Adapting the Model Driven Security strategy to generate contextual security policy for multi-cloud systems

Authors : W. F. Ouedraogo[1], F. Biennier[2] , P. Ghodous[3]

[1, 2]Université de Lyon, CNRS INSA-Lyon. LIRIS. UMR5205. F-69621. France,

[3]Université de Lyon, CNRS Université Claude Bernard Lyon 1. LIRIS. UMR5205. F-69621, France

E-mail: wendpanga-francis.ouedraogo@liris.cnrs.fr[1], frederique.biennier@liris.cnrs.fr[2] , ghodous@liris.cnrs.fr[3]

23/10/2013

Plan

- Context
- State of the art
- Model-Driven Security approach
- Conclusion and further works

Context

- Globalized economic environment involve for companies to :
  - focus on their core business
  - develop new collaborative strategies

  ➔build their IS (Information System) around on the Business Process (BP).
- SOA(Service Oriented Architecture) provides companies a new model [1]:
  - Build activities functionalities as business services and combine them dynamically with the partner companies service.

  ➔Interoperable, and agile services;

  ➔Open system mean security threats
- Collaborative IS involve to share data, service and BP(Business Process) coming from different companies.

  ➔ companies assets, which required to be protected

  ➔ each has its own security policies

Context

- To protect IS : EBIOS, MEHARI, OCTAVE approach [8]
  - Approach based on the vulnerabilities and threats analysis,
  - use knowledge bases

  ➔Not adapted to the dynamic environment imposed by process and SOA

  ➔Difficult and so long to implement

  ➔Not end users oriented (security expert is required)

- Cloud computing [3] emerge thank to :
  - Web 2.0
  - Development of broadband and network,
  - Virtualization

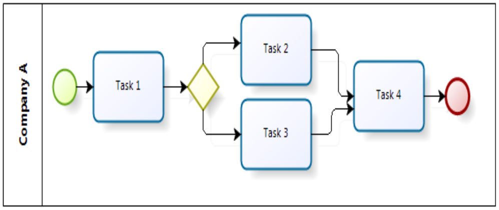
  ➔New solution to consume services and deploy collaborative IS (BP)

  ➔Allow to have on demand “unlimited” capacity for storage and processing

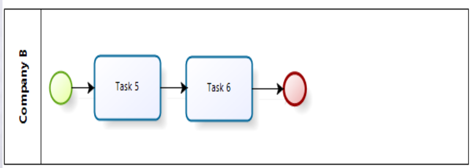
  ➔Involve a externalization strategy and new challenges to secure the SI

Context

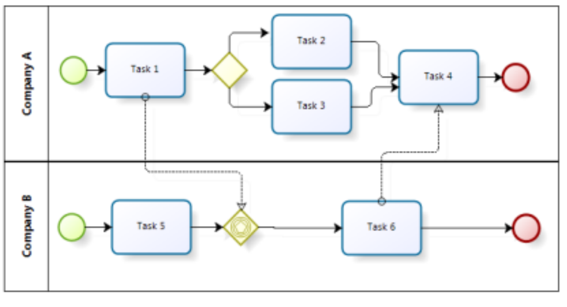
Challenge



Company A BP with security requirements



Company B BP with security requirements



Collaborate together

Secure BP take account
each compagny security requirements and
platforms specifications

Our approach based on a Model-Driven Engineering (MDE).

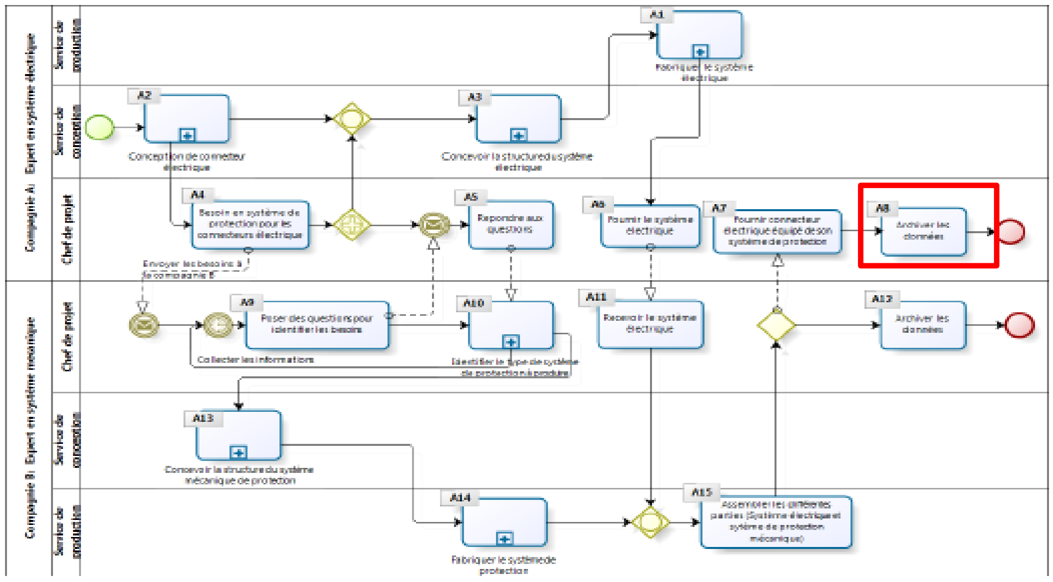
- identify BP security requirements of each company,
- define an adapted Quality of Protection,
- generate adapted security policies, paying attention on the deployment platforms.

State of the art

Business process modeling

- *Various types of modeling tools and languages : EPC, BPEL, WS-CDL, XPDL, BPMN,…*

- *BPMN is mostly used to describe flows between the different activities as well as “launching” conditions of a particular part of the process.*


Compagnie A: Expert en système électrique
Service de production
Service de conception
Chef de projet
Compagnie B: Expert en système mécanique
Chef de projet
Service de conception
Service de production
A1
Fabriquer le système électrique
A2
Conception de connecteur électrique
A3
Concevoir la structure du système électrique
A4
Besoin en système de protection pour les connecteurs électrique
A5
Reponrdre aux questions
A6
Fournir le système électrique
A7
Fournir connecteur électrique équipé de son système de protection
A8
Archiver les données
Envoyer les besoins à la compagnie B
A9
Poser des questions pour identifier les besoins
Collecter les informations
A10
Identifier le type de système de protection à produire
A11
Recevoir le système électrique
A12
Archiver les données
A13
Concevoir la structure du système mécanique de protection
A14
Fabriquer le système de protection
A15
Assembler les différentes parties (Système électrique et système de protection mécanique)

State of the art

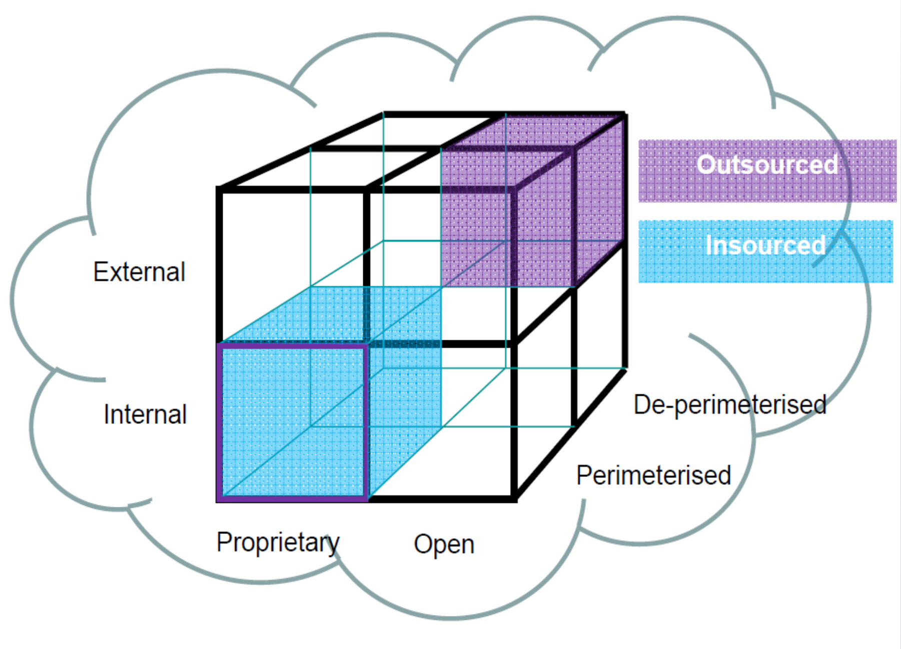
Secure BP

Framework Evaluate criteria	OpenPMF [7]	SECTET [6]	BP Sec [4]	KIT Serure BP[5]
Abstractions levels	PIM-PSM-code	PIM-PSM-Code	CIM-UML Use case (PIM)	PIM-PSM
Approach used	UML	Annotation based+ UML	UML	Annotation
Oriented end user	No	No	Yes	No
Automatic Policy generation	Yes	Yes		Yes
Modification language and transformation	UML+DSL	UML2+SECTET-DSL	UML +QVT	Ad-hoc
Take account infrastructure	No	No	No	No
Take account execution context	No	No	No	Yes
Security criteria	Authentication, Authorization, Monitoring	Encryption, Intégrité, Non-repudiation, Authentication	Non-Repudiation, Privacy, intrusion Détection, Access control, Authorization	Authorization
Policy monitoring	Yes	No	Yes	No
SecaaS (security as a Service)	No	Yes	No	No
Security Standard	XACML	SAML, WS-policy, XACML		XACML

State of the art

Cloud security

- **Cloud Cube Model: Selecting Cloud Formations for Secure Collaboration, Jericho Forum, Version 1.0, (April 2009) [2]**
  - Define cloud security cube model that allows companies to choose the type of cloud that is adapted to their business needs

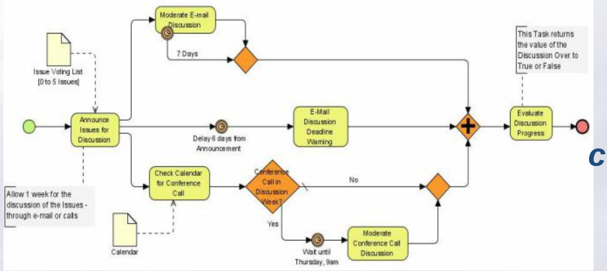


The Cloud Cube Model

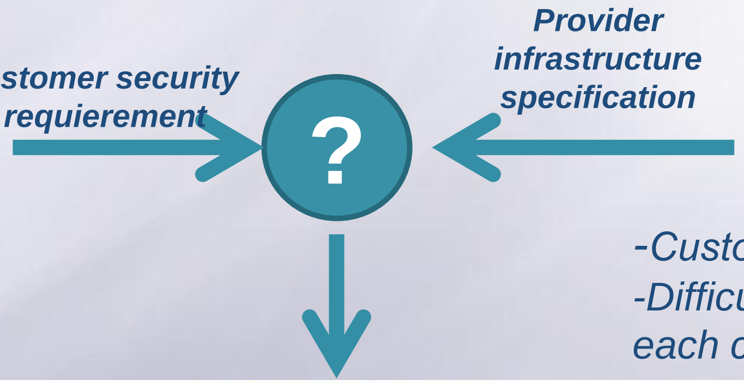
State of the art

conclusion

Business and application level


Moderate E-mail Discussion
7 Days
This Task returns the value of the Discussion Over to True or False
Issue Voting List [0 to 5 Issues]
Announce Issues for Discussion
Delay 6 days from Announcement
E-Mail Discussion Deadline Warning
Evaluate Discussion Progress
Check Calendar for Conference Call
Conference Call in Discussion Week?
No
Yes
Allow 1 week for the discussion of the Issues - through e-mail or calls
Calendar
Wait until Thursday, 9am
Moderate Conference Call Discussion

