IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

CHAIRE
CYBER CNI
Sécurité des infrastructures critiques

UBO
université de bretagne occidentale

Software defined security for Network Function Virtualization

Manel Smine

Supervision: David Espes, Nora Cuppens and Frédéric Cuppens (04/2019 - 02/2020)

David Espes and Marc-Oliver Pahl (02/2020)

June 24, 2021

Agenda

1. Thesis context and objectives
2. 1st year contributions (04/2019 - 04/2020)
3. 2nd year contributions (04/2020 - 04/2021)
4. Future work (04/2021 - 04/2022)

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
4. Future work

Thesis Context and Objectives

- **Context**
  - Network Function Virtualization (NFV) security
    - It has many advantages:
      - reducing hardware cost
      - deployment in fast time
    - Suffers from several security issues (e.g., security policy violation [1])

- **Research Questions**
  - How to deploy access control policies on NFV architectures?
  - How to take into account different access control models?
  - How to deal with conflicting rules such as exceptions?
  - How to deploy optimally an access control policies on NFV services?

[1] Lal, S., Taleb, T., Dutta, A.: Nfv: Security threats and best practices. IEEE Com-munications Magazine55(8), 211–217 (2017)

:

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
4. Future work

1st year contribution: NFV access & flow control as a service

- A formal model is developed to specify the access control policy to be deployed on VNFs services.
  - It can deploy most types of access control policy such as RBAC,ORBAC, ABAC, etc.
  - It does not require any modification of the NFV infrastructure.
- A provably correct transformation of a high level security policy into a Domain Type Enforcement concrete specification.
- Automatic deployment of the concrete specification on NFV services.

Smine, M., Espes, D., Cuppens-Boulahia, N., & Cuppens, F. (2020, June). Network Functions Virtualization Access Control as a Service. In *IFIP Annual Conference on Data and Applications Security and Privacy* (pp. 100-117). Springer, Cham.

Open Question: How to deal with exceptions ?

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
   d. Implementation
4. Future work

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
   d. Implementation
4. Future work

Access Control Policy

- **Open policy:** Contains only negative authorizations.
- **Closed policy:** Contains only positive authorizations.
- **Mixed policy**
  - Contains both positive and negative authorizations.
  - Express high level specification and specify more complex security policies
  - More expressive language for access policy specification

Access Control Policy deployment

- High level access control models are based on mixed policies
- At low level, the enforcement are based on closed policies
- Mixed policies should be transformed to closed policy.
    - seems to be straightforward
    - but, it is challenging if the high level mixed policy contains exceptions.

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
   d. Implementation
4. Future work

Related Work: Management of access control policies on NFV

[Suarez, et al'20]

- propose RDAC – an approach that combines the best of RBAC and DTE.
- It allow secure resource sharing among the different players involved in providing services over 5G networks.

[Smine, et al'20]

- Propose an access control as a service model for NFV services.
- It can handle most kind of access control policy models.
- It provides an efficient method for deploying access control policies at the concrete level without requiring the modification of the NFV infrastructure.

The deployment of complex policies containing exceptions leads often to very complex DTE specification

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
   d. Implementation
4. Future work

A Priority-based Domain Type Enforcement for Exception Management

- A provably correct DTE-based access control model that can efficiently enforce complex policies.
  - It allows an efficient deployment of complex access control policies containing exceptions on NFV services.

- Our proposed model is up to 5 orders of magnitude more efficient than the existing solution when dealing with high-level complex policies containing exceptions.

