web Services, Fairfax, VA, USA, pages: 1-8, (2004), available from: http://citeseer.ist.psu.edu/750176.html.

27. L. Kagal, T. Berners-Lee, D. Connolly, D. Weitzner, "Promoting Interoperability Between Heterogeneous Policy Domains", In the proceedings of the W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement, (2006), web link: www.w3.org/2006/07/privacy-ws/papers/32-kagal-rein/

28. A. Uszok, J.M. Bradshaw, R. Jeffers, M. Johnson, A. Tate, A., Dalton, J. and Aitken, S. "KAoS Policy Management for Semantic Web Services", IEEE Intelligent Systems, pages: 32-41, (2004).

29. G. Tonti, J. M. Bradshaw, R. Jeffers, R. Montanari, N. Suri, A. Uszok, "Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder". In International Semantic Web Conference pages: 419-437, (2003).

30. T. W. Finin, A. Joshi, L. Kagal, J. Niu, R. S. Sandhu, W. H. Winsborough, B. M. Thuraisingham, "ROWLBAC: Representing Role Based Access Control in OWL", In ACM Symposium on Access Control Models and Technologies (SACMAT), pages : 73–82, (2008)

31. M. Sensoy, Timothy J. Norman, Wamberto W. Vasconcelos, and Katia Sycara. "OWL-POLAR: A framework for semantic policy representation and reasoning". Journal of Web Semantics. Volume, 12, pages: 148-160, (2012).

32. J. Park, R. Sandhu, “Towards Usage Control Models: Beyond Traditional Access Control”, In the proceedings of the seventh ACM symposium on Access control models and technologies, SACMAT '02, Pages 57-64 , (2002).

33. I. Sriram, A. Khajeh-Hosseini, “Research Agenda in Cloud Technologies”, CoRR online Journal, (2010), web link: http://arxiv.org/abs/1001.325

34. Multipol ITEA 2 project, web link: http://www.itea2-multipol.org/project.php

35. R. Sandhu, R. Boppana, R. Krishnan, J. Reich, T. Wolff, J. Zachary, “Towards a Discipline of Mission-Aware Cloud Computing”, In Proceedings of the 2nd ACM Cloud Computing Security Workshop (CCSW 10), Chicago, Illinois, October 08, 2010, pages: 13-17

36. C. Henrich , M. Huber, C. Kempka , J. Muller-Quade, R. Reussner, “Towards Secure Information Sharing Models for Community Cyber Security”, Technical Report: Secure Cloud Computing through a Separation of Duties, submitted to European Symposium on Research in Computer Security (ESORICS 2010).

37. Cloud Best Practices. “Cloud 2.0 applications”, in “cloud computing best practices guide”. blog (CloudBestPractices.info), (2011), Available from: http://cloudbestpractices.net/

38. F. Cuppens, A. Miege, “Modelling Contexts in the Or-BAC Model”, In 19th Annual Computer Security Applications Conference (ACSAC ’03), (2003).

39. W. Hommel, “Using XACML for Privacy Control in SAML-Based Identity Federations”, IFIP International Federation for Information Processing CMS 2005 LNCS 3677 pages:160-169, (2005).

40. S. Park, Y. Han, T. Chung, “Context-Aware Application, In High Performance Computing and Communications”, Second International Conference, HPCC 2006, Munich, Germany, September 13--15, 2006, Proceedings, volume 4208 of Lecture Notes in Computer Science, pages: 572-580. Springer, (2006).

41. D. Ferraiolo, R. Kuhn, R. Sandhu, RBAC Standard Rationale: Comments on “A Critique of the ANSI Standard on Role-Based Access Control”, IEEE Security & Privacy Journal, Volume 5, Issue 6, pages 51-53, (2007).

42. A. Pretschner, M. Hilty, F. Schütz, C. Schaefer, T. Walter, “Usage Control Enforcement: Present and Future”, IEEE Security & Privacy Journal, Volume 6, Issue 4, pages: 44-53, (2008)

43. A. Lazouski, F. Martinelli, P. Mori, Usage control in computer security: A survey, Computer science review, Volume 4, pages: 81-99, Elsevier, (2010)

44. D. Chen, X. Huang, X. Ren, “Access Control of Cloud Service Based on UCON”. In proceedings of CloudCom 2009, pages: 559-564, (2009).

