Workshop Security

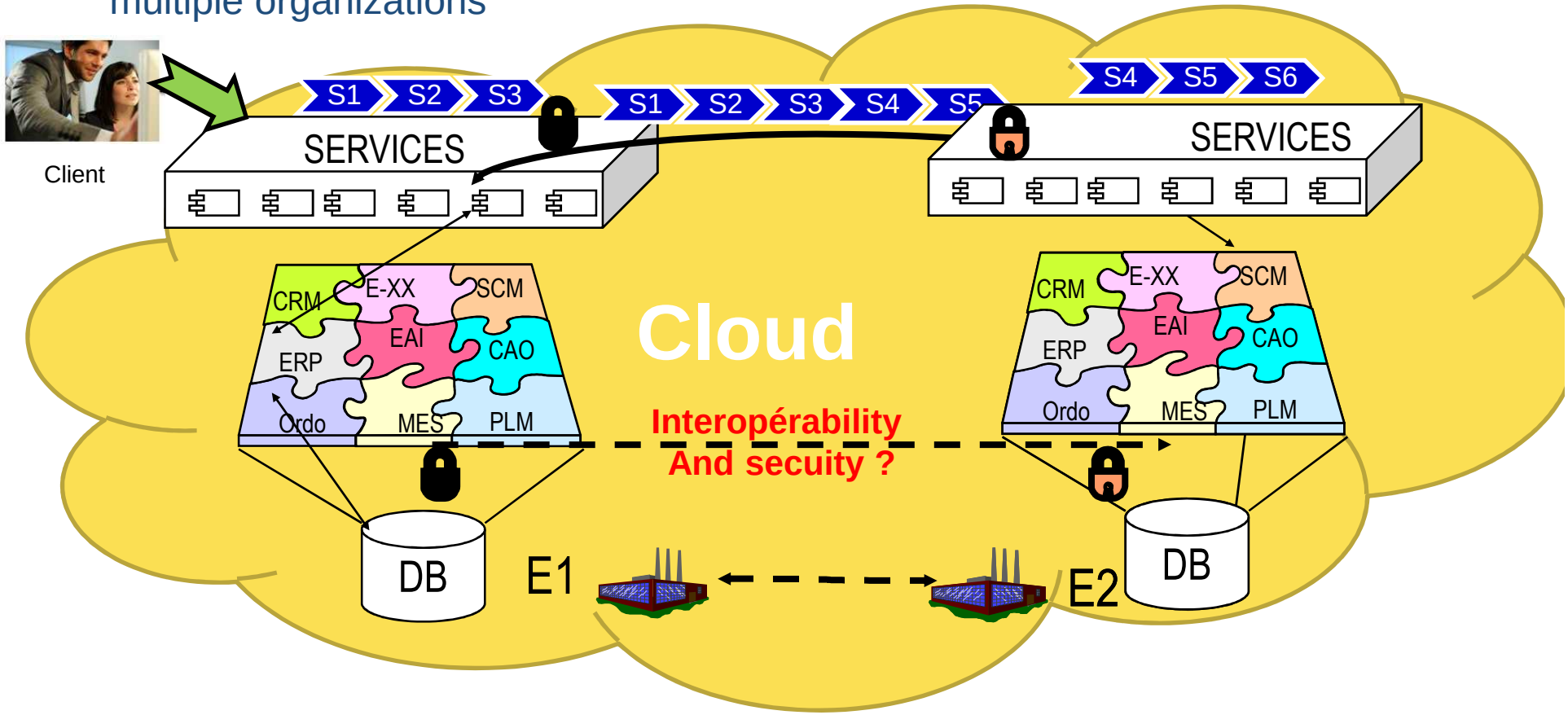
Integration of DRM in Service Systems

Authors:
Ziyi Su, W. Francis Ouédraogo & Frederique Biennier

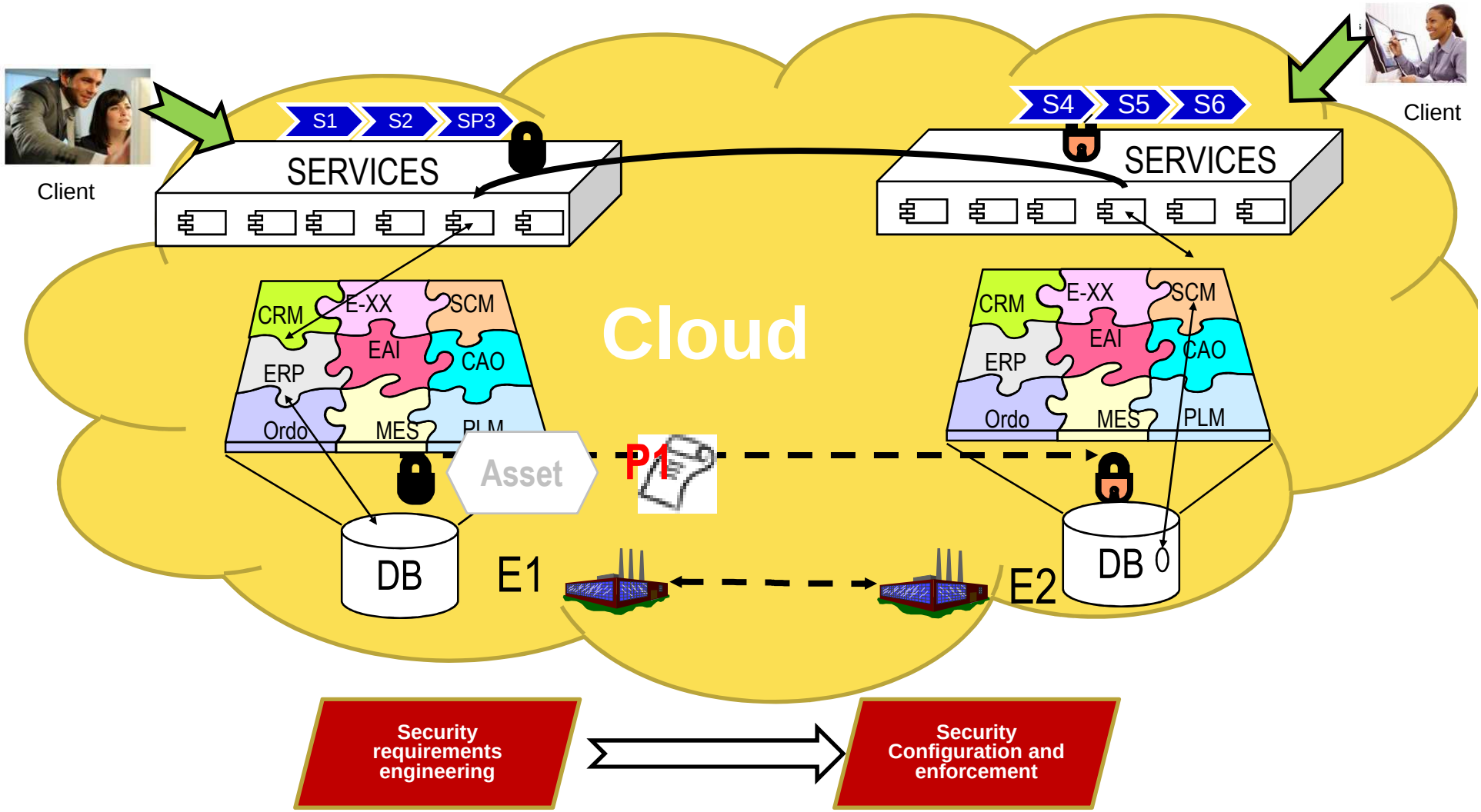
Université de Lyon, CNRS INSA-Lyon. LIRIS. UMR5205. F-69621. France
North East Normal University - China

Problematic

- Collaborative business process relies mostly on software services spanning multiple organizations



Problematic



Requirements

Intra-organizational security factors

(1)Basic criteria(“instant” protection):
Confidentiality, integrity, availability

(2) ‘Best practice’:
ISO/IEC 17799, ISO/IEC27002; OCTAVE; EBIOS and SNA

(3) Security Layer:
Physical (hardware) security,
system (OS and platforme related software) security,
Application & data security,
communication (network) security and
human aspect (organizational factors).

Requirements

- **Inter-organizational security**

  (1) **Trust assessment**

  Direct-trust, Reputation

  (2**) Refined trust**

  Access control: Identify-based (MAC,DAC), Role-based (RBAC),Attribute-based (XACML)

