VPP Modular Middlebox: update

Korian Edeline, Justin Iurman

University of Liège

measurement and architecture for a middleboxed internet

measurement architecture experimentation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 688421.The opinions expressed and arguments employed reflect only the authors' view. The European Commission is not responsible for any use that may be made of that information.

Reminder

MMB plugin built in **VPP** (**FD.io**) environment to simulate middleboxes behavior

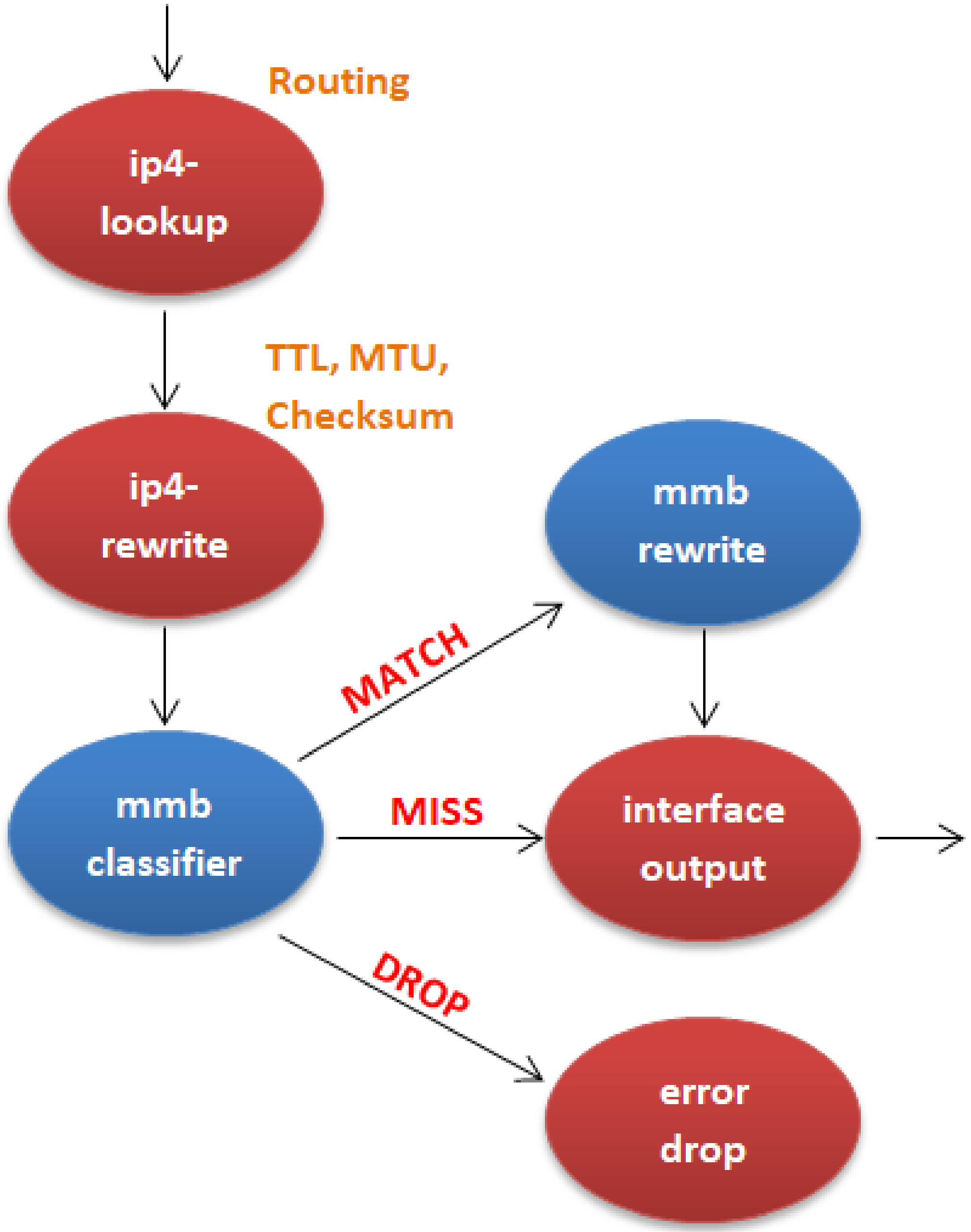
Easy to deploy or use (physical/virtual)

Light and fast

Grammar defined for rules:

mmb add <match> [<match> …]

<target> [<target> …]



Mask-based approach

- 3 logical paths:
  - Fast path: **==** constraints, no options
  - Slow path 1: complex constraints (**!=**, **<**, **>**, **<=**, **>=**), no options
  - Slow path 2: options