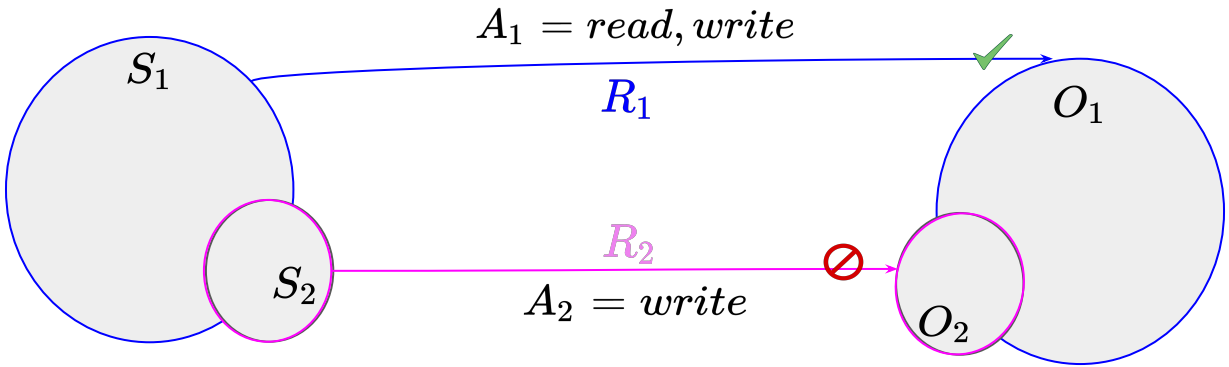
Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
      i. Exception management in DTE
   d. Implementation
4. Future work

Exception Formalization

$$R_1 = \langle S_1, A_1, O_1, D_1 = allow \rangle \qquad R_2 = \langle S_2, A_2, O_2, D_2 = deny \rangle$$

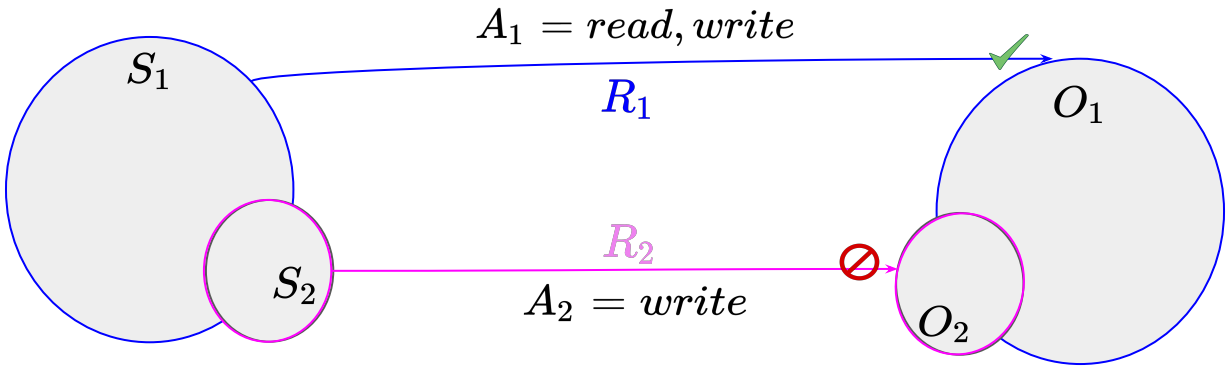


$$S_1 \cap S_2 \neq \emptyset \ \wedge \ O_1 \cap O_2 \neq \emptyset \ \wedge \ A_1 \cap A_2 \neq \emptyset \ \wedge \ D_1 \neq D_2$$

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Exception: transformation towards closed policies

$$R_1 = \langle S_1, A_1, O_1, D_1 = allow \rangle \qquad R_2 = \langle S_2, A_2, O_2, D_2 = deny \rangle$$



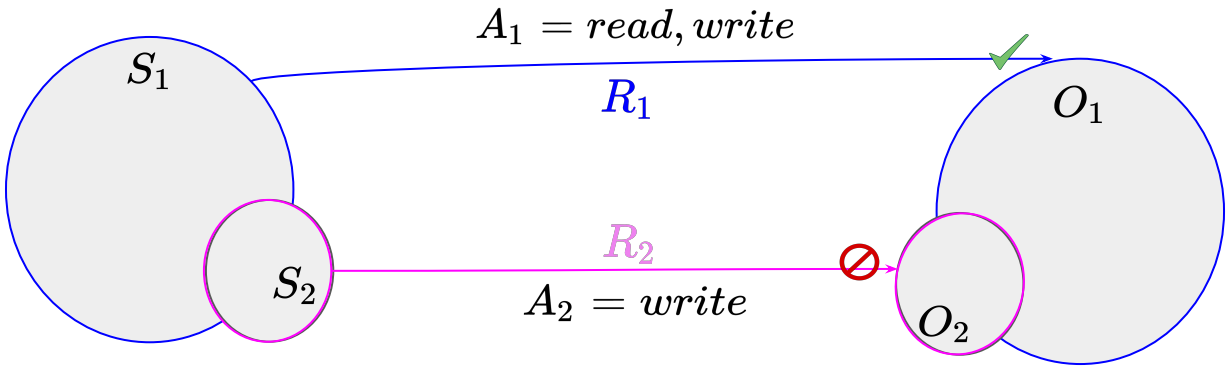
Case 1: The rule R_1 is enforced before R_2