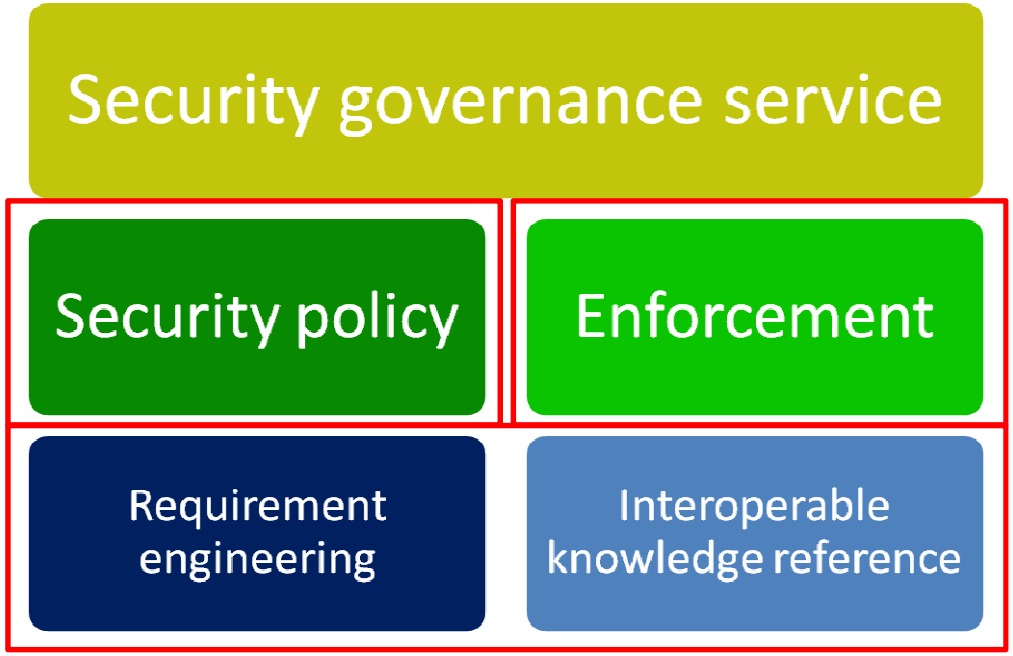
  DRM: continuous usage session, usage & management actions

  (3) Monitoring

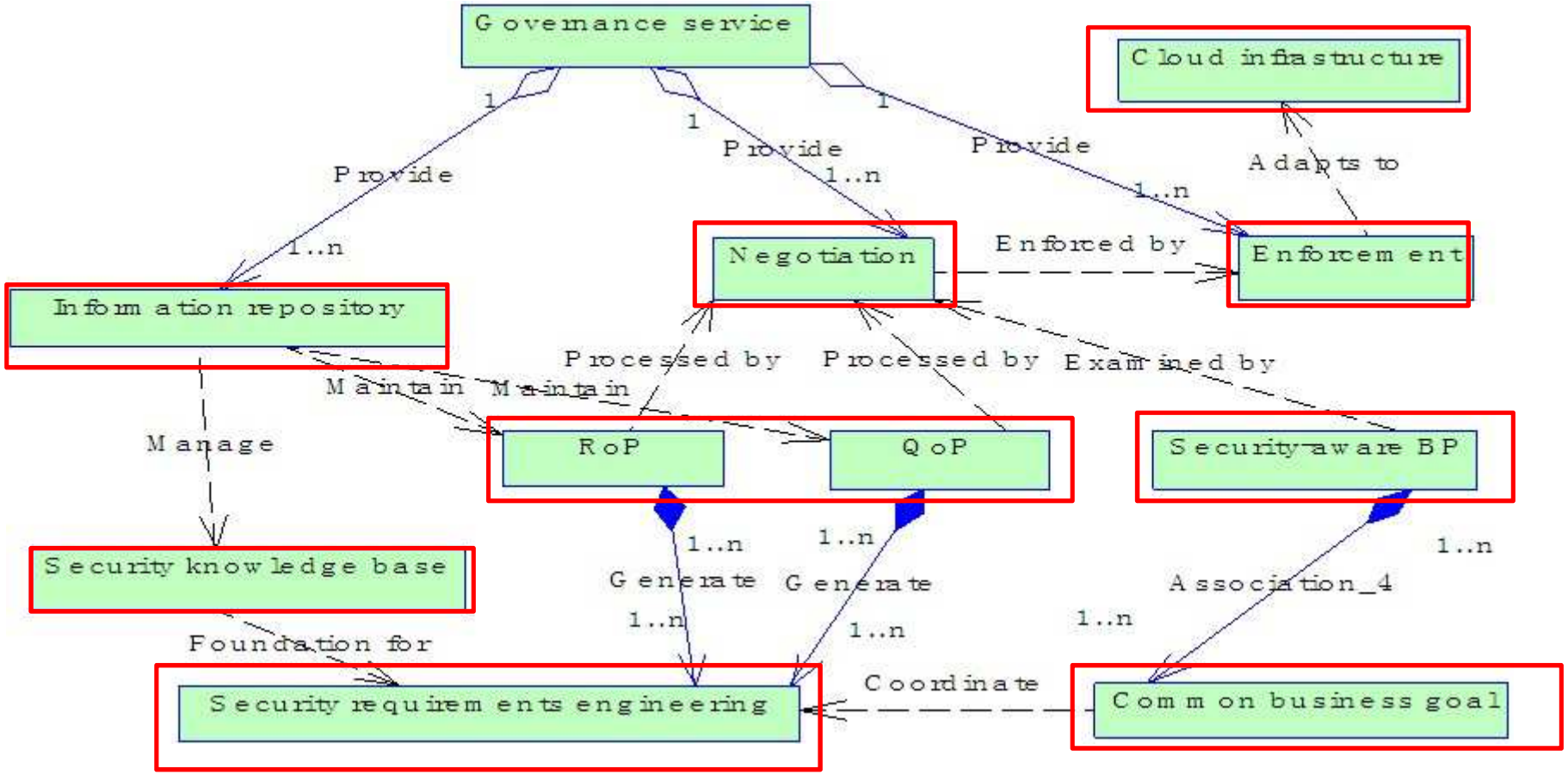
  message intercepting, agent:

  prohibition, modification, observation

Framework Overview



Framework in-depth view



Security requirements engineering

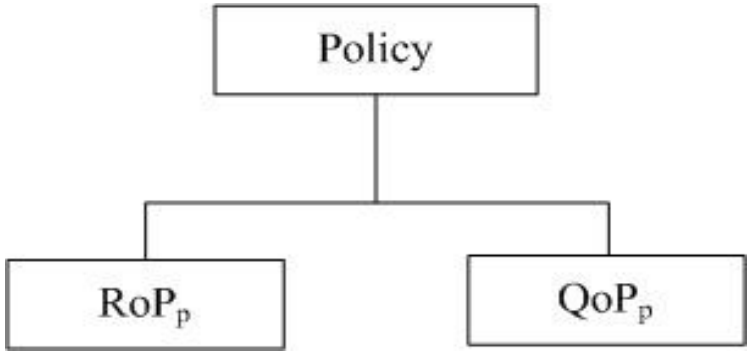
Risk analysis methods	Requirements analysis	Design	Implementation
EBIOS	Text risk and objectives identifications	Protection pattern	
OCTAVE	Structured information access identification	Objectives prioritization Best practices	Audit and implementation project management
SNA	Process and resources workflow identification	“Survival process” design	CERT attacks information and knowledge base
MEHARI	Shortened risk analysis	Best practices	Implementation project management

Integration of DRM in Service Systems

•Security goal	•Questions	•Answers
	IS & assets questions	
•-	•Which functionalities & assets?	•List of information assets and functionalities
•CIAN	•Which security goal on these functionalities & assets?	•CIAN
•CIAN	•Which security/assurance mechanisms on these functionalities & assets?	•Hardware/OS/platform/net-work/application/human level mechanisms
	Openness & assets sharing questions	
•CIAN	•Which functionalities & assets are shared?	•List of information assets and functionalities
•N	•Shared with which partners?	•'pre-difined'/ random
	Risks & compensation questions	
•CIAN	•Which security/assurance mechanisms negatively affected by the openness?	•List of mechanisms
•CIAN	•Which level the negative effects have achieve?	•Neutralize/damage/ineffect at times
•CIAN	•Which level of compensentation you want to have?	•Total restore/partial restore
•CIAN	•Which security level should be achieved after the compensentation?	•C/I/A/N
•CIAN	•Should these security level be maintained by partners or collaboration system?	•Partner/system
•-	•Any other requirements on partners?	•-
•-	•Any other requirements for the collaboration system?	•-
•Legend:	***•C (Confidentiality), I (integrity), A (Availability), N (Non repudiation)***	

Policy Model

- Requirements of Protection (**RoP**)
  & Quality of Protection **(QoP)**



Asset Provider (AP) uses **RoP** to specify its requirements
Asset Consumer(AC) uses **QoP** to declare its qualification for access rights.

A Participant (service provider or consumer) may have both RoP and QoP.