Infrastructure level



customer security requierement

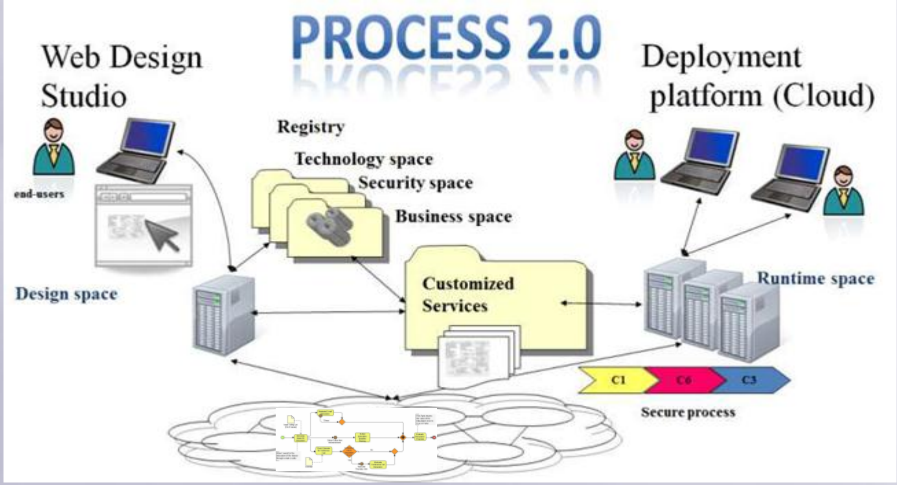
Provider infrastructure specification

-Do not pay more attention on vulnerabilities of infrastructure

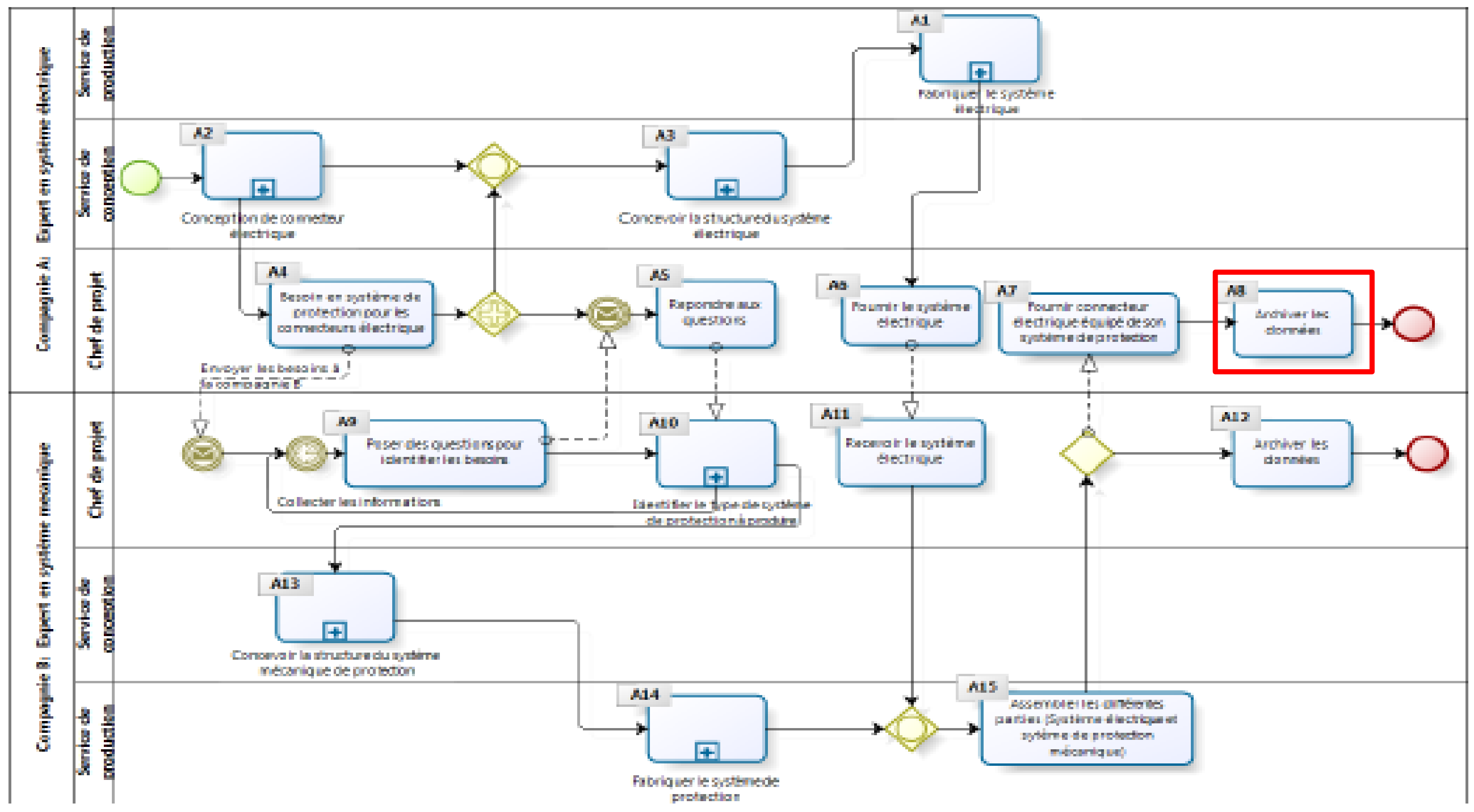
-Not end user oriented.

-Customers don’t trust providers

-Difficult for providers to enforce each company policies.

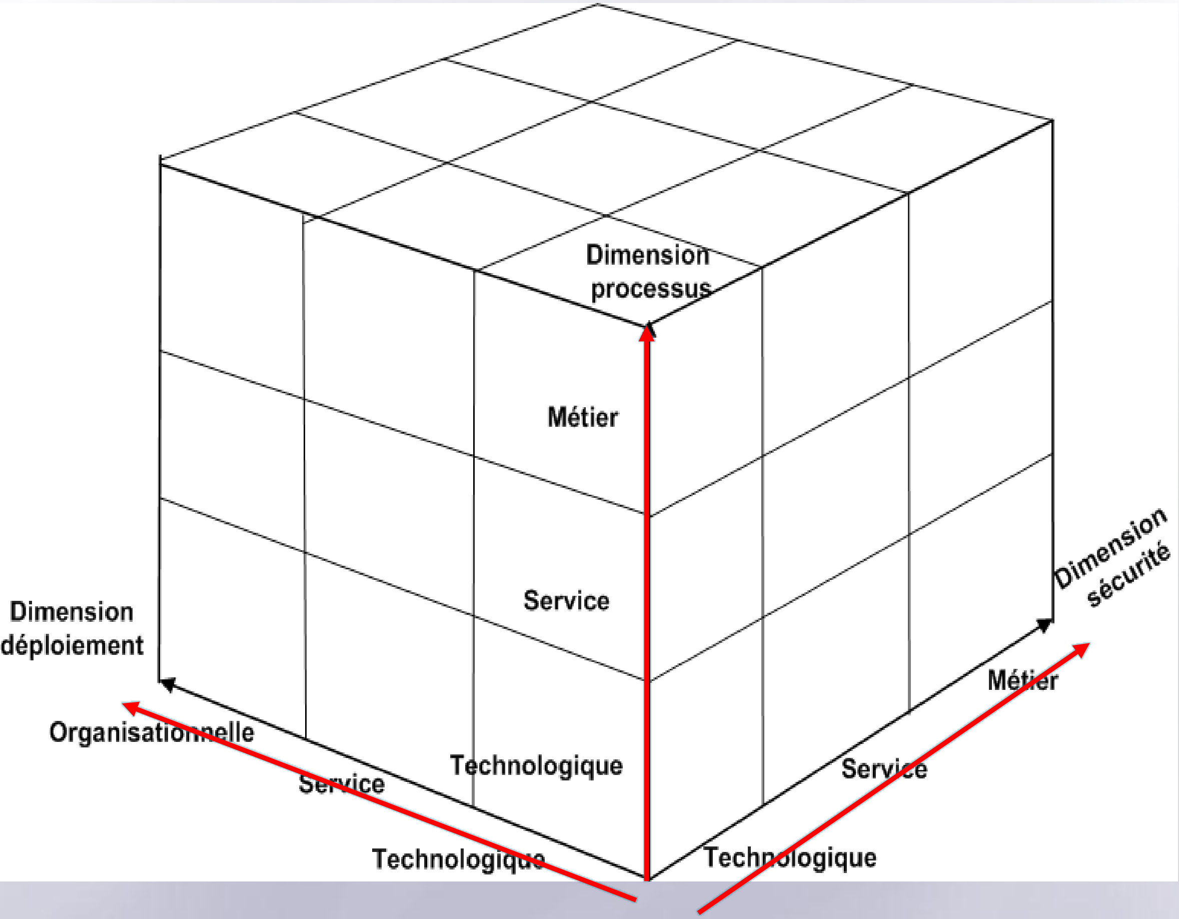


Model-Driven Security approach


Compagnie A: Expert en système électrique
Service de production
Service de conception
Chef de projet
Compagnie B: Expert en système mécanique
Chef de projet
Service de conception
Service de production
A1
Fabriquer le système électrique
A2
Conception de connecteur électrique
A3
Concevoir la structure du système électrique
A4
Besoin en système de protection pour les connecteurs électrique
A5
Repondre aux questions
A6
Fournir le système électrique
A7
Fournir connecteur électrique équipé de son système de protection
A8
Archiver les données
Envoyer les besoins à la compagnie B
A9
Poser des questions pour identifier les besoins
Collecter les informations
A10
Identifier le type de système de protection à produire
A11
Recevoir le système électrique
A12
Archiver les données
A13
Concevoir la structure du système mécanique de protection
A14
Fabriquer le système de protection
A15
Assembler les différentes parties (Système électrique et système de protection mécanique)

