Scuola Superiore Sant'Anna
di Studi Universitari e di Perfezionamento

Scuola Superiore Sant'Anna, Pisa, Italy

cnit

Consorzio Nazionale Interuniversitario per le Telecomunicazioni, Pisa, Italy

Behavior-based Policies for preserving Confidentiality in PCE-based Multi-Domain MPLS networks

Molka Gharbaoui, Francesco Paolucci, Barbara Martini,
Filippo Cugini, Piero Castoldi

POLICY 2011
Pisa, Italy
June 6th – 8th, 2011

Introduction

- In Multi-Domain Multi-Carrier networks, different administrative entities cooperate in order to provide efficient computations of end-to-end paths between source and destination nodes.
- Adoption of Traffic Engineering (TE) techniques to efficiently manage resources and provide Quality of Service (QoS).
- Inter-domain TE implies the exchange of information between crossed domains, using the Path Computation Element (PCE) Architecture.
- Need to address the confidentiality issues arisen by the open advertisement of detailed network resources and the disclosure of salient intra-domain information to other carriers.

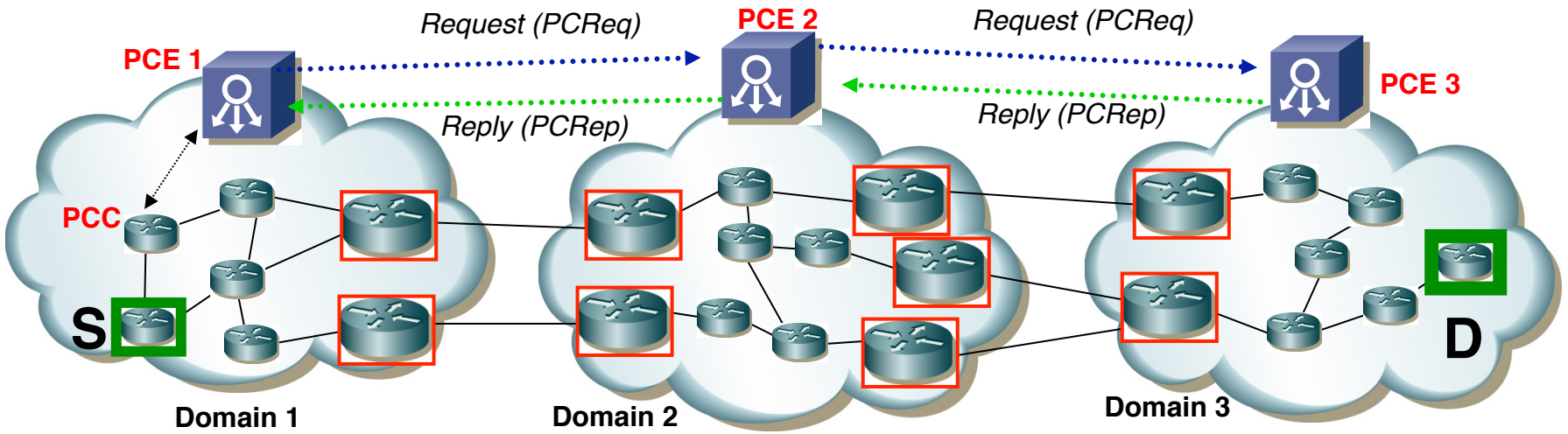
Inter-PCE path computation

- Multi-Domain Multi-Carrier networks:
  - Each PCE is responsible for intra-domain path computation requested by PCCs.
  - Distributed PCEs cooperate for inter-domain TE to find a shortest path between a source and a destination node.
- PCEP protocol is utilized to ask for a path:
  - Open and maintain a session
  - PCReq: Request message (end-points addresses, path attributes)
  - PCRep: Reply message
    - Path nodes list (Explicit Routing Object, ERO)
    - NO-PATH

PCE: Path Computation Element

PCC: Path Computation Client

TE: Traffic Engineering



Confidentiality in Inter-PCE Communications

- General agreements exist between adjacent domains to perform inter-domain path computations (technical and economical specifications especially the encryption of traversed paths)
- Basic level of trustworthiness: not sufficient to fully guarantee the information confidentiality.
- Despite users **authentication** and response **encryption**, correlations between apparently independent requests might reveal salient information.
- In fact, path computation is independent from resource reservation:
  - Multiple requests with different parameters may reveal congested network portions, node architectural limitations and constraints, etc.
  - Positive path replies not followed by path setup may be considered suspected.
  - Requests bursts must be carefully treated.