Security Policy Model

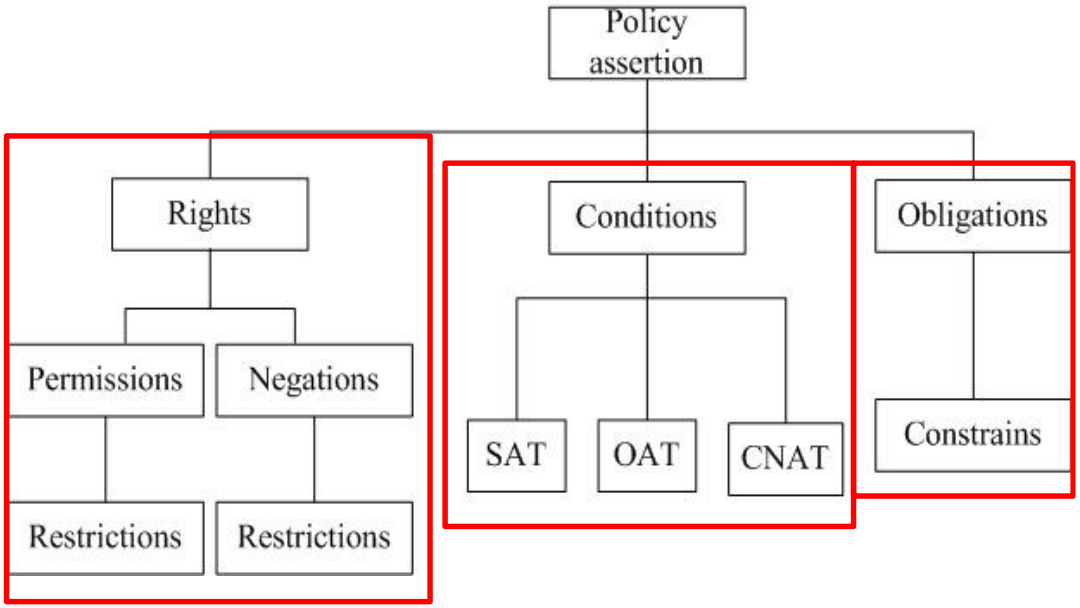
- **Policy Assertion:** Refined Access control

Policy Assertion:

Rights:
- Permissions
- Negations

Conditions:
- Subject Attributes (SAT)
- Object Attributes (OAT)
- Context Attributes (CNAT)

Obligations

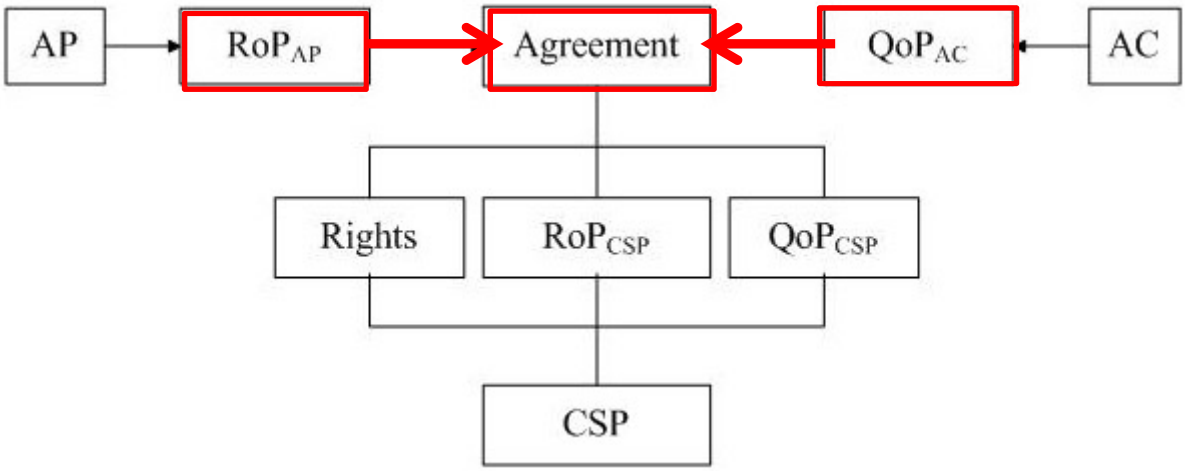


Sample policy

$$Rt(read(O,S)) \wedge Ob(delete(O,S,with(30days))) \leftarrow Sh(x=100) \wedge OAT(ID=M) \wedge SAT(certify(S,A) \wedge contract(S,B)) \wedge CNAT(deliveryChannel = "SSL")$$

