A Secure EPC Discovery Service System in EPCGlobal Network

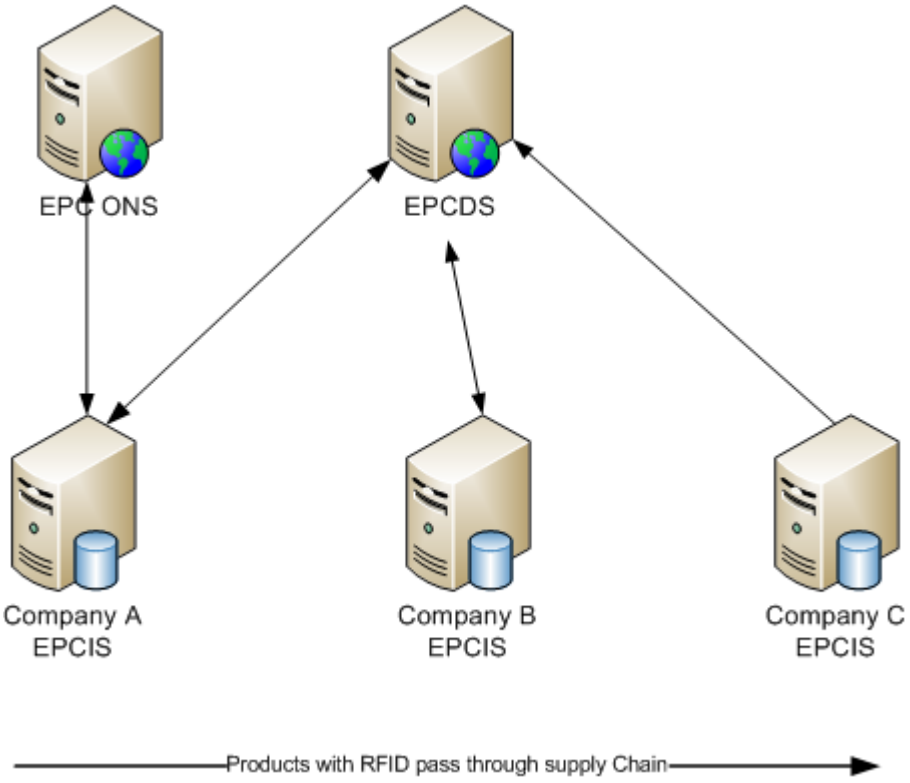
- Jie Shi, Darren Sim, Yingjiu Li, Robert Deng

SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Information Systems

Background

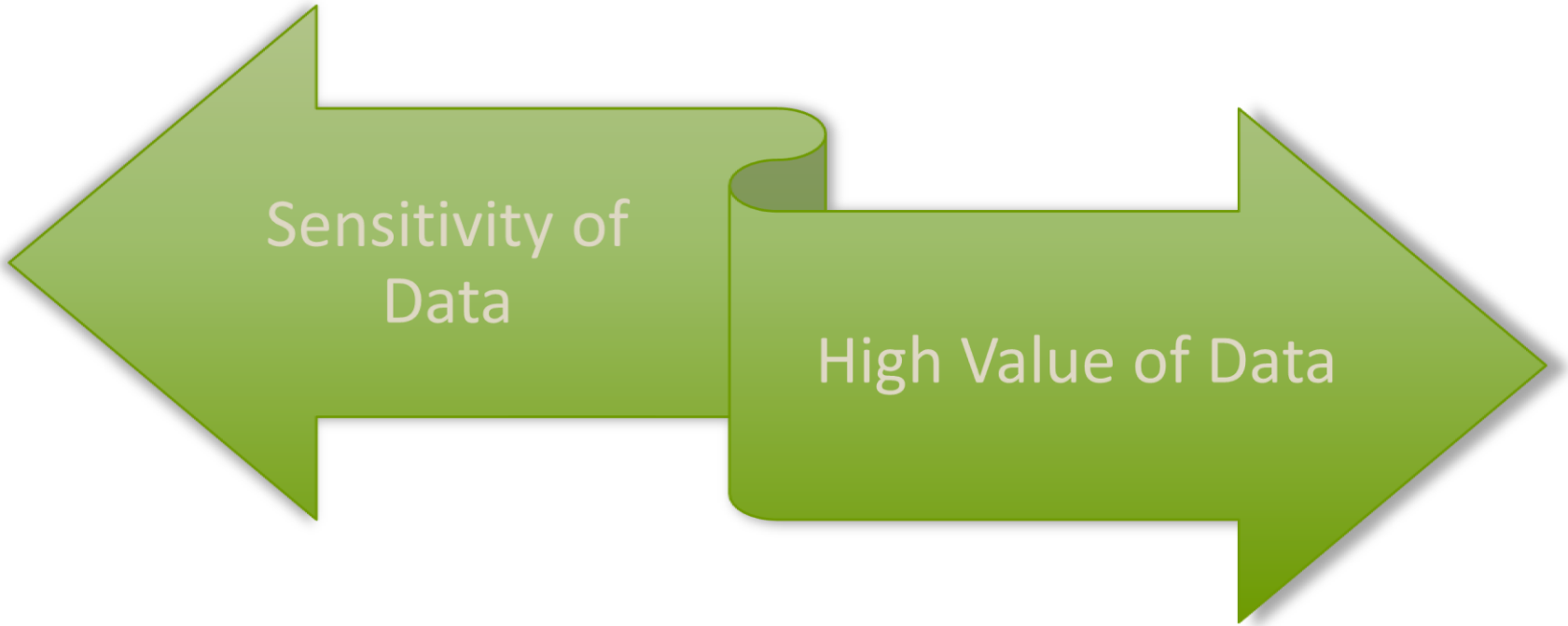
- Internet-of-Things (IOT) is the next big thing
- Vast amount of Data to be generated through IOT devices & users
- Urgent need for an effective search engine to make sense of this data
- System needs to process search efficiently, while remaining secure