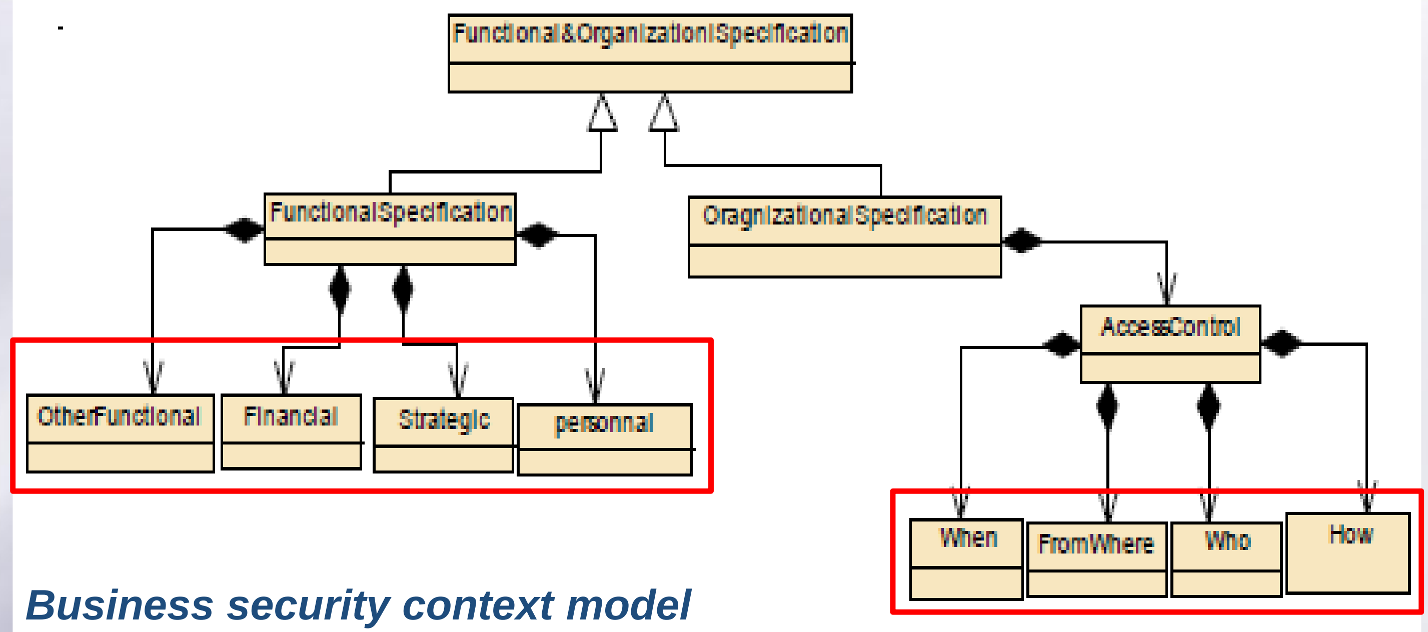
Model-Driven Security approach

Multidimensional model to secure BP



Multidimensional model

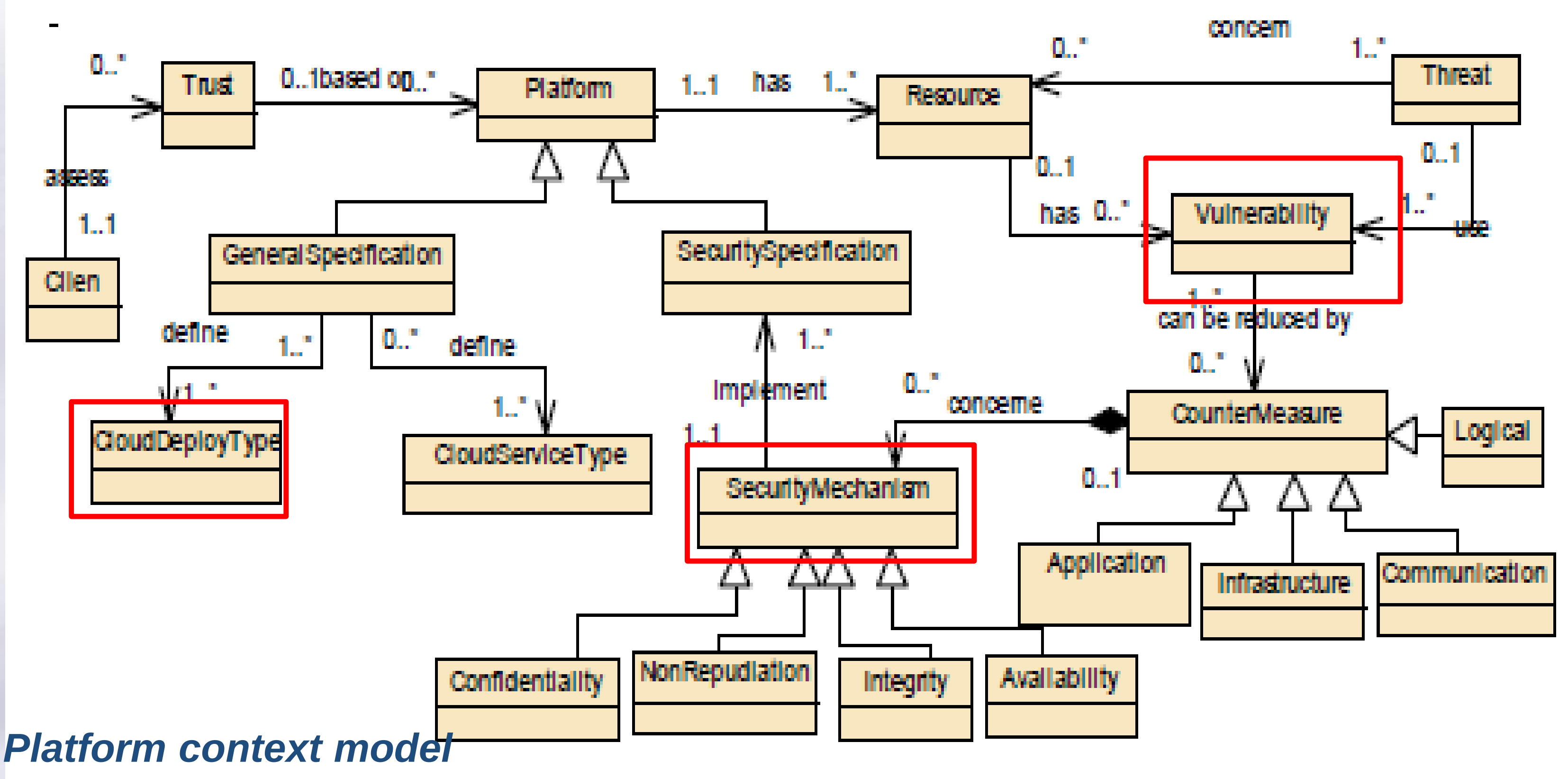
Weaving BP/Security : Business Security context Model



Business security context model

Multidimensional model

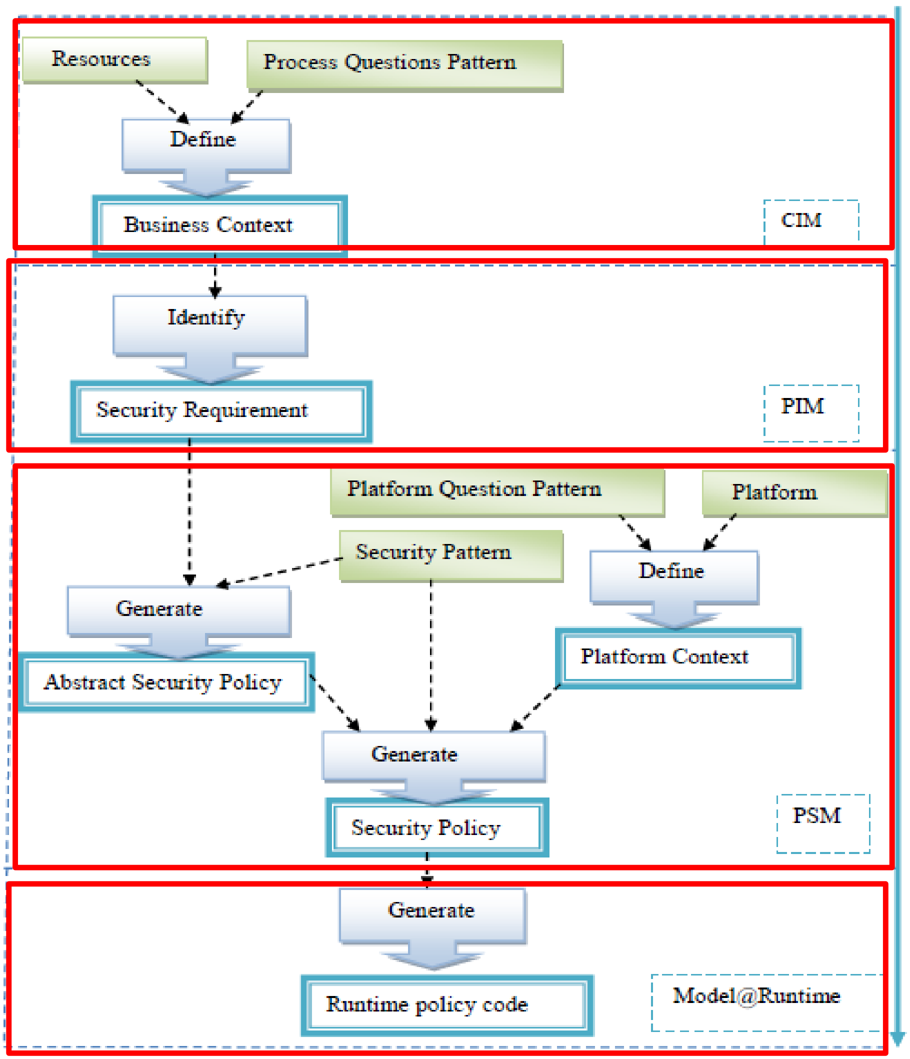
Weaving Deployment/Security : Platform Security context Model



Platform context model

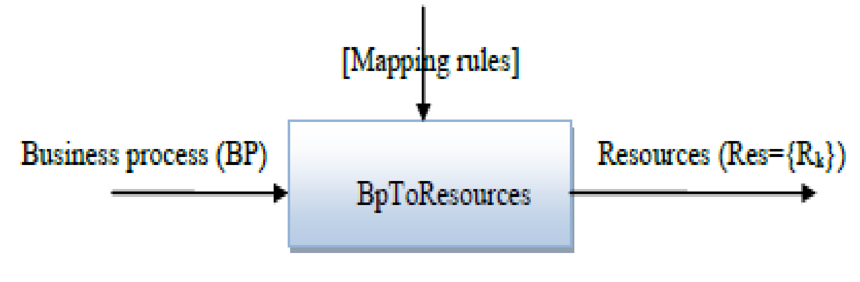
Model-Driven Security approach

MDS Approach



Model-Driven Security approach

CIM specification



R = (N, T , L, {R})
- N: Resource Name
- T: Resource Type
- L: Resource Layer
-U: the Resource URI (reference)
- R: Related Resources

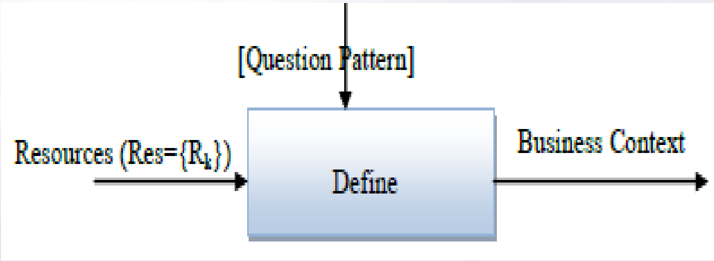
```
R1(A1" "Activity" "Business"," http://com.insa.bp/connecteur/A1",{S11})
R11("S1" "Service" "Business"," http://com.insa.bp/connecteur/A1",{D11,S12})
R3(A3" "Activity" "Business"," http://com.insa.bp/connecteur/A3",{S31})
R8("A8" "Activity" "Business"," http://com.insa.bp/connecteur/A8",{S81})
R81(S81" "Service" "Service"," http://com.insa.bp/connecteur/A8/S81",{ D811, D812})
R82(S82" "Service" "Service"," http://com.insa.bp/connecteur/A8/S82",{ D821, D822})
R811(D811" "Data" "Service"," http://com.insa.bp/connecteur/ A8/S81/D811",{ })
R812("D821","Data"," Service ","http://com.insa.bp/connecteur/A8/S81/D821",{})
....
...,,
```

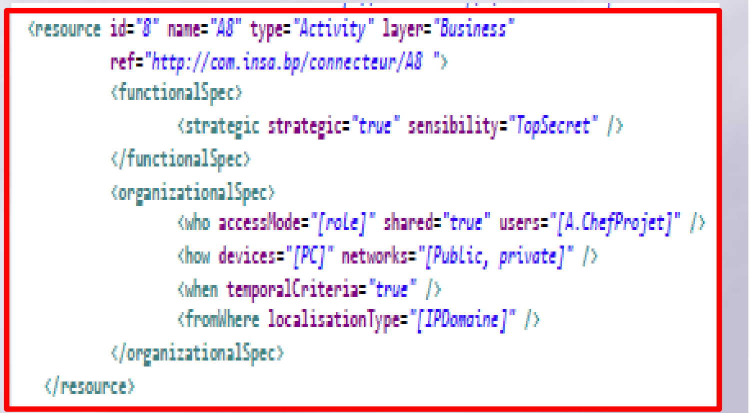
All the resources as :
$Res=\{R_i\}$ where $0<k<N_k$; (1)
Where "i" is the resource number and N_i the total of the all resources.