45. X. Zhang, M. Nakae, M. Covington, R. Sandhu, “Toward a Usage-Based Security Framework for Collaborative Computing Systems”, ACM TISSEC Journal, Volume 11, Issue 1, pages 1-36, (2008)

46. H. Lee, H. Luedemann, “A Lightweight Decentralized Authorization Model for Inter-domain Collaborations”, In the proceedings of the ACM workshop on Secure web services, pages = 83-89, (2007).

47. D. E. Bell, “Looking back at the bell-la padula model”, In the proceedings of the 21st Annual Computer Security Applications Conference, pages 337–351, (2005).

48. D. M. Cullough, “A hookup theorem for multilevel security”, IEEE Transactions on Software Engineering, Volume 16, Issue 6, pages : 563–568, (1990).

49. J. Park, R. Sandhu, “The UCONABC Usage Control Model”, ACM Transactions on Information and System Security, Volume. 7, Issue. 1, pages : 128–174, (2004)

50. I. Mavridis, C. Georgiadis, G. Pangalos, M. Khair, “Access Control based on Attribute Certificates for Medical Intranet Applications” , Jorunal of Medical Internet Research ;Volume 3, Issue 1, (2001).

51. A. Gouglidis and I. Mavridis, “On the Definition of Access Control Requirements for Grid and Cloud Computing Systems Networks for Grid Applications”, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Volume 25, Springer Berlin Heidelberg, ISBN 978-3-642-11732-9, (2010)

52. S. I. Gavrila, J. F. Barkley, “Formal specification for role based access control user/role and role/role relationship management”. In the proceedings of the third ACM workshop on Role-based access control RBAC’98, pages 81–90, New York, NY, USA, (1998).

53. H. Chen, N. Li, “Constraint generation for separation of duty”, ACM Symposium on Access Control Models and Technologies, Lake Tahoe, California, USA, pages: 130–138, (2006).

54. R. Sandhu, V. Bhamidipati,”The ASCAA principles for next-generation role-based access control”. In the proceedings Availability, Reliability and Security,. ARES 08, xxvii--xxxii ,(2008)

55. Q. Ni, D. Lin, E. Bertino, J. Lobo, “Conditional privacy-aware role based access control”. In the proceedings of ESORICS'07, pages 72–89, (2007)

56. Q. Ni, E. Bertino, J. Lobo, “An Obligation Model Bridging Access Control Policies and Privacy Policies”, in Proceedings of the 13th ACM symposium on Access control models and technologies (SACMAT '08). ACM, New York, NY, USA, pages : 133-142, (2008).

57. J. Reid, I. Cheong, M. Henricksen, and J. Smith, “A Novel Use of RBAC to Protect Privacy in Distributed Health Care Information Systems”, In Proceedings of the 8th Australasian conference on Information security and privacy (ACISP'03), Rei Safavi-Naini and Jennifer Seberry (Eds.). Springer-Verlag, Berlin, Heidelberg, pages : 403-415, (2003).

58 . J. B. Joshi, R. Bhatti, E. Bertino, A. Ghafoor, “Access-control language for multidomain environments”, IEEE Internet Computing, Volume 8, pages : 40–50, (2004).

59. E. Freudenthal, T. Pesin, L. Port, E. Keenan, V. Karamchet, “dRBAC: Distributed Role-based Access Control for Dynamic Coalition Environments”, In the proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02), pages : 411-, (2002).

60. J. Filho, H. Martin, “A generalized context-based access control model for pervasive environments”, SPRINGL '09: Proceedings of the 2nd SIGSPATIAL ACM GIS International Workshop on Security and Privacy in GIS and LBS, pages : 21-21, (2009).

61. J. Joshi, E. Bertino, A. Ghafoor, “Temporal Hierarchies and Inheritance Semantics for GTRBAC”. In proceedings of the 7th ACM Symposium on Access Control Models, pages: 74-83, (2002).

62. E. Yuan and J. Tong. “Attributed Based Access Control (ABAC) for Web Services”. In Proceedings of the IEEE International Conference on Web Services (ICW’05), pages = 561—569, (2005)

63. GridTrust: Gridtrust, Available from : http://www.gridtrust.eu/gridtrust (2009)

64. Priebe, T., Dobmeier, W., Kamprath, N.: “Supporting attribute-based access control with ontologies”. In: ARES '06: Proceedings of the First International Conference on Availability, Reliability and Security, Washington, DC, USA, IEEE Computer Society 465--472 (2006)

65. J. Joshi, A. Ghafoor, W. Aref, E. H. Spafford, Digital government security infrastructure design challenges, IEEE Computer , Vol. 34, No. 2, February 2001, pp 66-72, February 2001.