Critical PCEP Parameters

- **Bandwidth:**
  - Requested values guarantee resource reservations.
  - Significant amounts might reveal bottlenecks e.g., in case of negative reply.

- **Diversity and Bi-directionality:**
  - Needed to explore available resources along multiple disjoint routes or directions.
  - Risk for discovering topological limitations, node architectural constraints.

- **Metrics:**
  - Values returned in the PCRep such as number of traversed nodes, path computation time, etc.
  - Any change might indicate variations in the intra-domain resource availability.

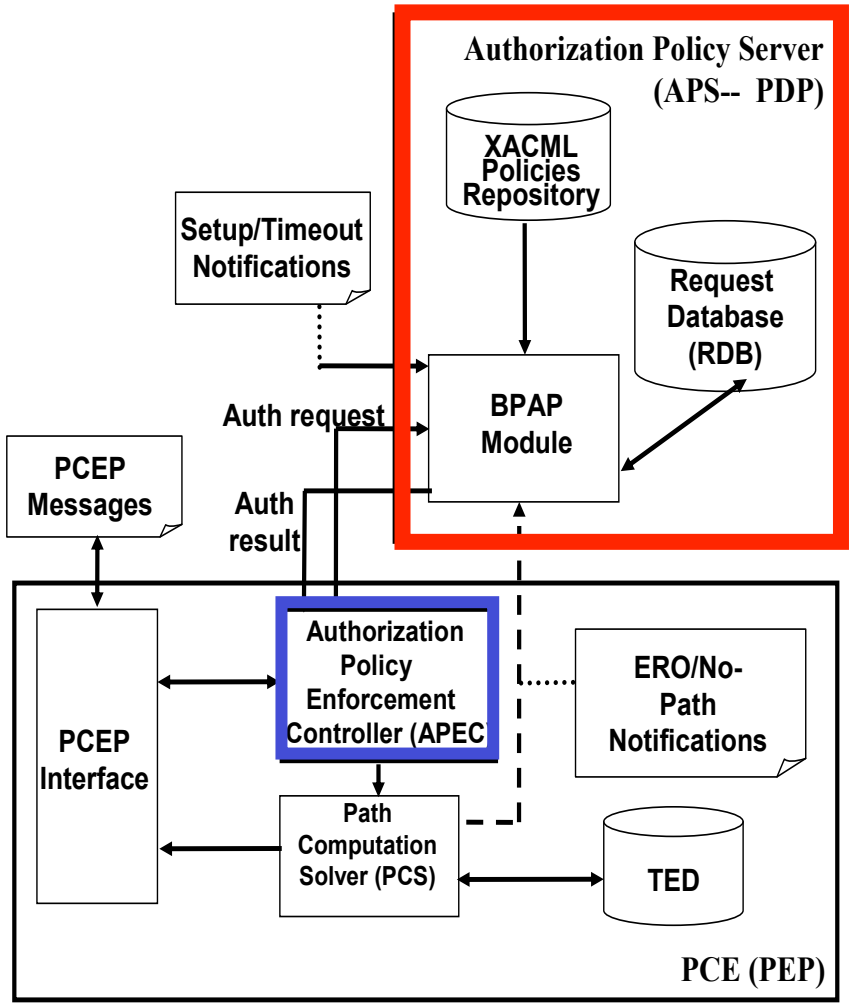
Proposed Approach

- A Behavior-based PCE Authorization Policy (BPAP) is proposed to prevent malicious utilizations of the PCEP procedures and preserve confidentiality across domains.
- It blocks requests in case of suspicious attacks following pre-determined patterns .
- XACML Policies are used to allow/deny the access to the path computation procedure, based on the risk estimation of an incoming request and on the previous behavior of the requester.