$Res(R_k)=\{ r \ / \ r \in Res \wedge r.N=Rk)$ (2)
Where R_k .is the resource Name

Model-Driven Security approach

CIM specification





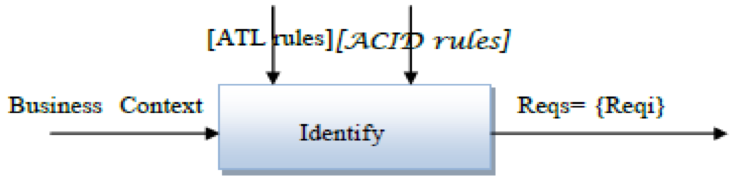
```
<resource id="8" name="A8" type="Activity" layer="Business"
        ref="http://com.insa.bp/connecteur/A8 ">
        <functionalSpec>
                <strategic strategic="true" sensibility="TopSecret" />
        </functionalSpec>
        <organizationalSpec>
                <who accessMode="[role]" shared="true" users="[A.ChefProjet]" />
                <how devices="[PC]" networks="[Public, private]" />
                <when temporalCriteria="true" />
                <fromWhere localisationType="[IPDomaine]" />
        </organizationalSpec>
  </resource>
```

N°		Questions	answers
Functional specifications			
1		Which services or activity manipulate personal data?	Any services and process
2		Which services or process manipulate financial data?	Any services and process
3		are there some activities in the process that handled data Strategic order (ie giving a strategic advantage to your business or associated with knowledge / expertise giving you a strategic advantage)?	yes
4		If yes, what level of sensitivity do you give to each activity which handle strategic data? Top secret? Secret? Access limited? public?	Limited [A9, A10, A15] Secret [A1, A3, A13, A14] Top secret[A8, A12]
Organizational specifications			
5		Are there activities that involve external stakeholders (partners, customers, ...)?	[A9, A10, A13]
7		Are there activities for which you wish to restrict access to specific time slots (eg access between 7 and 19h on working days)	Yes, ALL
8		For each activity subject to a restriction of access, how do you set the permissions: - Individually, ie giving a list of authorized users - For user groups (depending on their role, ...)	A.Production[A1] A.Conception[A2-A3] A.ChefProjet[A4-A8] B.ChefProjet[A9-A12] B.Production[A13] B.Conception[A14, A15]
9		Which means can you use to access to the resources (data or applications): - A public network (public Wifi, 3G network, home network of personal ...) - the company network (LAN, VPN) -Any Network	Any network

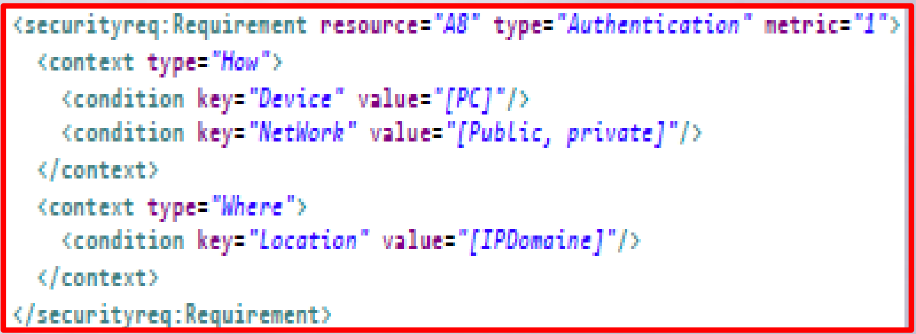
Model-Driven Security approach

CIM To PIM : define security requirement

$$M_{CIM2PIM}: \begin{array}{l} RES \longrightarrow REQS \\ (Res_1, Res_i \ldots Res_n) \longrightarrow \mathcal{M}_{CIM2PIM}(Res_1, Res_i \ldots Res_n) \end{array} \quad (8)$$



Reqi = (RR, (RT, RM), RG,{RCtx})
- RR (Requirement Resource)
- RT (Requirement Type)
- RM (Requirement Metrics)= [0-1]
-RG (Requirement Goal)
-RCtx (Requirement context)



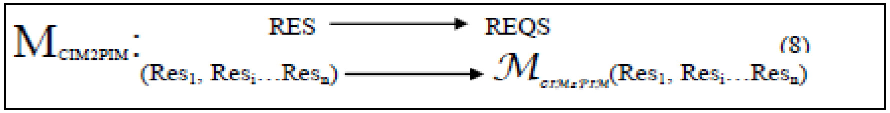
```
<securityreq:Requirement resource="A8" type="Authentication" metric="1">
  <context type="How">
    <condition key="Device" value="[PC]"/>
    <condition key="NetWork" value="[Public, private]"/>
  </context>
  <context type="Where">
    <condition key="Location" value="[IPDomaine]"/>
  </context>
</securityreq:Requirement>
```

All the requirements for all the resources as:
Reqs={Reqi} where 0<i<Ni; (3)
Where “i” is the requirement number and N the total of the all requirements of all resources.

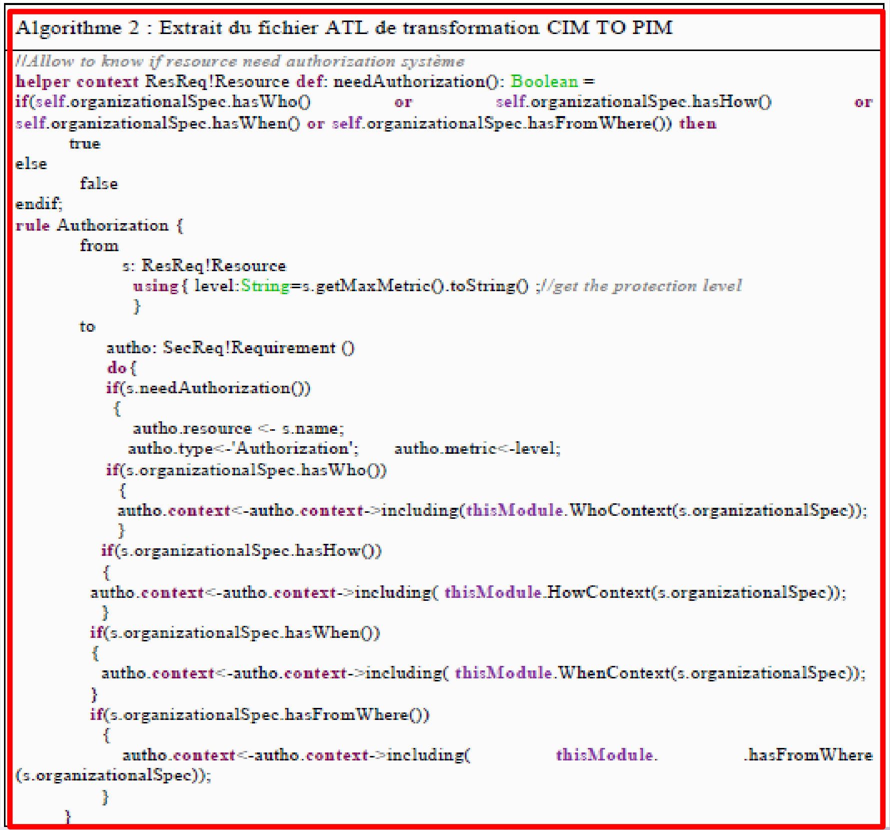
The requirements associated to the resource Rk is :
Reqs(R_k)= { {r} / r є Reqs ʌ r.PR=Rk } (4)

Model-Driven Security approach

CIM To PIM : define security requirement

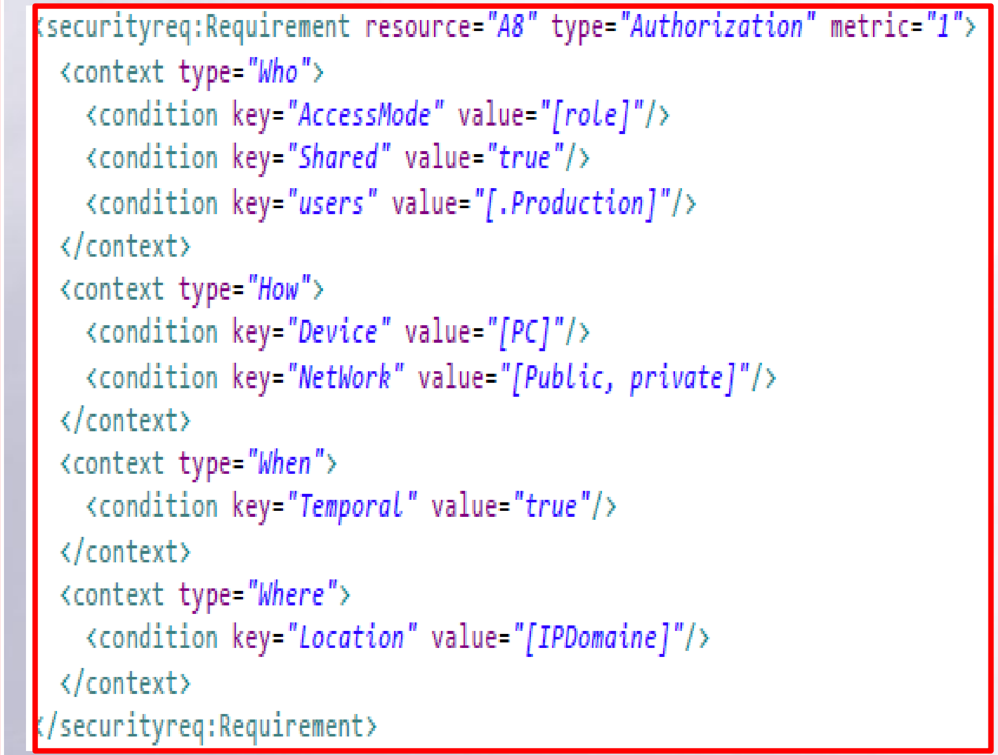


$$M_{CIM2PIM} : \begin{array}{l} RES \longrightarrow REQS \\ (Res_1, Res_i \ldots Res_n) \longrightarrow \mathcal{M}_{CIM2PIM}(Res_1, Res_i \ldots Res_n) \end{array} \quad (8)$$