Architecture of EPCglobal Network



Security of EPCDS

Motivation for Access Control



Security of EPCDS

Requirements and Challenges

Different access control policies from different EPCISes

Users may not be known in advance

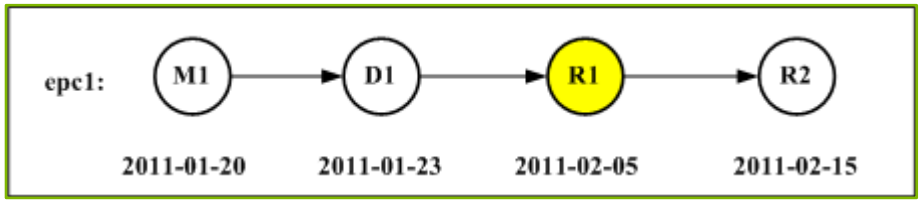
Visibility policy only in EPCDS

SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Information Systems

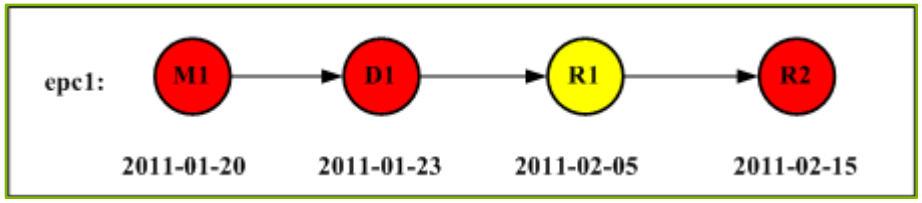
Visibility Policy

- Whole stream policy
- Up stream policy
- Down stream policy

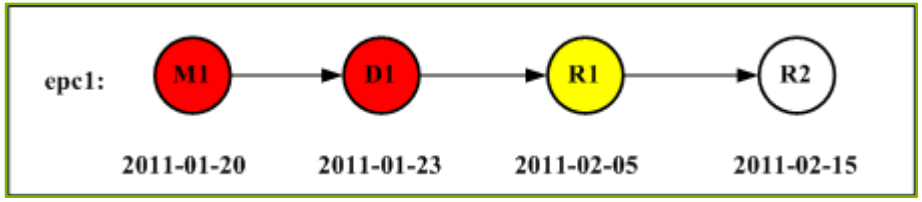
- Supply chain (R1 define policy for his event information about epc1)