Behavior-based PCE Authorization Policy (BPAP)

Architecture

- **APEC: PCE internal module**
- **Central Authorization Policy Server**



Two-step procedure

- **First step** (upon a new PCReq from external peer)
  - Authentication, basic authorization based on simple access/deny access list
  - Request tagged *risk-free*, *unacceptable*, *critical*
- **Second step** (if *critical*)
  - Complex evaluation based on past client behavior and request classification

  - Request Database (RDB) stores client requests status: *Failure*, *Setup*, *Pending*, *Expired.*

  - Requests toward the same target are analyzed to identify possible pre-defined attack patterns.

  - Threshold-based mechanism triggers permit (path computation + PCRep) or deny (PCErr msg) outcome.

  - Critical resource information can be partially filtered (Apply restrictions to hide real network availability)

Compliant with the PEP/PDP architecture RFC 2904 – ITU-T Rec X.812

Risk Evaluation and Decision Algorithm

- Attack Scenario: ***Bandwidth monitoring*** (Bm):
  - Targets a destination node in case of egress target PCE or a remote border node in case of transit target PCE.

- Patterns recognition based on:
  - Requested bandwidth values
    - constant values, increasing-stepped values, decreasing-stepped values, saw-tooth values, etc.
  - Periodical behavior
    - Identical parameters requested N times
  - Time sequence of optional parameters occurrences
    - order of requests' arrivals, order of requested parameters

- Risk of an arriving request:

$$\rho=\alpha\rho_P+(1-\alpha)\rho_S$$

ρ_P: accounts for the detection of *patterns* ($\in \{0,1\}$)
ρ_S: accounts for the request *status* ($\in [0,1]$)
α: a (0,1) tunable weight that enhances or reduces the impact of the pattern discovery on the authorization decision

Example of Bandwidth monitoring

```
<PDP_req>
<pcc_id>1</pcc_id>
<req_id>1</req_id>
<src>1.1.1.1</src>
<dst>8.8.8.8</dst>
…
<bw>800</bw>
…
</PDP_req>
```

Arriving request tagged as critical

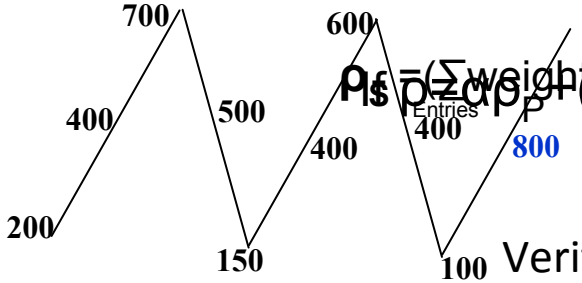
pcc_id && Destination address

Pcc_id	Target	Bandwidth	Status
1	8.8.8.8	200	Setup
1	8.8.8.8	400	Expired
1	8.8.8.8	700	Expired
1	8.8.8.8	500	Failure
1	8.8.8.8	150	Expired
1	8.8.8.8	400	Expired
1	8.8.8.8	600	Failure
1	8.8.8.8	400	Expired
1	8.8.8.8	100	Failure
1	8.8.8.8	800	

RDB portion

Pattern Identification ?

Compute the status of past requests and check anomalous behaviors



$\rho_S = (\sum \text{weight} \cdot \text{status}) / (\text{Max\_weight} * \text{number of entries})$

$\rho_P = \rho_C \rho_P + (1-d)\rho_S$ > Threshold : Pattern identified and confirmed by the arriving request: **sawtooth bandwidth => $\rho_P = 1$**

N identical entries in the RDB portion ?