We can just remove the negative authorization (deny rule)

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Exception: transformation towards closed policies

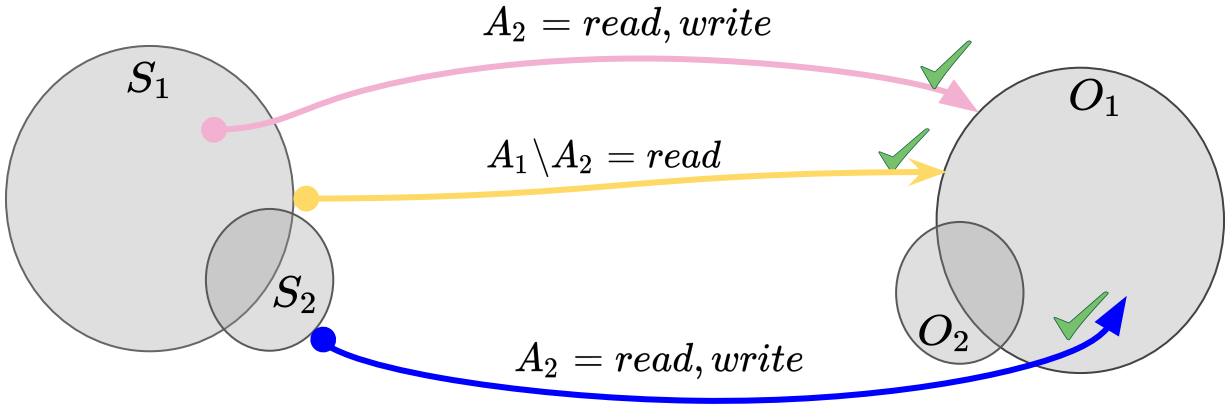
$$R_1 = \langle S_1, A_1, O_1, D_1 = allow \rangle \qquad R_2 = \langle S_2, A_2, O_2, D_2 = deny \rangle$$



Case 2: The rule R_2 is enforced before R_1

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Exception: transformation towards closed policies



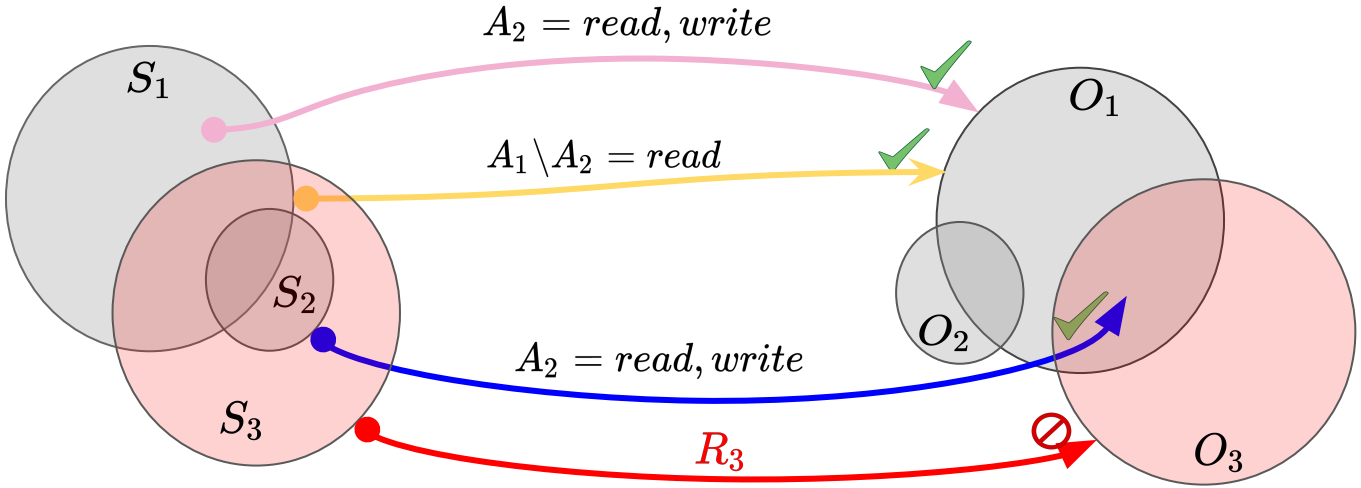
Case 2: The rule R_2 is enforced before R_1