66. N. Ragouzis, J. Hughes, R. Philpott, E. Maler, “Security Assertion Markup Language (SAML) V2.0”, Technical Overview Committee Draft, (2008). Available from : https://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf

67. P. A. Bonatti, S. D. C. diVimercati, P. Samarati, “An algebra for composing access control policies”, ACM Transactions on Information System Security, Volume 5, Issue 1, (2002)

68. J Kim, S Hong , “Consolidated Authentication Model in Cloud Computing Environments”, Available from: http://www.sersc.org/journals/IJMUE/vol7_no3_2012/18.pdf

69. W. Tfaili, A. Chibani, Y. Amirat, “On the Composition of Access Control Management in Multidomain Environment”. Proceedings of the IEEE Information Reuse and Integration (IRI 2010), Las Vegas (USA), pages: 153-158, (2010)

70. P. Rao, D. Lin, E. Bertino, N. Li, J. Lobo, Fine-Grained Integration of Access Control Policies, Computer & Security, Volume 30, Issue 2, pages: 91-107, (2011)

71. D. Wijesekera, S. Jajodia, A Propositional Policy Algebra for Access Control, ACM Transactions on Information and System Security, Volume 6, Issue 2, pages: 286–325, (2003)

72. Open Digital Rights Language Initiative, Open Digital Rights Language, Available from: http://odrl.net/. 2001.

73. S. Hada, M. Kudo, “XML Access Control Language: Provisional Authorization for XML Documents”, Tokyo Research Laboratory, IBM Research (2000), Available from : http://www.research.ibm.com/trl/projects/xml/xacl/xacl-spec.html

74. S. Pearson. 2009. Taking account of privacy when designing cloud computing services. In Proceedings of ICSE Workshop on Software Engineering Challenges of Cloud Computing (CLOUD '09). IEEE Computer Society, Washington, DC, USA, pages : 44-52, (2009).

75. N. Damianou, N. Dulay, E. Lupu, M. Sloman, “The Ponder Policy Specification Language”, In the proceedings of the International Workshop on Policies for Distributed Systems and Networks, POLICY '01, Bristol, U.K., Springer-Verlag, Pages 18-38, (2001).

76. B. Lang,I. Foster, F. Siebenlist, R. Ananthakrishnan, T. Freeman, “Attribute Based Access Control for Grid Computing”. Journal of Grid computing, Volume 7, Issue 2, pages: 169-180, (2009).

77. A. C. Squicciarinia, F. Pacib, E. Bertino, “Trust establishment in the formation of Virtual Organizations”, Computer Standards & Interfaces,
Volume 33, Issue 1, pages: 13-23, (2011).

78. R. Bhatti, J. Joshi, E. Bertino, A. Ghafoor, “Access Control in Dynamic XML-based Web-Services with X-RBAC”, The First International Conference on Web Services, Las Vegas, June 23-26, pages : 243-249, (2003).

79. XACML Profile for RBAC, Available from : http://docs.oasis-open.org/xacml/cd-xacml-rbacprofile-01.pdf

80. P. Samarati, Extending XACML for Open Web-based Scenarios, W3C Workshop on Access Control Application Scenarios, (2009), Available from : http://www.w3.org/2009/policy-ws/papers/Samarati.pdf

81. X. Zhang, M. Nakae, M.J. Covington, R. Sandhu, “A usage based authorization framework for collaborative computing systems”. In the proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies SACMAT.06, New York, NY, USA, pages:. 180-189, (2006)

82. M. Sastry, R. Krishnan, R. Sandhu, “A new modeling paradigm for dynamic authorization in multi-domain systems, in: Communications in Computer and Information Science”, Springer, Berlin, Heidelberg, Volume 1, pages : 153-158, (2007)

83. J. Park, X. Zhang, R.S. Sandhu, “Attribute mutability in usage control”, in: DBSec, Kluwer, pages: 15.29, (2004).

83. XACML v3.0 Privacy Policy Profile Version 1.0 Committee Specification 01, 10 August 2010, available from : http://docs.oasis-open.org/xacml/3.0/xacml-3.0-privacy-v1-spec-cs-01-en.pdf

84. Anne H. Anderson, “A comparison of two privacy policy languages: EPAL and XACML”. In proceedings of the 3rd ACM workshop on Secure web services (SWS '06). ACM, New York, NY, USA, pages : 53-60, (2006).