Verified periodical trend => **ρ_P increases**

Authorization Policy

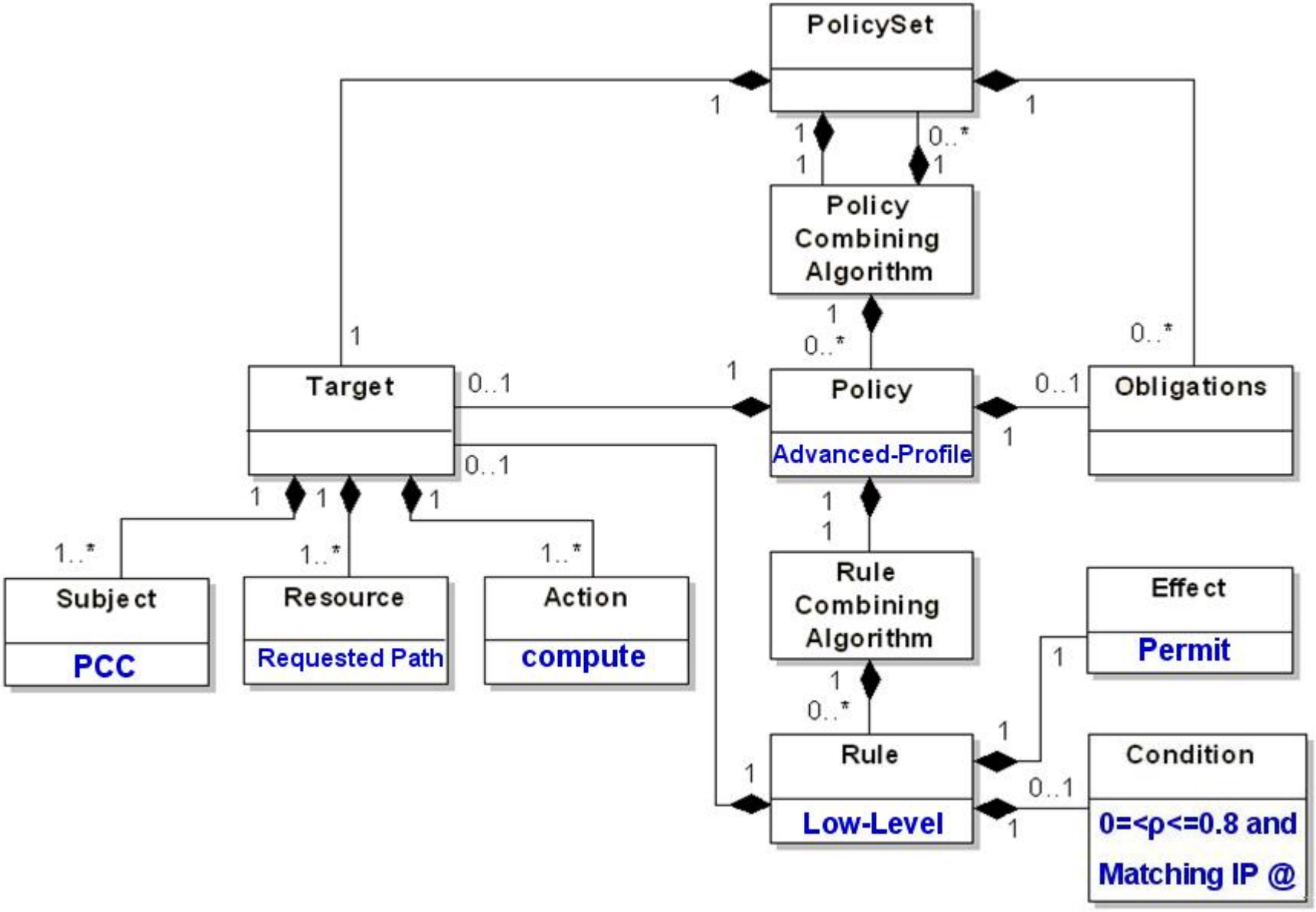
- A new set of trustworthiness agreements defined to specify **PCC profiles**. Each profile indicates what to send in the PCRep:
  - Advanced: Full set of information: all details about failures can be given. Metrics are returned if requested.
  - Standard: Restricted set: some failures can be explained. In case of success, only the Path-Key is sent back without any metric.
  - Basic: Minimal set: No details are sent along with the path computation result.

- Within each profile, 3 **disclosure levels** are defined to consider the recent behavior of a PCC during limited periods of time:
  - Low: The value of ρ is low enough to consider the request as safe. All agreed PCEP objects can be described.
  - High: The PCC behavior starts to be suspicious. Some privileges are removed.
  - Critical: The value of ρ is considered as critical. Some details can still be sent if it is lower than a given threshold, otherwise the session is closed and the PCC is considered malicious.