$R_1^* = \langle S_2, A_2, O_1 \setminus O_2, allow \rangle$ $R_2^* = \langle S_1 \setminus S_2, A_2, O_1, allow \rangle$ $R_3^* = \langle S_1, A_1 \setminus A_2, O_1, allow \rangle$

Each exception represented by two rules gives 3 positive authorization rules.

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Exception: transformation towards closed policies



Case 2: The rule R_2 is enforced before R_1

$R_1^* = \langle S_2, A_2, O_1 \setminus O_2, allow \rangle$ $R_2^* = \langle S_1 \setminus S_2, A_2, O_1, allow \rangle$ $R_3^* = \langle S_1, A_1 \setminus A_2, O_1, allow \rangle$

3 more exceptions to transform => each exception will add more rules to the closed policy

Asymptotic complexity

the number of rules in the closed policy $= \sum_{i=1}^{n} \Theta_i$ with $\Theta_i = \left\{ \begin{array}{rl} 3^{\|\Omega_i\|} & \text{if } D_i = allow \\ 0 & \text{if } D_i = deny \end{array} \right\}$

r : number of rules in the closed policy

Ω_i : number of exceptions

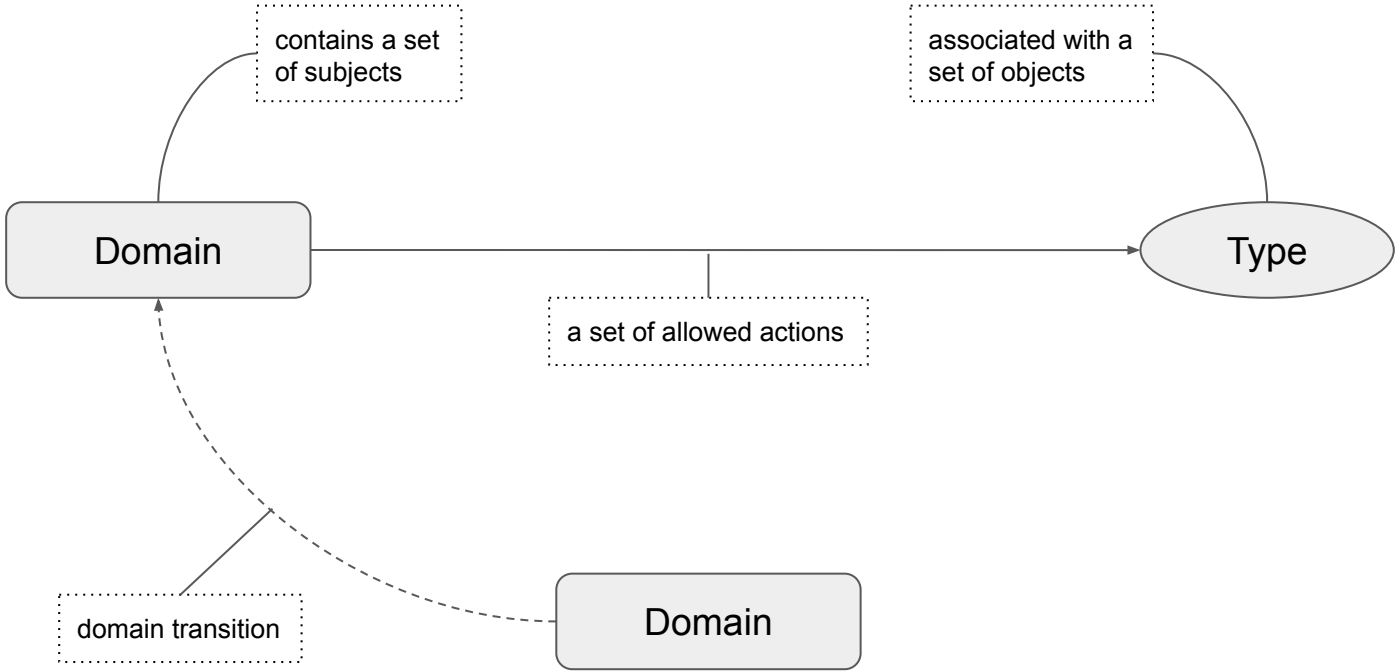
- A closed policy is transformed into a DTE by considering
  - Each subject as a domain
  - Each object as a type.
- Exponential growth of the number of rules in the closed policy
- Exponential growth of the number of DTE domains and types
- Increase policy evaluation