Algorithme 2 : Extrait du fichier ATL de transformation CIM TO PIM

```
//Allow to know if resource need authorization système
helper context ResReq!Resource def: needAuthorization(): Boolean =
if(self.organizationalSpec.hasWho()          or          self.organizationalSpec.hasHow()          or
self.organizationalSpec.hasWhen() or self.organizationalSpec.hasFromWhere()) then
     true
else
          false
endif;
rule Authorization {
          from
               s: ResReq!Resource
                using{ level:String=s.getMaxMetric().toString() ;//get the protection level
                }
          to
              autho: SecReq!Requirement ()
              do{
              if(s.needAuthorization())
               {
                  autho.resource <- s.name;
                 autho.type<-'Authorization';     autho.metric<-level;
              if(s.organizationalSpec.hasWho())
                {
                autho.context<-autho.context->including(thisModule.WhoContext(s.organizationalSpec));
                }
             if(s.organizationalSpec.hasHow())
              {
           autho.context<-autho.context->including( thisModule.HowContext(s.organizationalSpec));
              }
           if(s.organizationalSpec.hasWhen())
           {
             autho.context<-autho.context->including( thisModule.WhenContext(s.organizationalSpec));
           }
           if(s.organizationalSpec.hasFromWhere())
             {
               autho.context<-autho.context->including(          thisModule.          .hasFromWhere
(s.organizationalSpec));
             }
        }
```



```
<securityreq:Requirement resource="A8" type="Authorization" metric="1">
  <context type="Who">
    <condition key="AccessMode" value="[role]"/>
    <condition key="Shared" value="true"/>
    <condition key="users" value="[.Production]"/>
  </context>
  <context type="How">
    <condition key="Device" value="[PC]"/>
    <condition key="NetWork" value="[Public, private]"/>
  </context>
  <context type="When">
    <condition key="Temporal" value="true"/>
  </context>
  <context type="Where">
    <condition key="Location" value="[IPDomaine]"/>
  </context>
</securityreq:Requirement>
```

Model-Driven Security approach

PIM To PSM : security pattern

Patj= (PatN, PatG, PatTech, {PatR}, {PatM} {PatL}, {PatCtx}, {PatCol}, {PatParm})

-PatN : pattern’s name;

-PatG : pattern’s goal;

-PatType : Abstract or technical pattern

-{PatL} : pattern’s layers;

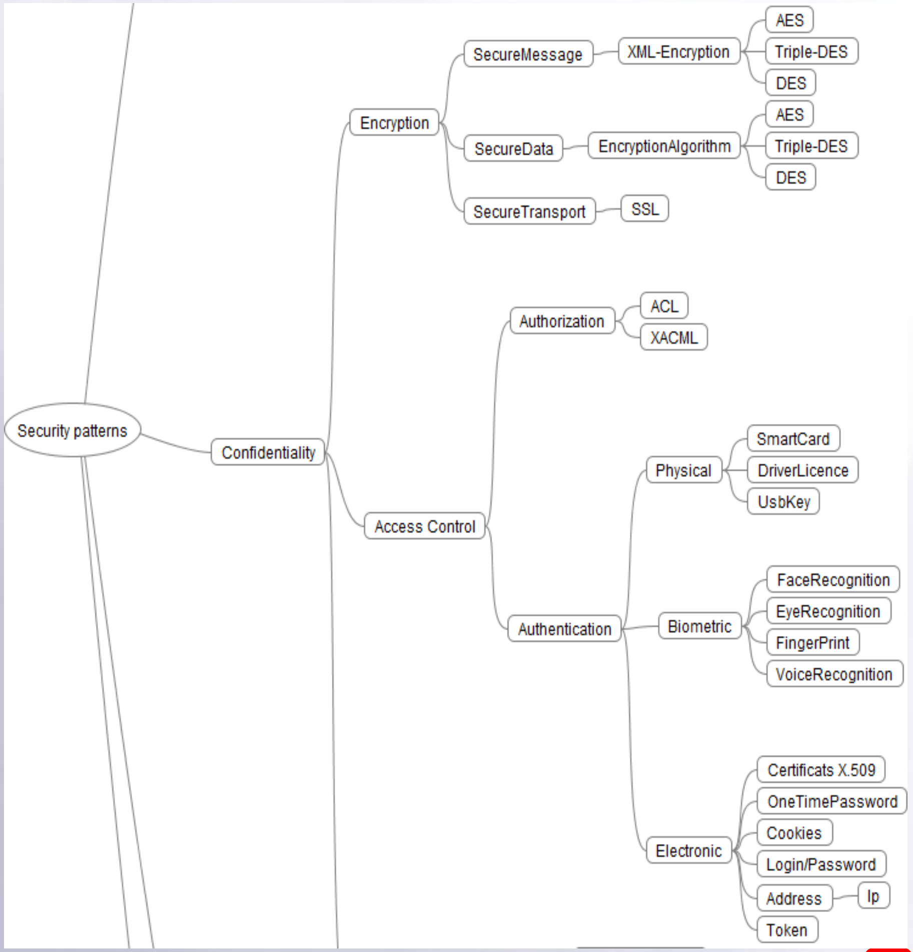
-{PatCtx}: Pattern context (set of conditions and obligations)

-{PatR} : related patterns (sub-patterns);

-{PatM}: set of level of protection offer by the pattern

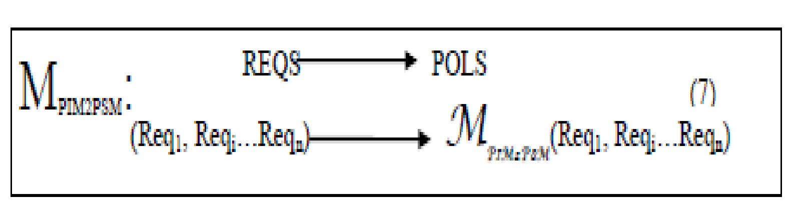
-{PatCol} : pattern collaboration;

-{PatParm} : setting elements;

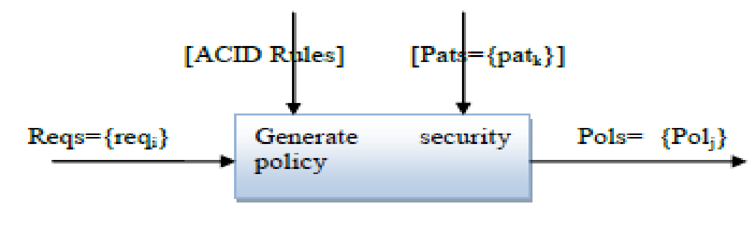


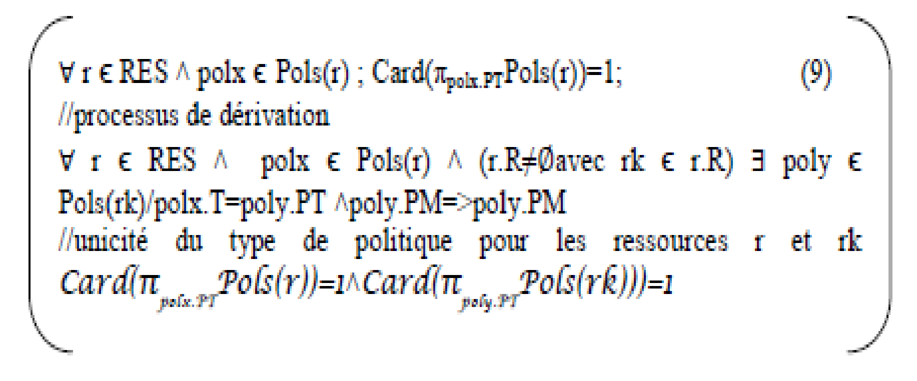
Model-Driven Security approach

PIM To PSM : define abstract policy



$$M_{PIM2PSM}: \begin{array}{l} REQS \longrightarrow POLS \\ (Req_1, Req_j \ldots Req_n) \longrightarrow \mathcal{M}_{PIM2PSM}(Req_1, Req_j \ldots Req_n) \end{array} \quad (7)$$





$\forall$ r $\in$ RES $\wedge$ polx $\in$ Pols(r) ; Card($\pi_{polx.PT}$Pols(r))=1; (9)

//processus de dérivation

$\forall$ r $\in$ RES $\wedge$ polx $\in$ Pols(r) $\wedge$ (r.R$\neq\emptyset$avec rk $\in$ r.R) $\exists$ poly $\in$ Pols(rk)/polx.T=poly.PT $\wedge$poly.PM=>poly.PM

//unicité du type de politique pour les ressources r et rk

$Card(\pi_{polx.PT}Pols(r))=1 \wedge Card(\pi_{poly.PT}Pols(rk)))=1$

Polx= (PR, PT, PG, PL,PM, {PC}, PP)
- PR : policy resource;
- PT : Policy type
- PG : Policy goal
- PL : the layer of this policy
- PM : the metric of this policy
- {PR} : the policy rules
- PP : the pattern to use

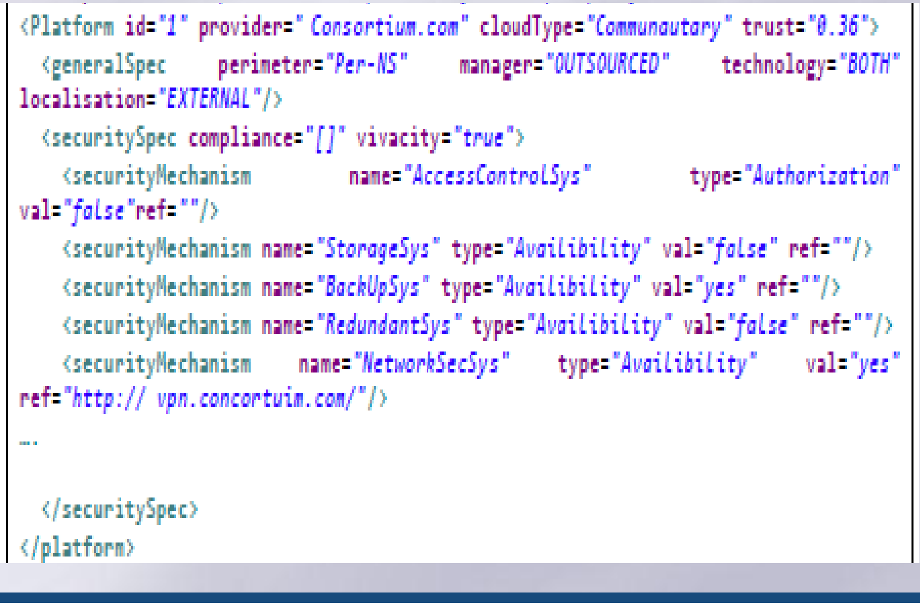
All the policy rules of all resources as:
Pols= {Polj} where 0<j<Nj;(6)

The policies rules associated to the resource Rk is :
Pols(Rk) = {{p} / pє Pols ʌ p.PR=Rk) ; (7)

Model-Driven Security approach

PDM specification

Plat= (PlaN, PlaT, PlaTst,{PlatSM})
- PlaN : platform provider;
- PlaT : platform type (public, comminatory, private,..)
PlatTst: the level of client Trust to the platform
{PlaSM} : Security mechanisms implemented



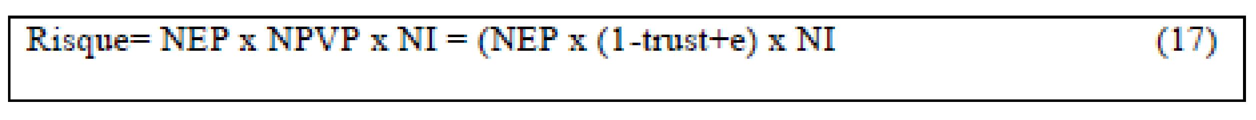
```
<Platform id="1" provider=" Consortium.com" cloudType="Communautary" trust="0.36">
  <generalSpec    perimeter="Per-NS"    manager="OUTSOURCED"    technology="BOTH"
localisation="EXTERNAL"/>
  <securitySpec compliance="[]" vivacity="true">
    <securityMechanism          name="AccessControlSys"          type="Authorization"
val="false"ref=""/>
    <securityMechanism name="StorageSys" type="Availibility" val="false" ref=""/>
    <securityMechanism name="BackUpSys" type="Availibility" val="yes" ref=""/>
    <securityMechanism name="RedundantSys" type="Availibility" val="false" ref=""/>
    <securityMechanism    name="NetworkSecSys"    type="Availibility"    val="yes"
ref="http:// vpn.concortuim.com/"/>
...


  </securitySpec>
</platform>
```