85 M.Y. Becker, C. Fournet, A.D. Gordon: “SecPAL: Design and semantics of a decentralized authorization language”. Journal of Computer Security - Digital Identity Management, Volume 18 Issue 4, Pages 619-665, (2010).
86. Protune Policy Framework, Available from : http://policy.l3s.uni-hannover.de:9080/policyFramework/protune/

87. Y.J. Hu and H. Boley, "SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies", In the proceedings of IEEE Web Intelligence Conference (WI 2010), Pages 302-307, (2010)

88. J. Diggelen, J. M. Bradshaw, M. Johnson, A. Uszok, P. J. Feltovich, “Implementing Collective Obligations in Human-Agent Teams Using KAoS Policies”. AAMAS, IJCAI, MALLOW, pages : 36-52, (2009).

89. Rei , Available from : http://rei.umbc.edu/

90. OWL Ontology Language, Available from: http://www.w3.org/TR/owl-features/

91. L. Kagal, T. Finin, A. Joshi. 2003. “A Policy Language for a Pervasive Computing Environment”. In Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY '03). IEEE Computer Society, Washington, DC, USA, pages: 63-68, (2003).

92. R. Ferrini, E. Bertino, “Supporting RBAC with XACML+OWL”, In the proceedings of the 14th ACM symposium on Access control models and technologies, SACMAT, pages: 145-154, (2009).

93 E. Prud’hommeaux, A. Seaborne, SPARQL Query Language for RDF, Tech. Rep., W3C, (2006). Available from: http://www.w3.org/TR/rdf-sparql-query.

94 E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur, Y. Katz, “Pellet: a practical OWL-DL reasoner, Web Semantics”, Web Semantics: Science, Services and Agents on the World Wide Web. Volume 5, Issue 2, pages 51–53, (2007).

95 Pellet: OWL 2 Reasoner for Java, Available from: http://clarkparsia.com/pellet/

96 F. Kelbert and A. Pretschner. Towards a policy enforcement infrastructure for distributed usage control. In Proceedings of the 17th ACM symposium on Access Control Models and Technologies (SACMAT '12). ACM, New York, NY, USA. Pages =119-122, (2012).

97. E. Damiani, S. D. C. di Vimercati, S. Paraboschi, P. Samarati, “A Fine Grained Access Control System for XML Documents”, ACM Transactions on Information and System Security, Volume 5, Issue 2, (2002).

98. Y.J. Hu and H. Boley, “SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies”, In the proceedings of the IEEE Web Intelligence (WI) Conference 31-Sep. 3, (2010).

99, A. C. O’Connor, R. J. Loomis, “Economic Analysis of Role-Based Access Control, Final Report”, NIST public report, (2011) Available from: http://csrc.nist.gov/groups/SNS/rbac/documents/20101219_RBAC2_Final_Report.pdf,.

Adi, A., & Etzion, O. (2003). Amit - the situation manager. The VLDB Journal The International Journal on Very Large Data Bases, 13(2), 177–203. doi:10.1007/s00778-003-0108-y

Barker, S. (2009). The next 700 access control models or a unifying meta-model? Presented at the SACMAT '09: Proceedings of the 14th ACM symposium on Access control models and technologies, ACM Request Permissions. doi:10.1145/1542207.1542238

Boutaba, R., & Aib, I. (2007). Policy-based management: A historical perspective. Journal of Network and Systems Management, 15(4), 447–480. doi:10.1007/s10922-007-9083-8

Etzion, O., & NIBLETT, P. (2011). Event Processing in Action. Manning Publications Co.
Fülöp, L. J., Tóth, G., Rácz, R., Pánczél, J., Gergely, T., Beszédes, A., & Farkas, L. (2010). Survey on complex event processing and predictive analytics.

Hamerling, C. (n.d.). D1.1 – State of the Art. play-project.eu. Retrieved March 5, 2013, from http://play-project.eu/documents/viewdownload/3/13

Jin, X., Krishnan, R., & Sandhu, R. (2012). A Unified Attribute-Based Access Control Model Covering DAC, MAC and RBAC. In N. Cuppens-Boulahia, F. CUPPENS, & J. Garcia-Alfaro (Eds.), Lecture Notes in Computer Science (Vol. 7371, pp. 41–55–55). Springer Berlin Heidelberg. doi:10.1007/978-3-642-31540-4_4

Jude, M. (2001). Policy-Based Management: beyond the hype. Business Communication Review, 52–56.

UKL, M. B., UKL, K. G. T., Mouzakitis, S., & Szkuta, K. (n.d.). Deliverable D3. 1 Initial Gap Analysis Report. Seventh Framework Programme. Seventh Framework Programme. Retrieved from http://www.aniketos.eu/content/downloads

. LAST PAGE .