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Plan

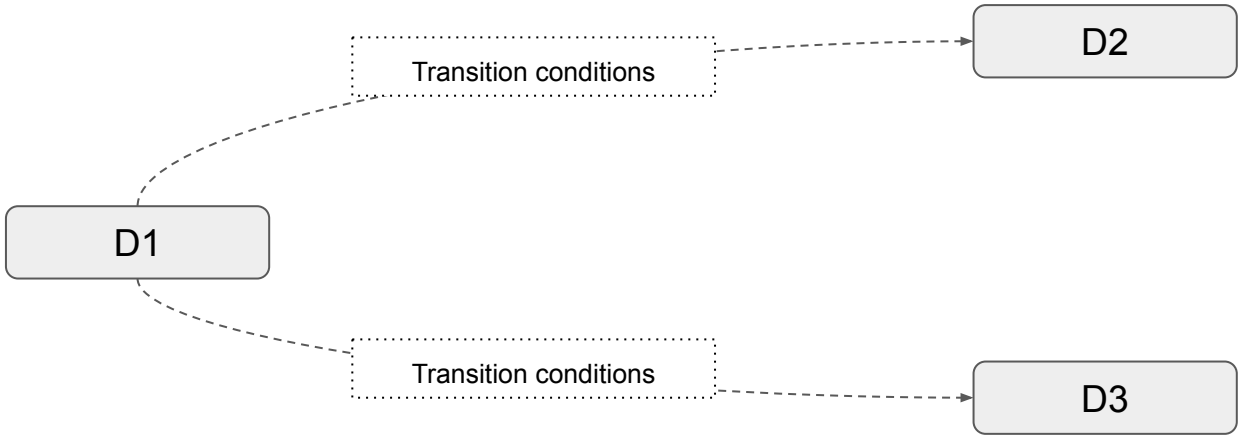
1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution
   a. Background
   b. Related work
   c. The proposed model
      i. Exception management in DTE
      ii. Priority-based DTE
   d. Implementation
4. Future work

Classic DTE


contains a set of subjects
associated with a set of objects
Domain
Type
a set of allowed actions
domain transition
Domain