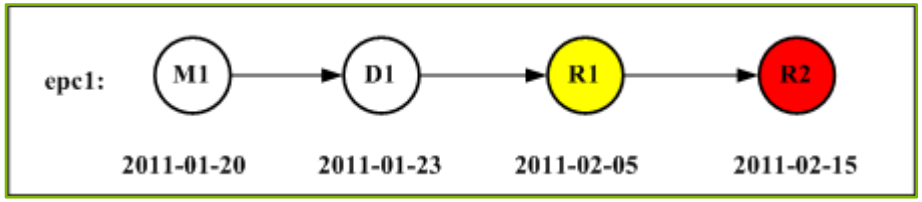
- Whole stream policy

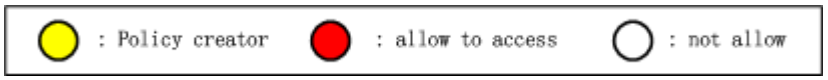


- Up stream policy



- Down stream policy





SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Information Systems

Attribute-based access control

- Subject attribute, object attribute, visibility attribute
- Authorization Language

AUL:=object condition ∧ subject condition
| object condition ∧ visibility condition
| object condition ∧ subject condition ∧ visibility condition

Example

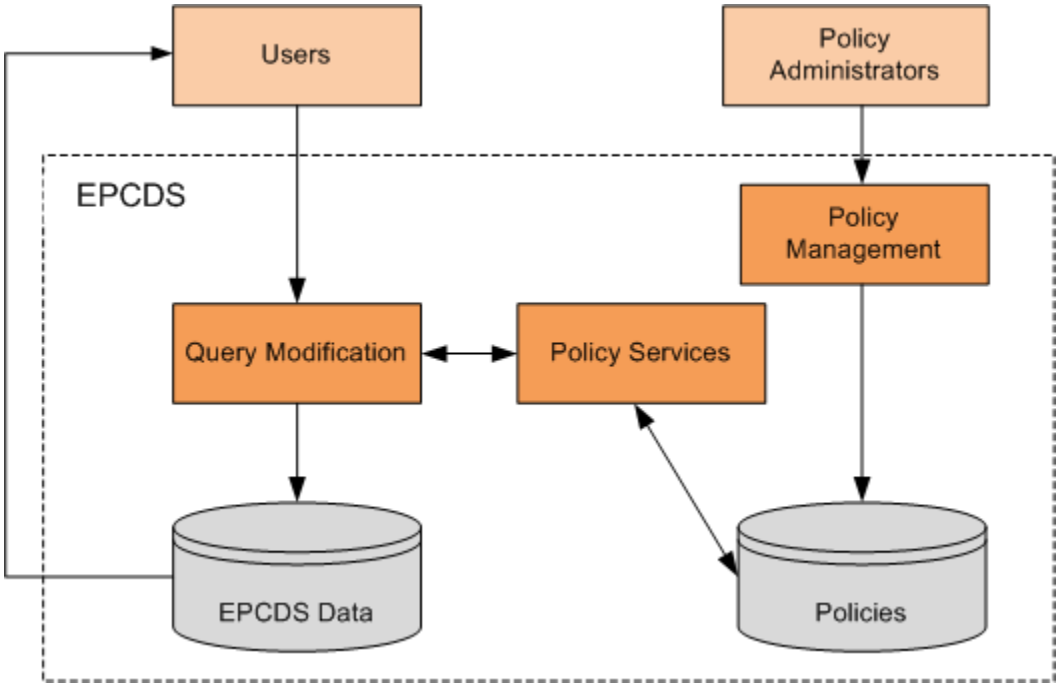
- Security requirements of company R1:

  For the information about any product handled after 2011-01-01, it is allowed to be accessed by the users of these companies who also handle this product and are distributor companies.

- Policy:

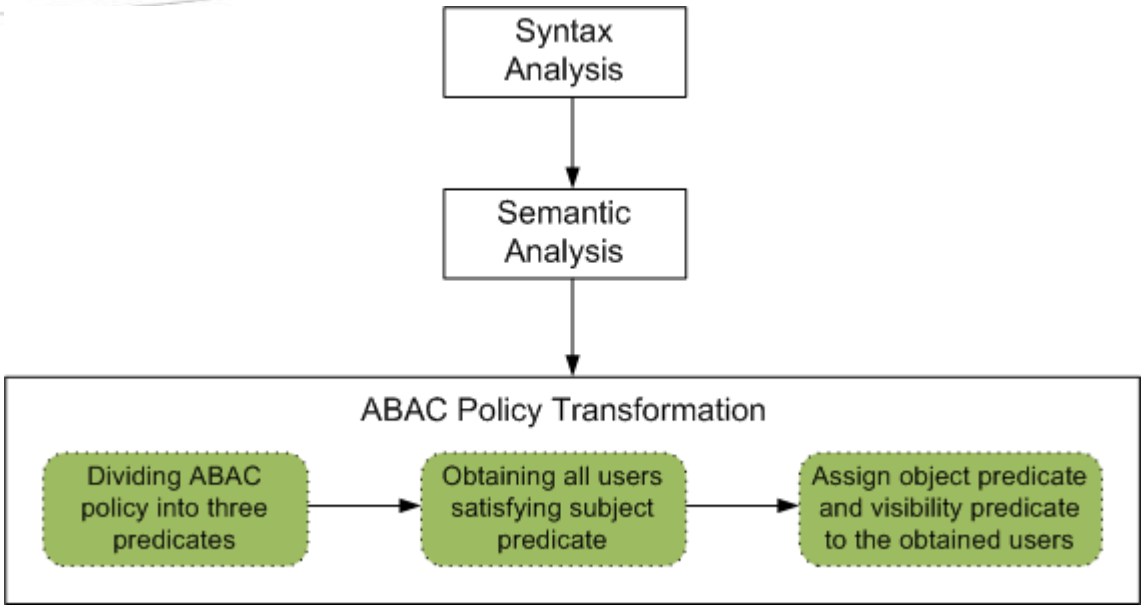
$$\text{Time} > 2011\text{-}01\text{-}01 \wedge \text{Visibility} = \text{whole-stream} \wedge \text{Role} = \text{Distributor}$$

SecDS system



Policy Management

- Manage access control policies
- Three steps taken in creation of ABAC policy



ABAC policy transformation enhances query performance at the cost of policy management

Example

(b) User-Companies

UserId	Name	CompanyId
U1001	Bob	C101
U1002	Andy	C102
U1003	John	C103
U1004	Peter	C104
U1005	Jack	C105

(c) Companies

CompanyId	Name	Role	URI
C101	M1	Manufacturer	http://www.m1.com
C102	D1	Distributor	http://www.d1.com
C103	D2	Distributor	http://www.d2.com
C104	R1	Retailer	http://www.r1.com
C105	R2	Retailer	http://www.r2.com

ID	Name	Predicate	Creator	CompanyId
1	pol_1	*Time* > 2011-01-01 ∧ (*Visibility* = whole-stream ∧ *Role* = Distributor)	C1001	C101
2	pol_2	*EPC* LIKE urn:epc:id:sgtin:4049588:083310:* ∧ *Name* IN (M1, D1, R1)	C1002	C102
3	pol_3	*EPC* NOT LIKE urn:epc:id:sgtin:4049588:083310:* ∧ *Visibility* = whole-stream	C1002	C102
4	pol_4	*Time* > 2011-03-01 ∧ *Visibility* = up-stream	C1004	C104

Table 2: ABAC Policy Table.

UserID	ABACPolicy	ObjectPredicate	Visibility	Creator	CompanyId
1002	pol_1	TIME > 2011-01-01	whole-stream	U1001	C101
1003	pol_1	TIME > 2011-01-01	whole-stream	U1001	C101
1001	pol_2	EPC LIKE urn:epc:id:sgtin:4049588:083310:*	NULL	U1002	C102
1002	pol_2	EPC LIKE urn:epc:id:sgtin:4049588:083310:*	NULL	U1002	C102
1004	pol_2	EPC LIKE urn:epc:id:sgtin:4049588:083310:*	NULL	U1002	C102
0	pol_3	EPC NOT LIKE urn:epc:id:sgtin:4049588:083310:*	whole-stream	U1002	C102
0	pol_4	TIME > 2011-03-01	up-stream	U1004	C104

Table 3: FGAC policy table.