Fast path

- **==** constraints, no options
- Complexity:
  - **M** = #Masks, **K** = max(#Keys)
  - find_key(mask, packet): O(log **K**)
  - classify(packet): O(**M** log **K**)
- Examples:
  - 100 rules on 5-tuples (**M**=1, **K**=100): O(1 log 100)
  - 50 rules on 5-tuples and 50 on 3-tuples (**M**=2, **K**=50): O(2 log 50)

Slow path 1

- “Complex” constraints (**!=**, **<**, **>**, **<=**, **>=**), no options
- No logarithmic time find_key()
- One check per individual complex sub-constraint

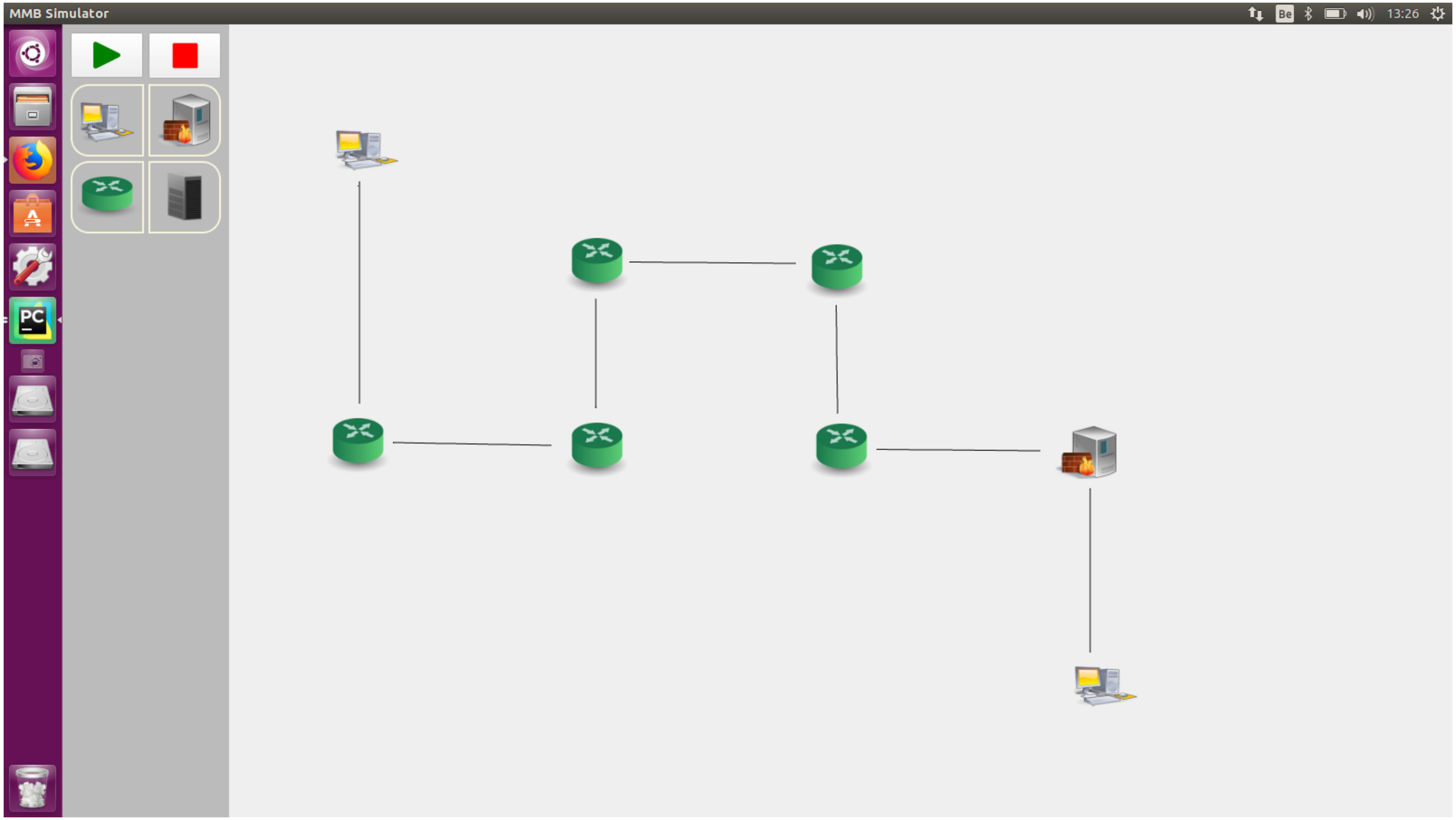
Slow path 2

- TCP Options
- Extra steps to find options offsets
- Rewrite operation also heavier

GUI

- Graphical User Interface for virtual network creation
- Users can add nodes (router, switch, middlebox, endpoint)
- Users can configure the routing and configure middlebox behavior.

GUI



Next steps

- Complete performance evaluation
- GUI