Domain transition extension

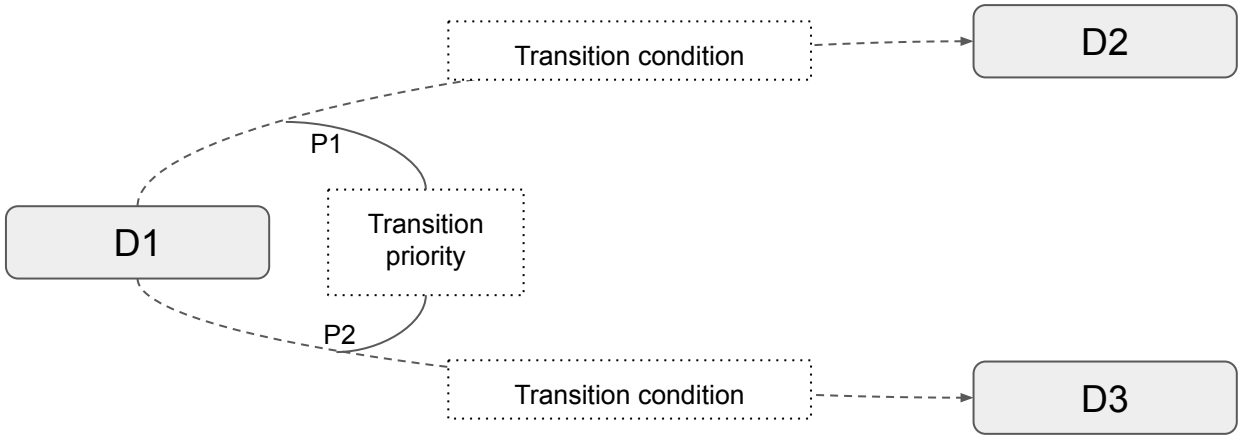
- Extend the concept of DTE domain transition
  - A transition conditions: a set of conditions that should be satisfied



Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Domain transition extension

- Extend the concept of DTE domain transition by adding
  - A transition conditions: a set of conditions that should be satisfied
  - A transition priority:



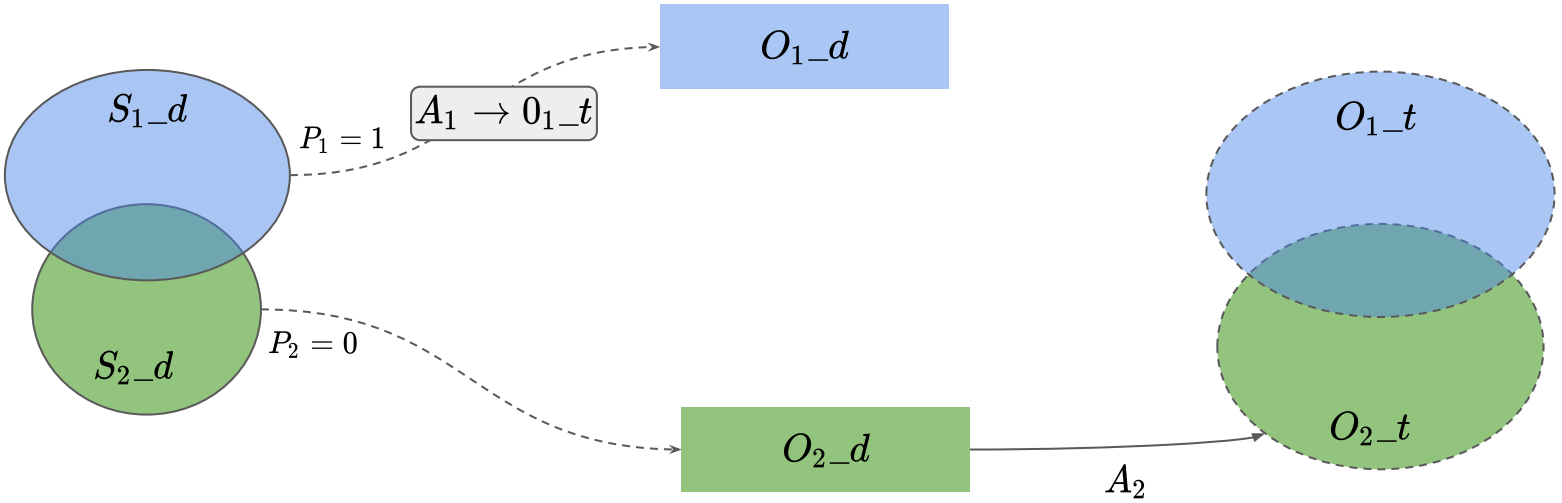
- The prioritized transition is the transition that has the highest priority
  - a subject in D1 transits to D2 iff P1 > P2
  - a subject in D1 transits to D3 iff P1 < P2