XACML Policies

- XACML policies are evaluated to determine whether or not the path computation request is allowed, i.e., does not affect or threatens the confidentiality of a receiving carrier.
- They describe the PCCs profiles coupled with the defined confidentiality levels.
- Policies are based on ρ. Once the decision is made, the disclosure level and the PCC profile are updated.
- If accepted, an addionnal filter might be applied on the information within the response sent back to the PCC, for further restrictions.

XACML Policy Model



Extract of Policies Adopted by BPAP at APS

```
<PolicySet PolicySetId="Behavior_Based_Policies" PolicyCombiningAlgId="...:permit-overrides">
  +<Target></Target>
   <!-- Policies 1/2/3 : PCCs having an Advanced Profile -->
     <Policy PolicyId="Low-Advanced-Profile" RuleCombiningAlgId="...:permit-overrides">
            +<Rule RuleId="Low_Level" Effect="Permit"></Rule>
            <Obligations>
                  + <Obligation ObligationId="Metric" FullfillOn="Permit"></Obligation>
                  +<Obligation ObligationId="C_Flag" FullfillOn="Permit"></Obligation>
                  + <Obligation ObligationId="TLV" FullfillOn="Permit"></Obligation>
            </Obligations>
     </Policy>
     + <Policy PolicyId="High-Advanced-Profile" RuleCombiningAlgId="...:permit-overrides">
     </Policy>
     + <Policy PolicyId="Critical-Advanced-Profile" RuleCombiningAlgId="...:permit-overrides">
     </Policy>
   <!-- Policies 4/5/6 : PCCs having a Standard Profile -->
     ...
  <!-- Policies 7/8/9 : PCCs having a Basic Profile -->
     ...
  <!-- Policy 10 : Unknown/Not authenticated PCCs -->
     ...
</PolicySet>
```

Extract of the Advanced Profile Policy

```
<Policy PolicyId="Low-Advanced-Profile" RuleCombiningAlgId="...:permit-overrides">
  <Target></Target>
  <Rule RuleId="Low_Level" Effect="Permit"></Rule>
        <Condition FunctionId=" ...: func t ion:and ">
        <Apply FunctionId=" ...: function:double-greater-than-or-equal ">
        <Apply FunctionId=" ...: function:double-one-and-only ">
              <SubjectAttributeDesignator AttributeId="... : rho " DataType=" ...#double " /></ Apply>
        <AttributeValue DataType=" ...#double ">0.0</ AttributeValue></ Apply>
        <Apply FunctionId=" ...: function:double-less-than">
        <Apply FunctionId="...: function:double-one-and-only ">
              <SubjectAttributeDesignator AttributeId="...: rho DataType="...#double " /></ Apply>
        <AttributeValue DataType="...#double ">0.8</ AttributeValue></ Apply>
        <Apply FunctionId="...: function: regexp-string-match">
        <AttributeValue DataType="...#string ">2.2.2.*</ AttributeValue>
        <Apply FunctionId="...: function: string-one-and-only ">
              <SubjectAttributeDesignator AttributeId="...:ip address " DataType="...# string " /></ Apply>
      </ Apply>
      </ Condition>
  </ Rule>
  <Obligations>
       <Obligation ObligationId="Metric " FulfillOn="Permit ">
            <AttributeAssignment AttributeId="...:Metric " DataType="...#string ">Send Metric </AttributeAssignment>
       </ Obligation>
       <Obligation ObligationId="C Flag" FulfillOn="Permit ">
            <AttributeAssignment AttributeId="...:C Flag " DataType="...#string ">Send C_Flag</AttributeAssignment>
      </ Obligation>
       <Obligation ObligationId="TLV" FulfillOn="Permit ">
            <AttributeAssignment AttributeId="...:TLV" DataType="...#string ">Send TLV</ AttributeAssignment>
       </ Obligation>
    </Obligations>
</Policy>
```