Policy Service

- FGAC Policy Searching Service (FPSS)
- FGAC Policy Combining Service (FPCS)

Transformation of Visibility Policy

- **Whole-stream policy**:
  exist (select 1 from T1 where T1.companyId = c_1 and T.EPC = T1.EPC)
- **Up-stream policy**:
  exist (select 1 from T1 where T1.companyId = c_1 and T.EPC = T1.EPC and T1.Time < T.Time)
- **Down-stream policy**:
  exist (select 1 from T1 where T1.companyId = c_1 and T.EPC = T1.EPC and T1.Time > T.Time)

Policy Combination

DEFINITION 7.1 (POLICY COMPOSITION). *Given policies* $p_i = (u, pr_i, c_i), i \in [1 \ldots n]$ *for user* u *defined by companies* $c_i, i \in [1 \ldots n]$, *the combined predicate* $pr = (pr_1 \wedge o.ower = c_1) \vee \ldots \vee (pr_n \wedge o.owner = c_n)$

Own Predicate

Query Modification

- The basic idea of query modification is that before being processed, user queries are transparently modified to ensure that users can access only what they are authorized to access.
- Using the predicate combined in Policy Service to construct a temporary view and replace the table in users queries by this temporary view.

Experiments

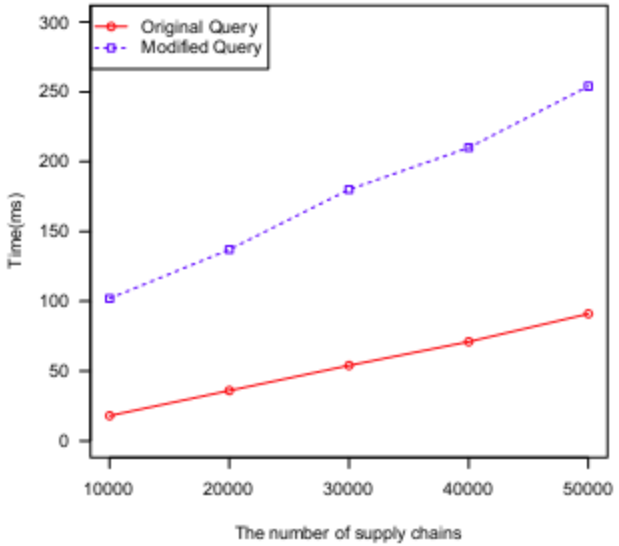


Figure 6: The performance of varying maximum number of supply chains. The other parameters: the number of EPCISes: 300; the max number of EPCISes in a supply chain: 30; the percentage of access control policies: 50%.

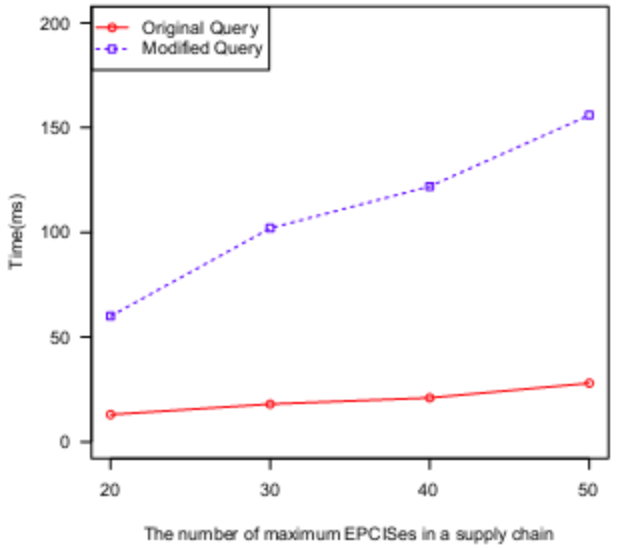


Figure 7: The performance of varying Max number of EPCISes in a supply chain. The other parameters: the number of supply chains: 10000; the number of EPCISes: 300; the percentage of access control policies: 50%.

Experiments

- The average query response time is about 260ms in a setting of 50,000 supply chains, 300 EPCISes, on average 20 EPCISes being involved in each supply chain and on average 10 policies being evaluated for each query.

Conclusion

- We analyzed and summarized the requirements of access control for EPCDS;
- We extended ABAC to satisfy these requirements, especially for visibility policy;
- We mainly use two approaches to enhance the performance of users queries
  - Transform ABAC to FGAC
  - Query modification
- We implemented prototype of SecDS and conducted rigorous experiment. The results validate SecDS is practical.

SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Information Systems

Do You Have Any Questions?

SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Information Systems