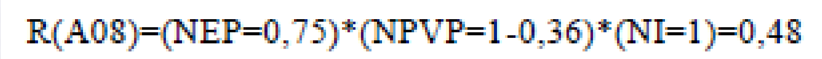
Questions	Answers
Deployment platform specification	
Who manages the Cloud infrastructure? You (the company) or the service provider?	The service provider
Where are data stored? Inside your company boundaries or outside.	Outside
Who owns the data? You (The company) or service provider?	The compagny
Is Cloud infrastructure shared to another's companies?	yes
Do infrastructure provides backup and versioning systems to restore the system in case of an incident?	No
Does Infrastructure provide services and protocols to secure communications (VPN, HTTPS, ...)?	Yes
Does Infrastructure provide security services and APIs to control access to business services and data?	No
Does infrastructure is certified (ISO 27001 certification, SAS 07, FISMA,)?	No
……	

Model-Driven Security approach

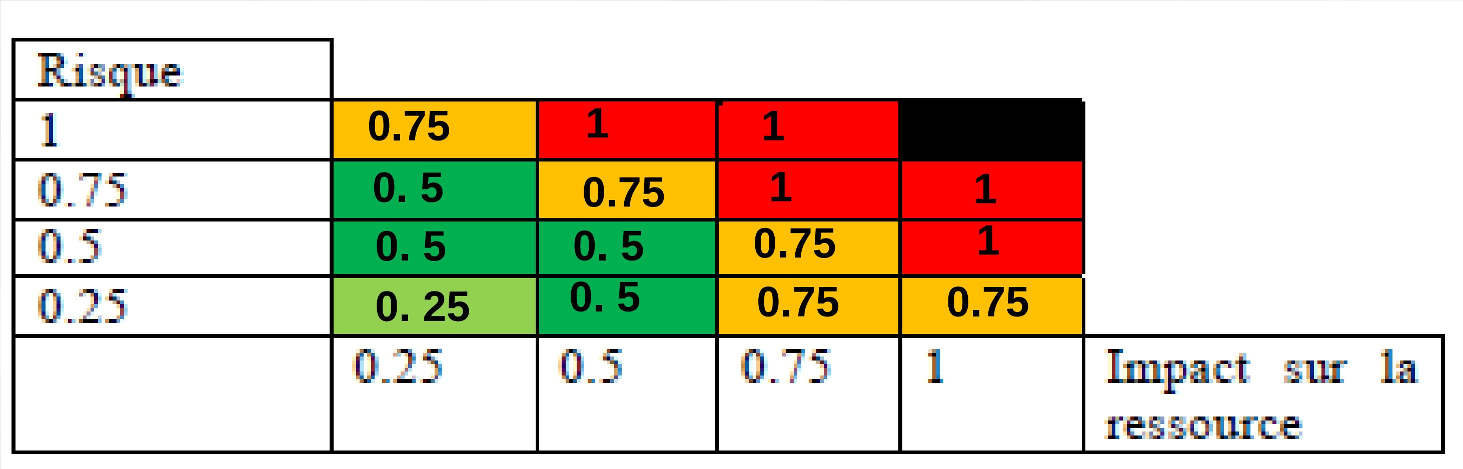
PSM To PSM : risk analysis and assessment



Risque= NEP x NPVP x NI = (NEP x (1-trust+e) x NI (17)



R(A08)=(NEP=0,75)*(NPVP=1-0,36)*(NI=1)=0,48



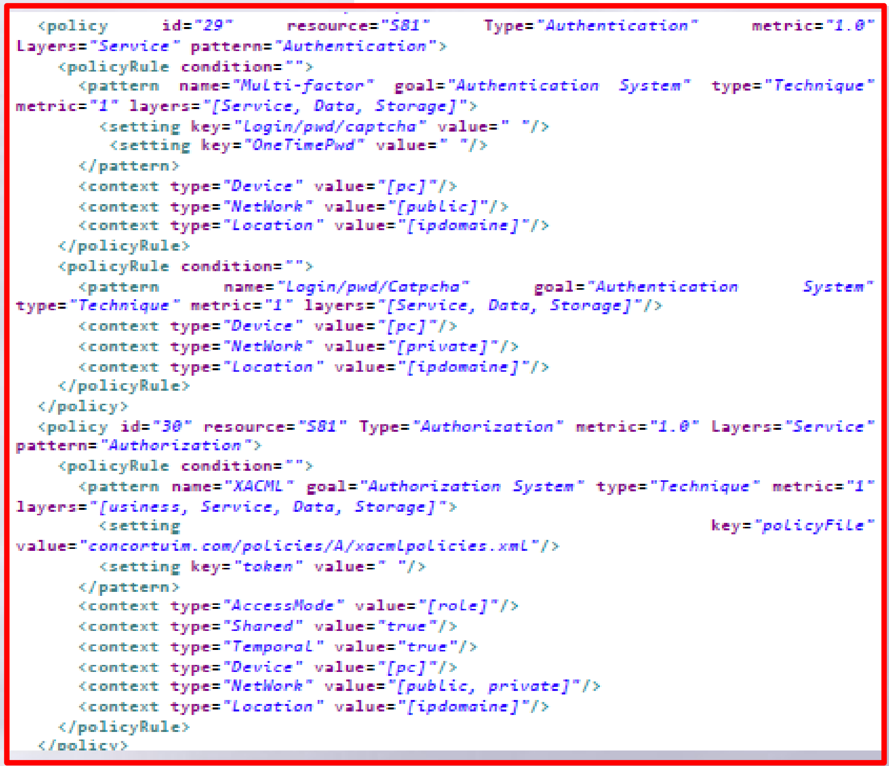
Risque					
1	0.75	1	1	■	
0.75	0. 5	0.75	1	1	
0.5	0. 5	0. 5	0.75	1	
0.25	0. 25	0. 5	0.75	0.75	
	0.25	0.5	0.75	1	Impact sur la ressource

Protection level assessment grid

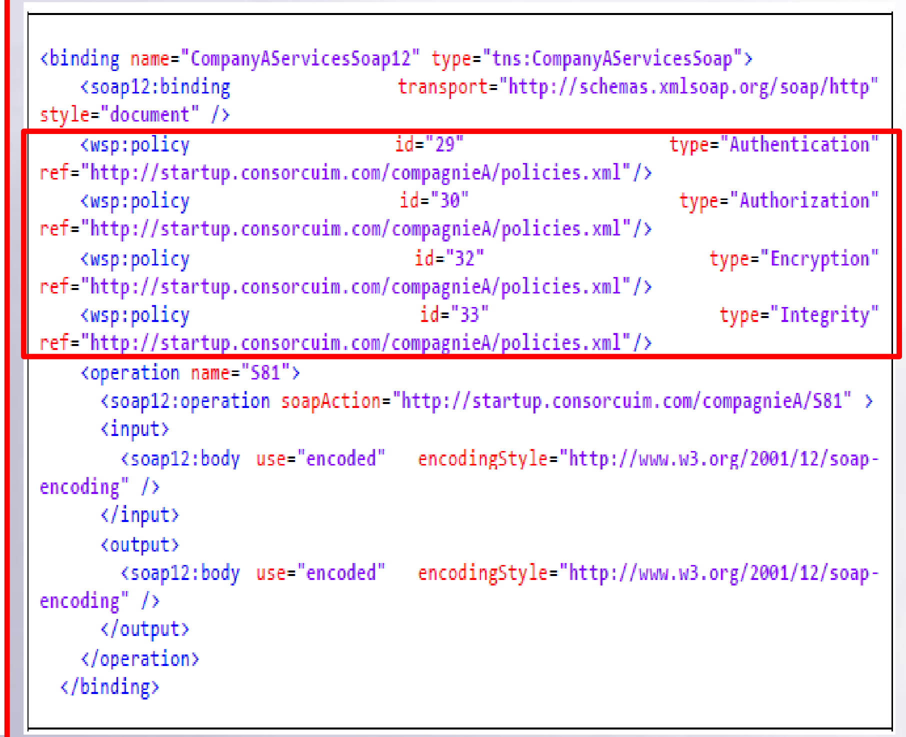
Model-Driven Security approach

PSM To PSM : Security policy generation

$$M_{PSM2PSM}: \begin{array}{l} POLS \longrightarrow POLS \\ (Pol_1, Pol_i \ldots Pol_n) \longrightarrow \mathcal{M}_{PSM2PSM}(Pol_1, Pol_i \ldots Pol_n) \end{array} \qquad (12)$$