Policy Model

- Collaboration Security Policy (CSP) management

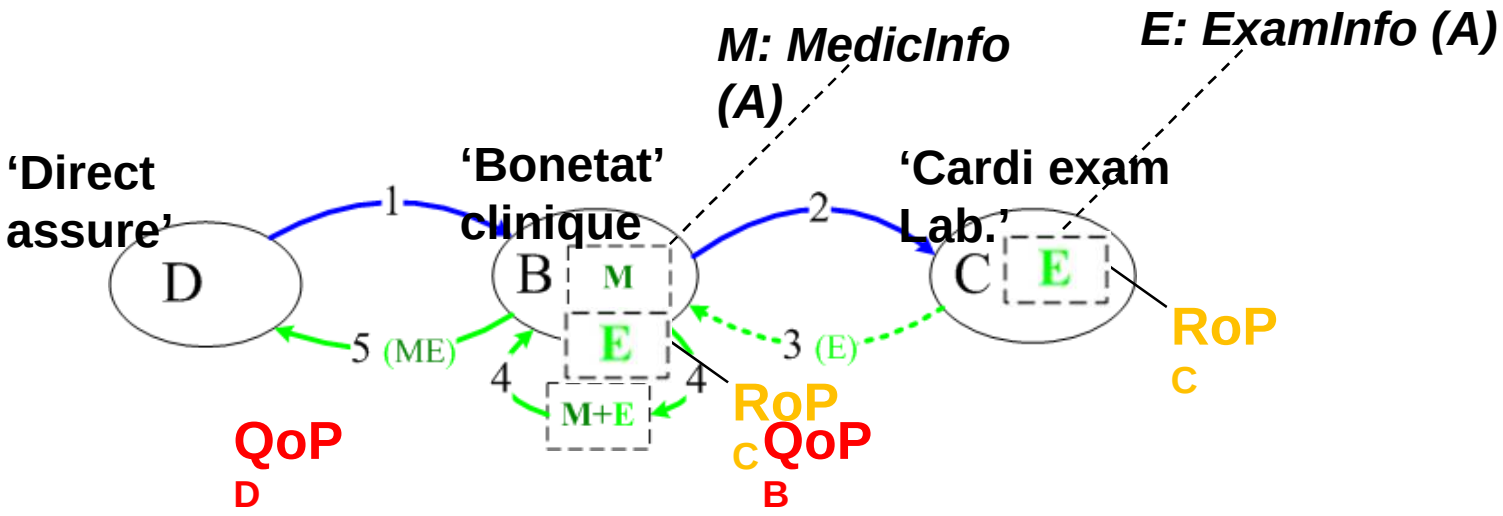


CSP includes QoP_{AC} and RoP_{AP}
Aggregation => $QOP_{AC} > RoP_{AP}$

$RoP_{CSP} = RoP_{CSP} + RoP_{AP}$

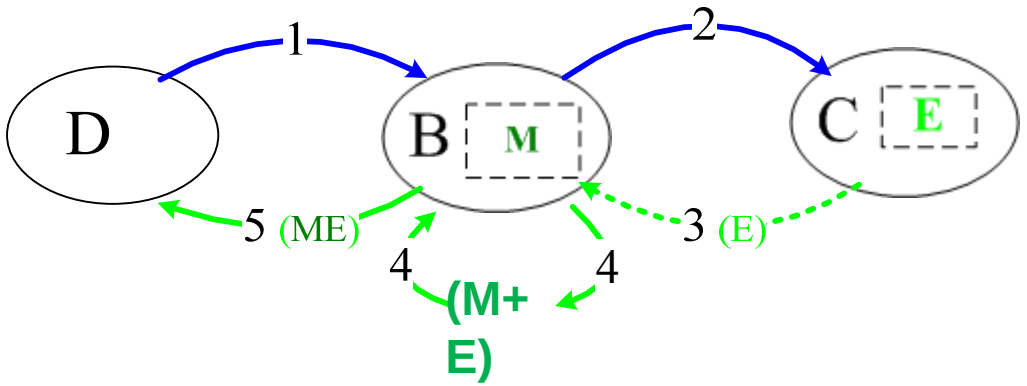
$QoP_{CSP} = QoP_{CSP} + QoP_{AP} + QoP_{AC}$

- Challenge: From stand alone requirements to contextual ones
  - How can we compute if a right can be granted depending on the service workflow?
    - Let policies to be associated to assets,
    - When an asset is disseminated, its policy must follow
    - In the following sample use case, RoP_C follows 'E' to examine QoP_B and QoP_D
    - How can we use the service composition to define the policy composition?