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Exception transformation towards extended DTE

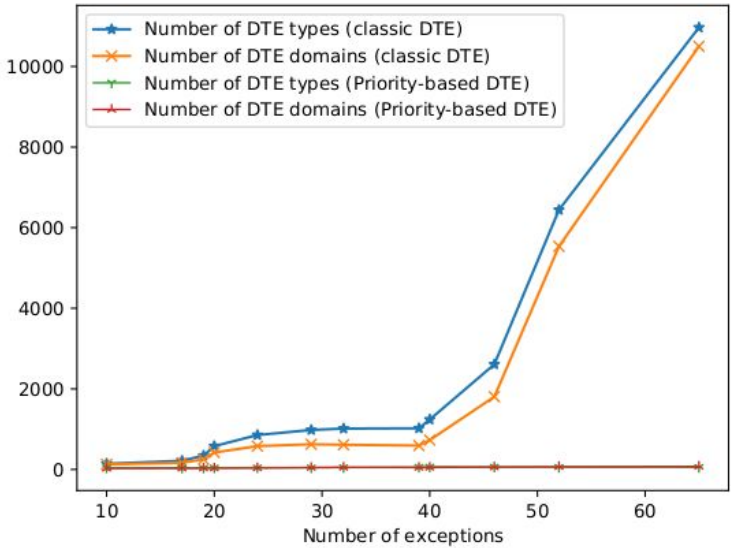
$$R_1 = \langle S_1, A_1, O_1, deny \rangle \qquad R_2 = \langle S_2, A_2, O_2, allow \rangle$$

$$S_1 \cap S_2 \neq \emptyset \wedge O_1 \cap O_2 \neq \emptyset \wedge A_1 \cap A_2 \neq \emptyset$$

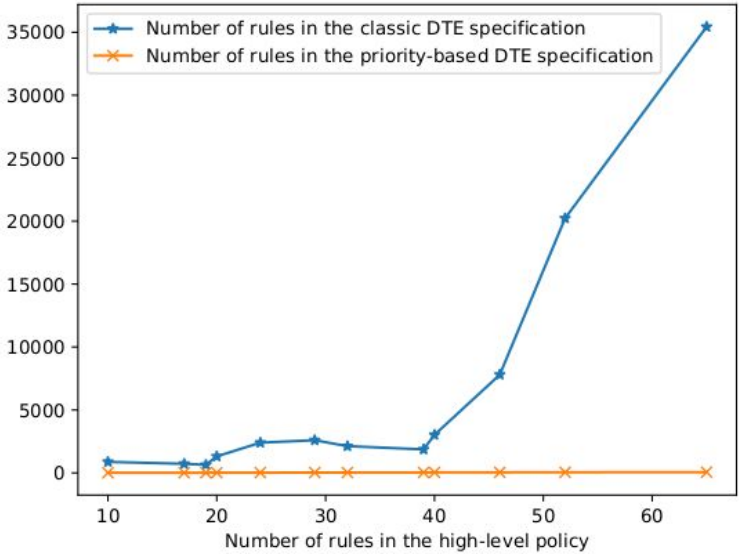


Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Experimental Results (1)