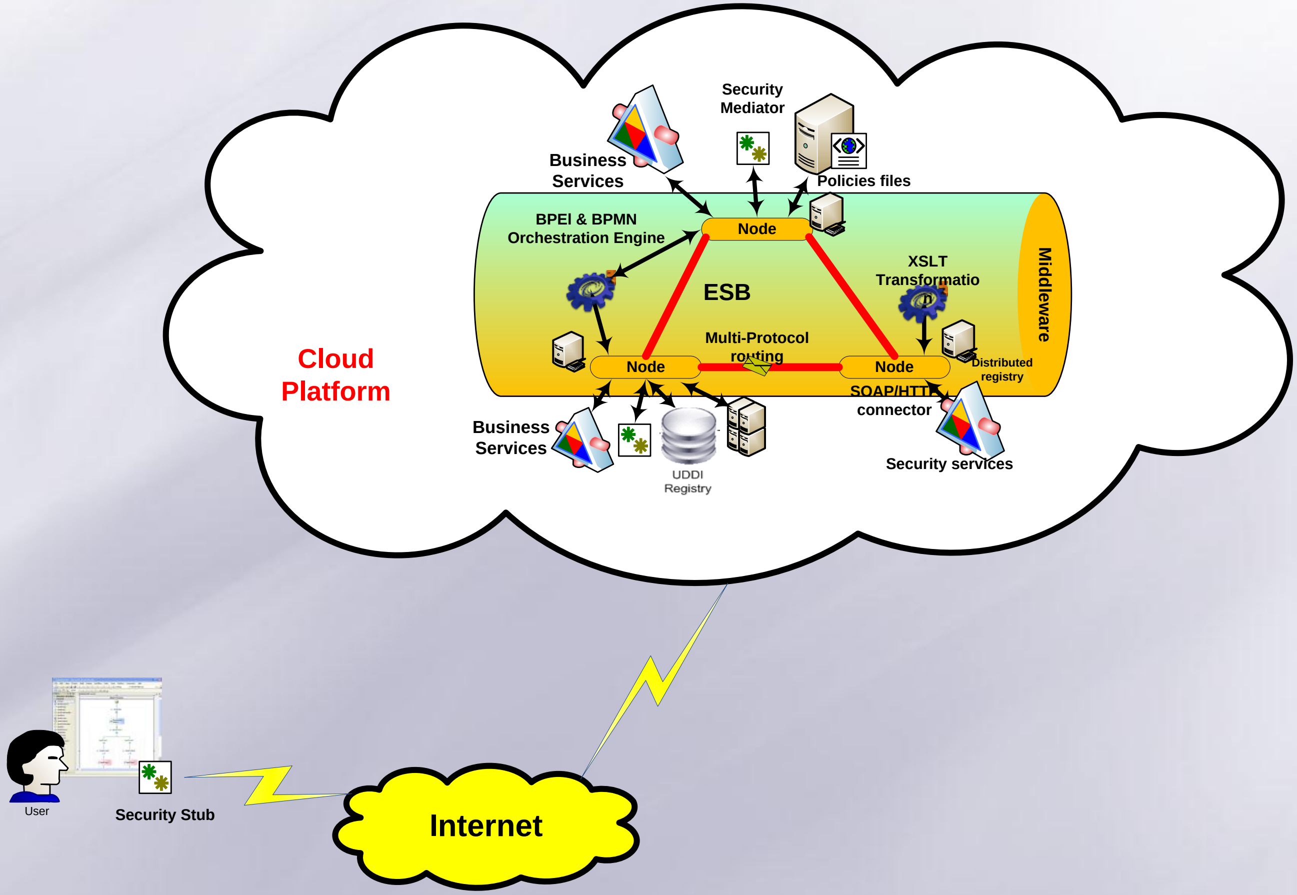
```
  <policy      id="29"      resource="S81"      Type="Authentication"      metric="1.0"
Layers="Service" pattern="Authentication">
    <policyRule condition="">
      <pattern  name="Multi-factor"  goal="Authentication  System"  type="Technique"
metric="1" layers="[Service, Data, Storage]">
        <setting key="Login/pwd/captcha" value=" "/>
         <setting key="OneTimePwd" value=" "/>
      </pattern>
      <context type="Device" value="[pc]"/>
      <context type="NetWork" value="[public]"/>
      <context type="Location" value="[ipdomaine]"/>
    </policyRule>
    <policyRule condition="">
      <pattern      name="Login/pwd/Catpcha"      goal="Authentication      System"
type="Technique" metric="1" layers="[Service, Data, Storage]"/>
      <context type="Device" value="[pc]"/>
      <context type="NetWork" value="[private]"/>
      <context type="Location" value="[ipdomaine]"/>
    </policyRule>
  </policy>
  <policy id="30" resource="S81" Type="Authorization" metric="1.0" Layers="Service"
pattern="Authorization">
    <policyRule condition="">
      <pattern name="XACML" goal="Authorization System" type="Technique" metric="1"
layers="[usiness, Service, Data, Storage]">
        <setting                                                   key="policyFile"
value="concortuim.com/policies/A/xacmlpolicies.xml"/>
        <setting key="token" value=" "/>
      </pattern>
      <context type="AccessMode" value="[role]"/>
      <context type="Shared" value="true"/>
      <context type="Temporal" value="true"/>
      <context type="Device" value="[pc]"/>
      <context type="NetWork" value="[public, private]"/>
      <context type="Location" value="[ipdomaine]"/>
    </policyRule>
  </policy>
```



```
<binding name="CompanyAServicesSoap12" type="tns:CompanyAServicesSoap">
    <soap12:binding              transport="http://schemas.xmlsoap.org/soap/http"
style="document" />
    <wsp:policy                 id="29"                   type="Authentication"
ref="http://startup.consorcuim.com/compagnieA/policies.xml"/>
    <wsp:policy                 id="30"                    type="Authorization"
ref="http://startup.consorcuim.com/compagnieA/policies.xml"/>
    <wsp:policy                  id="32"                      type="Encryption"
ref="http://startup.consorcuim.com/compagnieA/policies.xml"/>
    <wsp:policy                   id="33"                      type="Integrity"
ref="http://startup.consorcuim.com/compagnieA/policies.xml"/>
    <operation name="S81">
      <soap12:operation soapAction="http://startup.consorcuim.com/compagnieA/S81" >
      <input>
        <soap12:body  use="encoded"   encodingStyle="http://www.w3.org/2001/12/soap-
encoding" />
      </input>
      <output>
        <soap12:body  use="encoded"   encodingStyle="http://www.w3.org/2001/12/soap-
encoding" />
      </output>
    </operation>
  </binding>
```

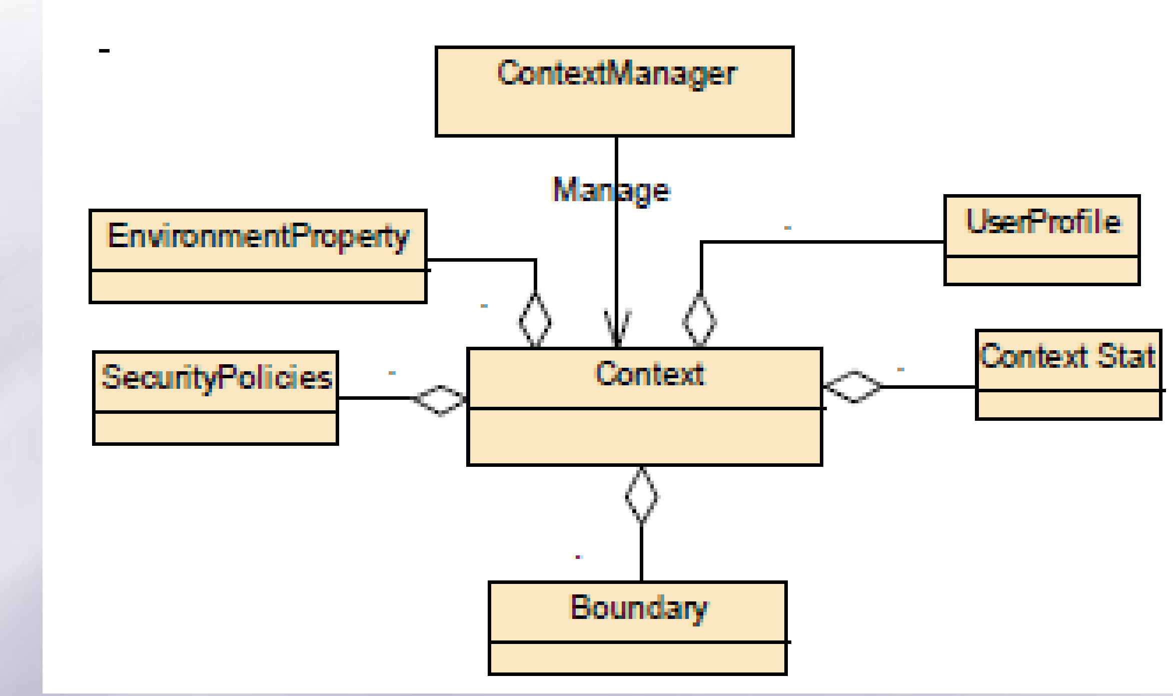
Model-Driven Security approach

Model@Runtime : Security architecture



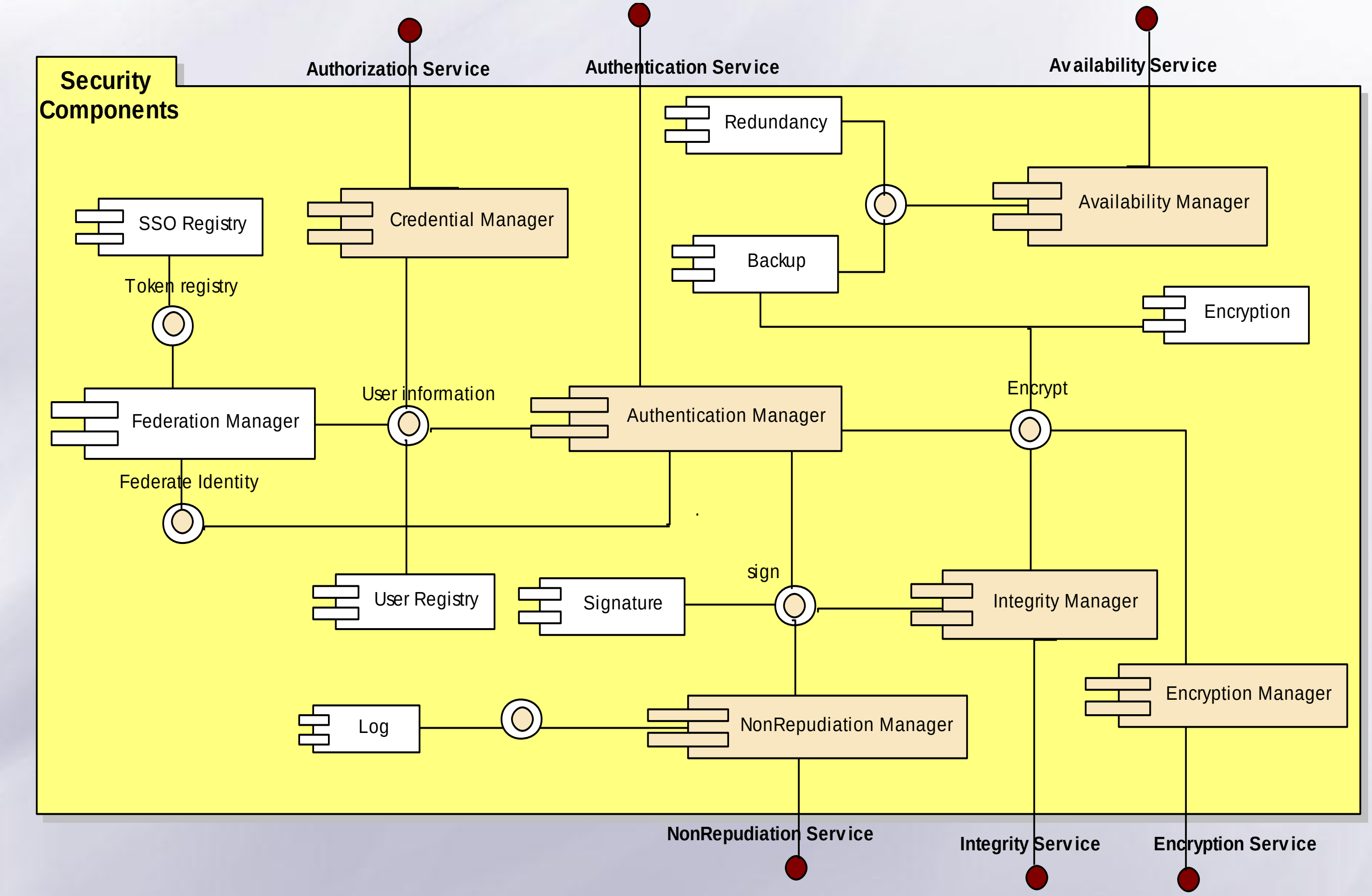
Model-Driven Security approach

Model@Runtime : execution context



Model-Driven Security approach

Security components implemented as SecaaS



Conclusion

Framework Evalute criteria	Ours solution (Secure BP)	OpenPMF [6]	SECTET [7]	BP Sec [4]	KIT secure BP[5]
Abstractions levels	CIM- PIM- PSM-code	PIM-PSM-code	PIM-PSM-Code	CIM-UML Use case (PIM)	PIM-PSM
Approach used	Annotation based + UML	UML	Annotation based+ UML	UML	Annotation
Oriented end user	Yes	No	No	Yes	No
Automatic Policy generation	Yes	Yes	Yes		Yes
Modelisation language and transformation	EMF+ATL+ Ad-hoc transformation	UML+DSL	UML2+SECTET-DSL	UML +QVT	Ad-hoc
Take account infrastructure	Yes	No	No	No	No
Take account execution context	Yes	No	No	No	Yes
Security criteria	Authentication, Authorization, Integrity, Encryption, Non-Repudiation, Availability, Privacy	Authentication, Authorization, Monitoring	Encryption, Intégrité, Non-répudiation, Authentication	Non-Repudiation, Privacy, intrusion Détection, Access control, Authorization	Authorization
Policy monitoring	No	Yes	No	Yes	No
SecaaS (security as a Service)	Yes	No	Yes	No	No
Security Standard	XACML, SAML, WS-Security	XACML	SAML, WS-policy, XACML		XACML