Service call graph model

- Service Call Graph(SCG)
  - Control dependency
    - Service calls
  - Data dependency
    - Data exchange
- 'Service call tuple' list
  - Represents SCG
  - Can be scaned, to analyze as
- This list can be scanned
  - To track assets derivations,which
  - Context '**slicing**'



$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftrightarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

- ‘Asset based’ slicing
  - Step 3: first asset ‘E’, first s used to evaluate ‘QoPB’

$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftrightarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

QoPB

<‘Rc’,1, (Φ), (E), (RoPc), step_3>

- ‘Asset based’ slicing
  - Step 4: ‘M’ merged with ‘E’ version ‘2’, two policies ‘R(aggregated

$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftrightarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

<‘Rc’,1, (Φ), (E), (RoPc), step_3>

<‘Rc’, 2, (‘Rc.1’),(E,M) (RoPc, RoPB),step_4>

,

- ‘Asset based’ slicing
  - Step 5: Assets & policies re
  - aggregated ‘RoPC,RoPB’ s QoPD.

$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftrightarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

<‘Rc’,1, (Φ), (E), (RoPc), step_3>

<‘Rc’, 2, (‘Rc.1’),(E,M) (RoPc, RoPB),step_4>

<‘Rc’, 3, (‘Rc.2’), (E,M), (RoPc, RoPB), step_5>

QoPD

- Problem
  - Untill ‘step 3’, we can not s
  - If there is conflict, step 1 ar and B.

$$< step\_1, D \xrightarrow{c} B >$$
$$< step\_2, B \xrightarrow{c} C >$$
$$< step\_3, B \xleftarrow{d} C,' E' >$$
$$< step\_4, B \xleftrightarrow{d} B,' E',' ME' >$$
$$< step\_5, D \xleftarrow{d} B,' ME' >$$

<‘Rc’,1, (Φ), (E), (RoPc), step_3>

<‘Rc’, 2, (‘Rc.1’),(E,M) (RoPc, RoPB),step_4>

<‘Rc’, 3, (‘Rc.2’), (E,M), (RoPc, RoPB), step_5>

- ‘Request based’ slicing
  - Step 1: first consumer ‘D’, a
    ‘QoPD’, compared with (ev

$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

RoPB

<‘QD’, 1, (Φ), QoPD), step_1>

- ‘Request based’ slicing
  - Step 2: ‘B’ joins and calls ‘C
    ‘QoPD,QoPB’ should be ag
  - ‘on behalf of’: after D callin
    ‘on behalf of D’.

$$< step\_1, D \xrightarrow{c} B >$$

$$< step\_2, B \xrightarrow{c} C >$$

$$< step\_3, B \xleftarrow{d} C,' E' >$$

$$< step\_4, B \xleftarrow{d} B,' E',' ME' >$$

$$< step\_5, D \xleftarrow{d} B,' ME' >$$

<‘Q_D’, 1, (Φ), QoP_D), step_1>

<‘Q_D’, 2, (Q_D.1), (QoP_D,QoP_B), step_2>

RoP_C

- ‘Request based’ slicing
  - Strong point: More timingly
  - Shortcoming: Not enough
    - RoPB & RoPC conflicting?

$$< step\_1, D \xrightarrow{c} B >$$
$$< step\_2, B \xrightarrow{c} C >$$
$$< step\_3, B \xleftarrow{d} C,' E' >$$
$$< step\_4, B \xleftarrow{d} B,' E',' ME' >$$
$$< step\_5, D \xleftarrow{d} B,' ME' >$$