0 ≤ ρ ≤ 0.8

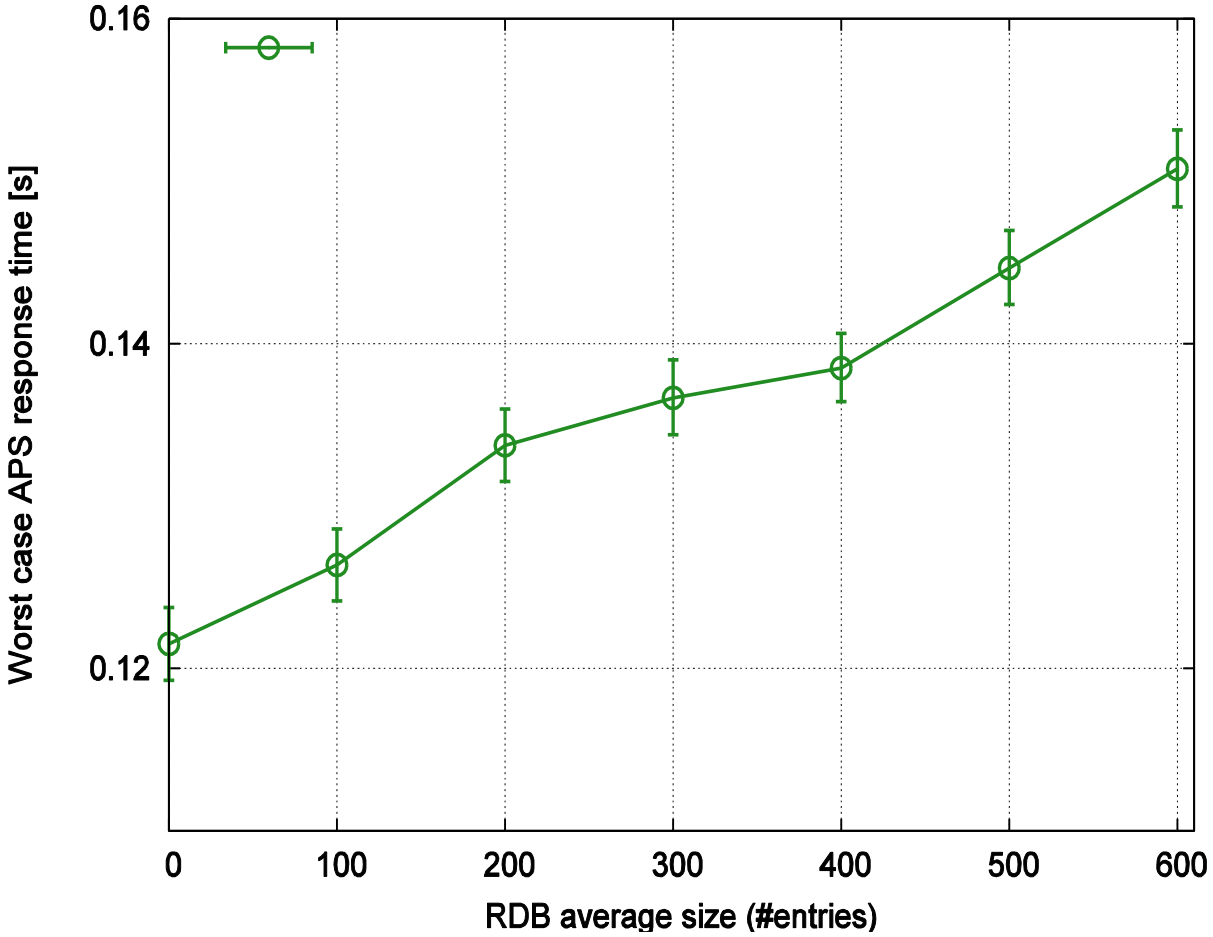
Profile based on PCC IP Address

Further restrictions on PCEP Objects

Experimental Results

Experimental testbed

- MPLS two-domain network made of commercial routers
- C++ based PCE running PCEP, APEC add-on module
- JAVA-based APS
- PCE<->APS: XML-based TCP socket



- Assuming the worst case, i.e. all the entries of the RDB are selected for pattern analysis, APS response time below 150ms => Good scalability performances.

Conclusion

- Confidentiality issues in Multi-Domain PCE communications were investigated.
- A BPAP scheme was proposed to analyze sequences of requests and allow/deny access to path computations using XACML policies.
- PCC profiles were additionally defined to filter the information sent back along with the path computation response.
- Trade-off between the need to protect information and the need to effectively utilize network resources.
- Scalable performances in terms of response time.

thank you!

m.gharbaoui@sssup.it