Conclusion and further works

Conclusion

- Use model driven approach to :
  - Identity, for each enterprise, their business process security requirements.
  - Define an adapted Quality of Protection
  - Generate contextual security policies
- Define security architecture to take account the execution context
- Define standardized security mechanisms as SecaaS which are invoked according to the runtime context and allow end to end security

Further works

- Extend security pattern for privacy
- Monitoring security policies

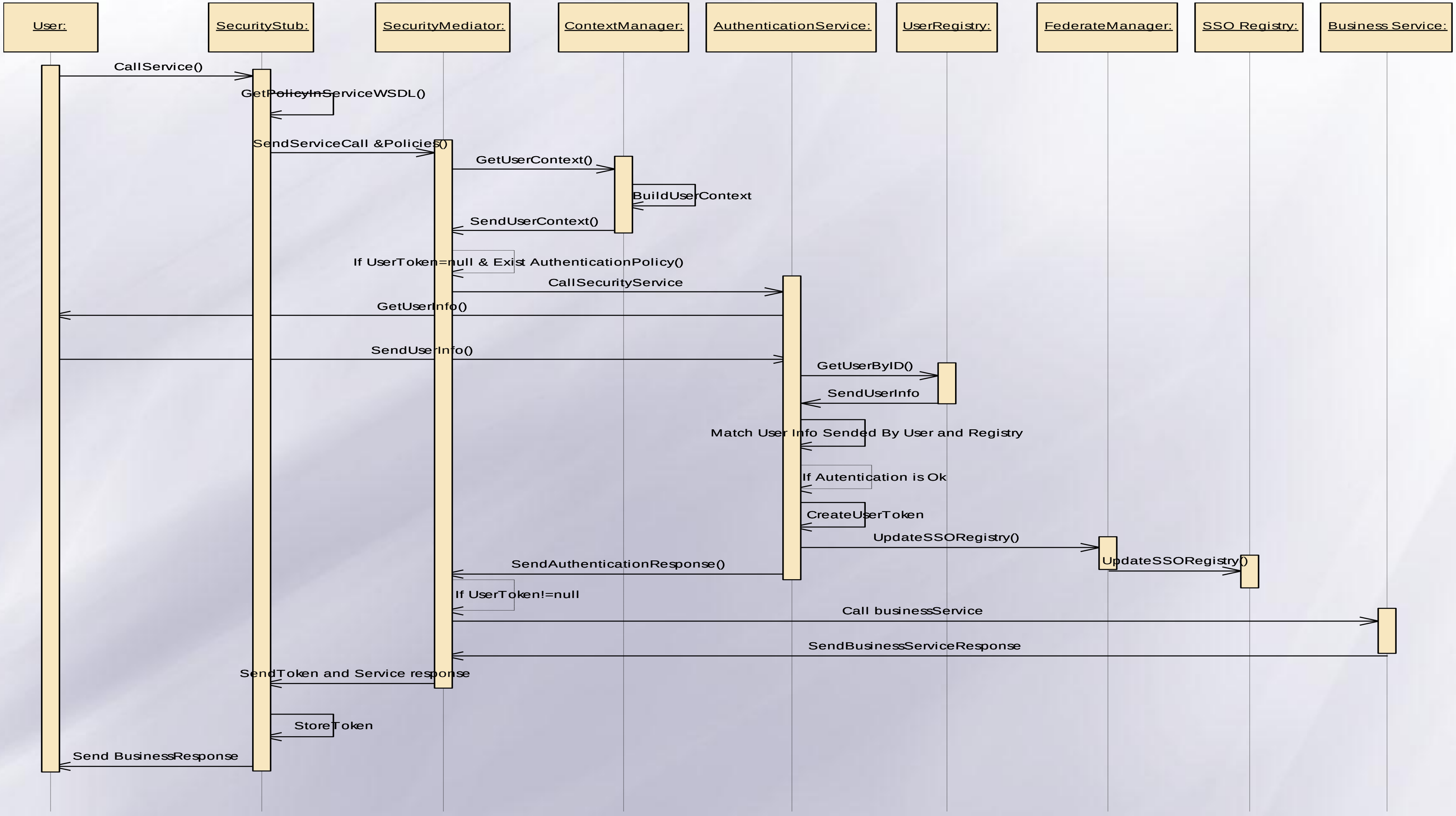
References

- **[1] Organization for the Advancement of Structured Information Standards (OASIS)**: “Reference Model for Service Oriented Architecture 1.0: OASIS Standard”, 12 October 2006.
- **[2] Jericho Forum**, “Cloud Cube Model: Selecting Cloud Formations for Secure Collaboration”, Version 1.0, (April 2009), http://www.opengroup.org/jericho/cloud_cube_model_v1.0.pdf.
- **[3] NIST**, Cloud Computing Standards Roadmap – Version 1.0, NIST National Institute of Standards and Technology Special Publication 500-291, 2011
- **[4] Rodriguez A., Fernandez-Medina E., Piattini M.**, “A BPMN extension for the modeling of security requirements in business processes”, the institute of electronics, Information and Communication Engineers (IEICE), Vol.E90-D, NO.4. 2007
- **[5] Mülle J, von Stackelberg S, Klemen A** Security Language for BPMN Process Models, Karlsruhe institute of technology, Germany, 2011.
- **[6] Objectsecurity**, OpenPMF, Model Driven Security Policy Automation, http://www.objectsecurity.com/doc/poster.pdf .
- **[7]SECTET Framwork**, http://www.sectet.org
- **[8] Ouedraogo, W. F., Biennier, F., Ghodous, P.**, “Adaptive Security Policy Model to Deploy Business Process in Cloud Infrastructure”, The 2nd International Conference on Cloud Computing and Services Science, CLOSER 2012, p.287-290, April 2012

Thank you for your attention

Model-Driven Security approach

Authentication sequence diagram

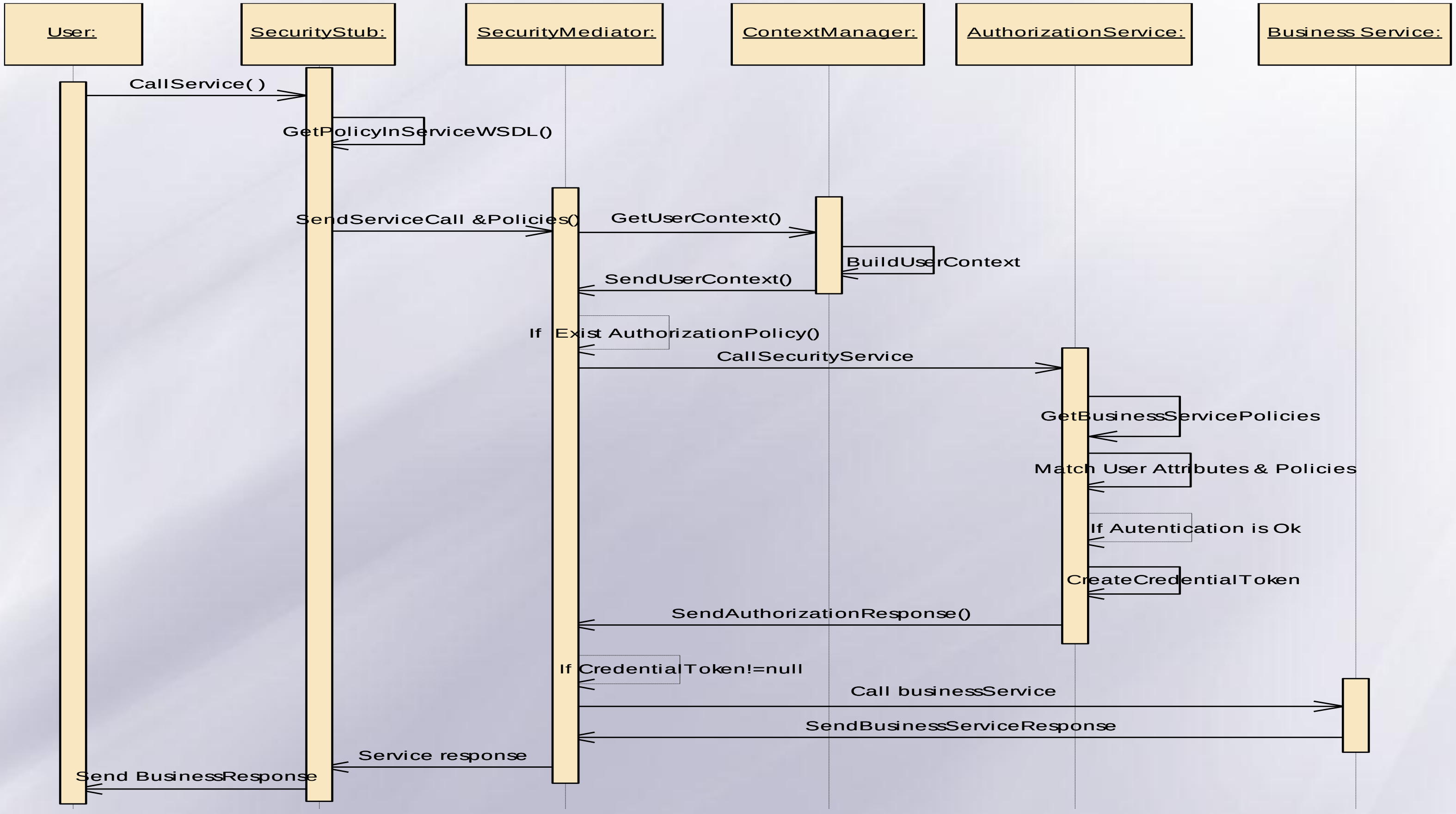


User:
SecurityStub:
SecurityMediator:
ContextManager:
AuthenticationService:
UserRegistry:
FederateManager:
SSO Registry:
Business Service:

CallService()
GetPolicyInServiceWSDL()
SendServiceCall &Policies()
GetUserContext()
BuildUserContext
SendUserContext()
If UserToken=null & Exist AuthenticationPolicy()
CallSecurityService
GetUserInfo()
SendUserInfo()
GetUserByID()
SendUserInfo
Match User Info Sended By User and Registry
If Autentication is Ok
CreateUserToken
UpdateSSORegistry()
UpdateSSORegistry()
SendAuthenticationResponse()
If UserToken!=null
Call businessService
SendBusinessServiceResponse
SendToken and Service response
StoreToken
Send BusinessResponse

Model-Driven Security approach

Authorization sequence diagram



User:

SecurityStub:

SecurityMediator:

ContextManager:

AuthorizationService:

Business Service:

CallService( )

GetPolicyInServiceWSDL()

SendServiceCall &Policies()

GetUserContext()

BuildUserContext

SendUserContext()

If Exist AuthorizationPolicy()

CallSecurityService

GetBusinessServicePolicies

Match User Attributes & Policies

If Autentication is Ok

CreateCredentialToken

SendAuthorizationResponse()

If CredentialToken!=null

Call businessService

SendBusinessServiceResponse

Service response

Send BusinessResponse