Growth of the number of required DTE domains and types

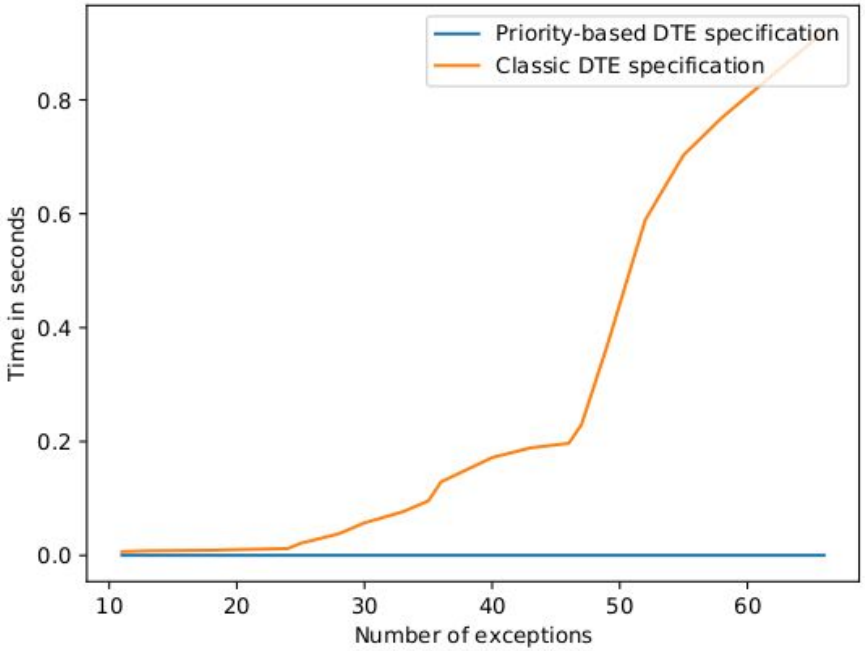


Growth of the number of rules

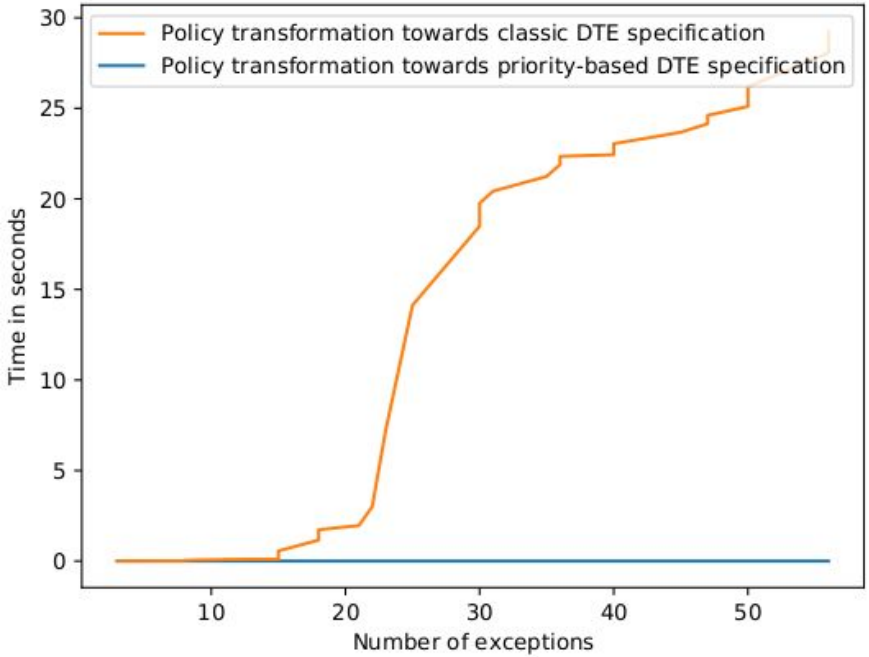
Experimental setup: Linux with an Intel Xeon E5-2680 v4 Processor with 16 vCPU and 32 GB of RAM

Manel Smine, David Espes, Nora Cuppens-Boulahia, Frédéric Cuppens, Marc-Oliver Pahl:A Priority-Based Domain Type Enforcement for Exception Management. FPS 2020: 65-81

Experimental Results (2)



The time required to transform the policy



The access query evaluation time

Experimental setup: Linux with an Intel Xeon E5-2680 v4 Processor with 16 vCPU and 32 GB of RAM

Plan

1. Thesis context and objectives
2. 1st year contribution
3. 2nd year contribution

   This work has been published in **FPS 2020**
4. Future work

Plan

1. Thesis context and objectives
2. 1st year contribution
3. Current work

Current Work

- Optimal deployment of an access policies
  - Optimal placement of policy deployment points (firewall)
    - Optimizing the impact in terms of latency
    - Optimizing the physical resource

Thank you