<‘Q_D’, 1, (Φ), QoP_D), step_1>

<‘Q_D’, 2, (Q_D.1), (QoP_D,QoP_B), step_2>

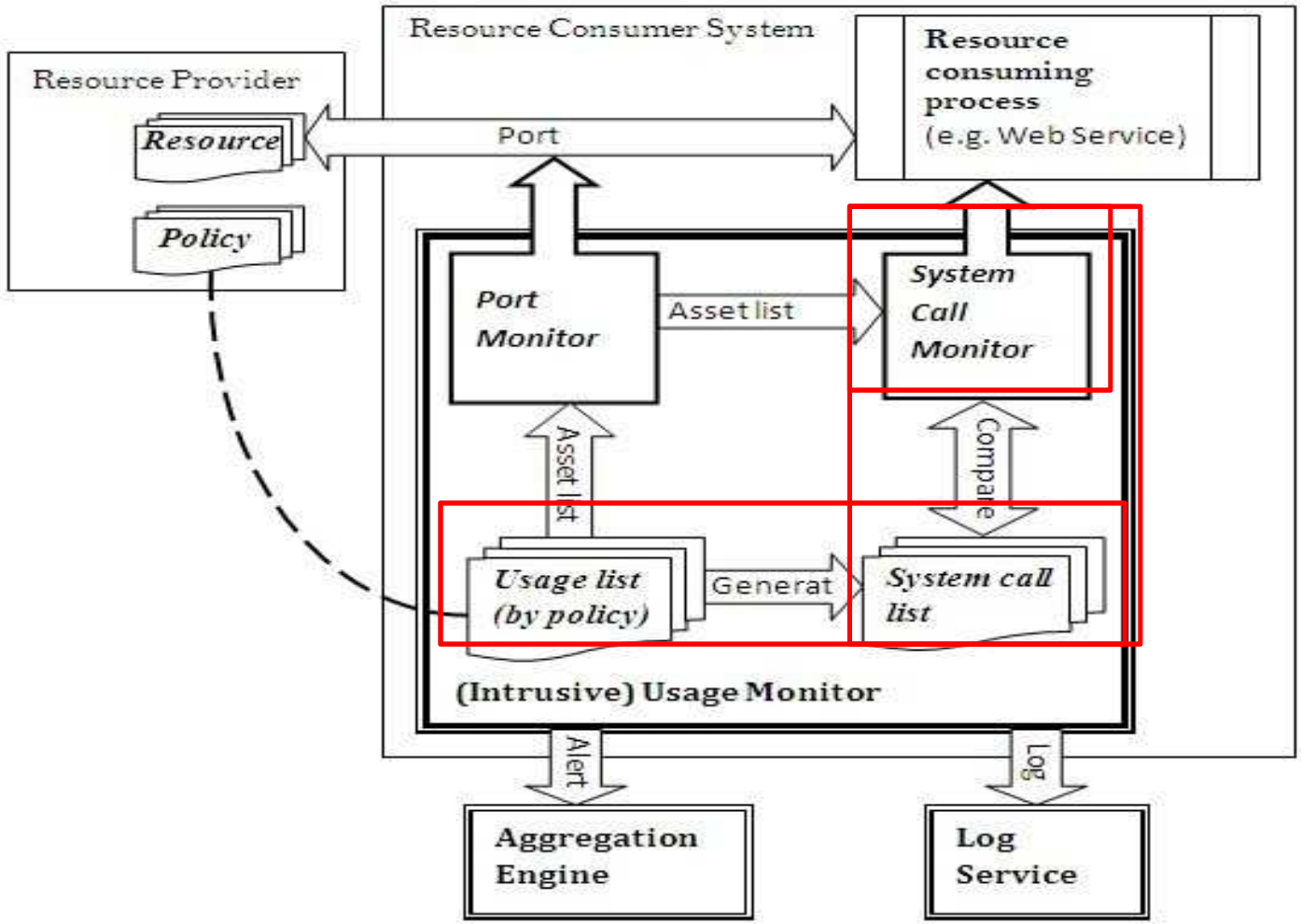
- ‘Pre-’ and ‘on the fly’ slicing
  - ‘Pre-processing’ a business process script, e.g. defined by ‘WS-BPEL2.0’
    - Can be done using ‘asset based’ slicing, not very timingly, but simple
    - Slice the script before the process is executed, to see whether security requirements / profiles are compatible or not.
    - No wast of partners’ resources
  - ‘On-the-fly’ processing a run-time service composition,
    - Should be done with both ‘request based’ and ‘asset based’ slicing
      - As partners that will join such a ‘random’ process can not be known at the starting time
      - Need to identify conflicts as early as possible, in order to avoid waste of partners’ (time & processing capabilities) resources

- Our approach is an ‘originator control’ (or ‘upstream provider control’, or ‘downstream information control’) approach which ensures
  - Providers’ policies upon their assets are maintained and respected in the whole business process, by all ‘downstream’ consumers

- Policy model
  - Express multiple ‘security attributes’ related to partners, assets and the context
  - Express various ‘consumption’ rights and obligations

- Context slicing
  - Track ‘assets derivation’ (service composition & information dissemination)
  - To know which asset is consumed by which consumers
  - In order to apply our policy model

Enforcement & monitoring



Conclusion

Contributions

A governance framework to enhance trust and assurance in virtual-enterprise.

Our security governance framework aims to :

- identify the risks and define security policies;
- enforcing fine-grained security & access control;
- ensure that providers' requirements have been fulfilled by consumers' security profiles;
- ensure end-to-end protection of shared assets and interoperability between security policies.


Thank You