Interactive Session #1

Panagiotis Famelis, Georgios P. Katsikas

ETSI TFS – Hackfest #3, October 17, 2023

TeraFlowSDN protobuf files

```
cd ~/controller/proto
```

2

Protobuf files

◉ Protobuf files define the way the components communicate between themselves and outside

◉ If we need to build a script or program to communicate directly with TFS' components, protobufs are our guide!

◉ Let's take a look at them

Writing monitoring probes

`cd ~/controller/proto`

Latency probe in Python

◉ The probe we used in the demo was written in Rust by **Carlos Natalino** (Chalmers University)

◉ The point of this session is to rewrite it in python, using:

  ◉ Compiled protobufs (`src/common/proto`)

  ◉ Component Clients (`src/<component>/client`)

◉ But first let's study the existing probe
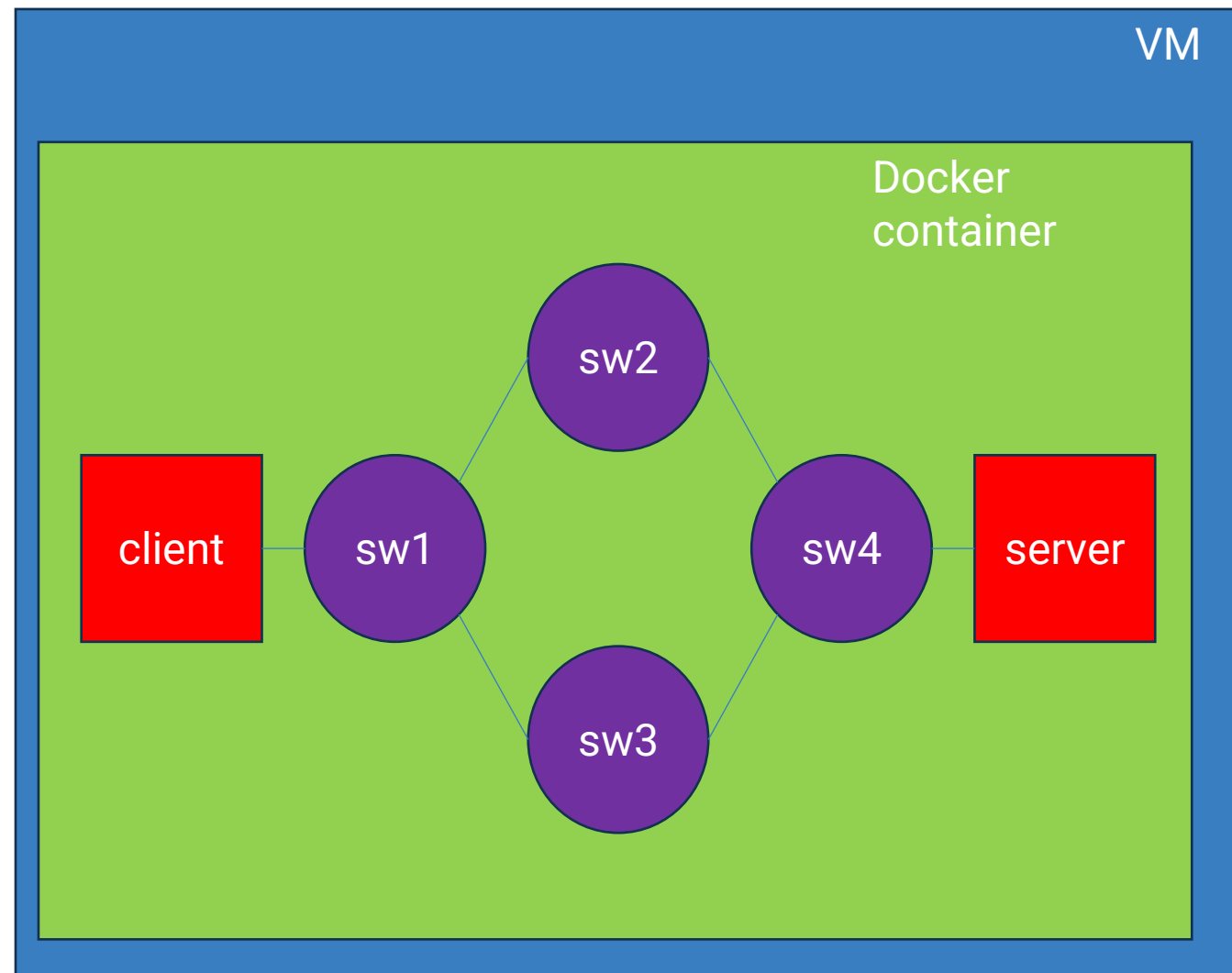
How the probe works

The probe has two parts:

◉ A pinger: periodically sends pings to calculate the latency, which is then forwarded to the agent

◉ An agent: Listens for Context events and registers KPIs for each service.

◉ Also creates KPI values from latency values
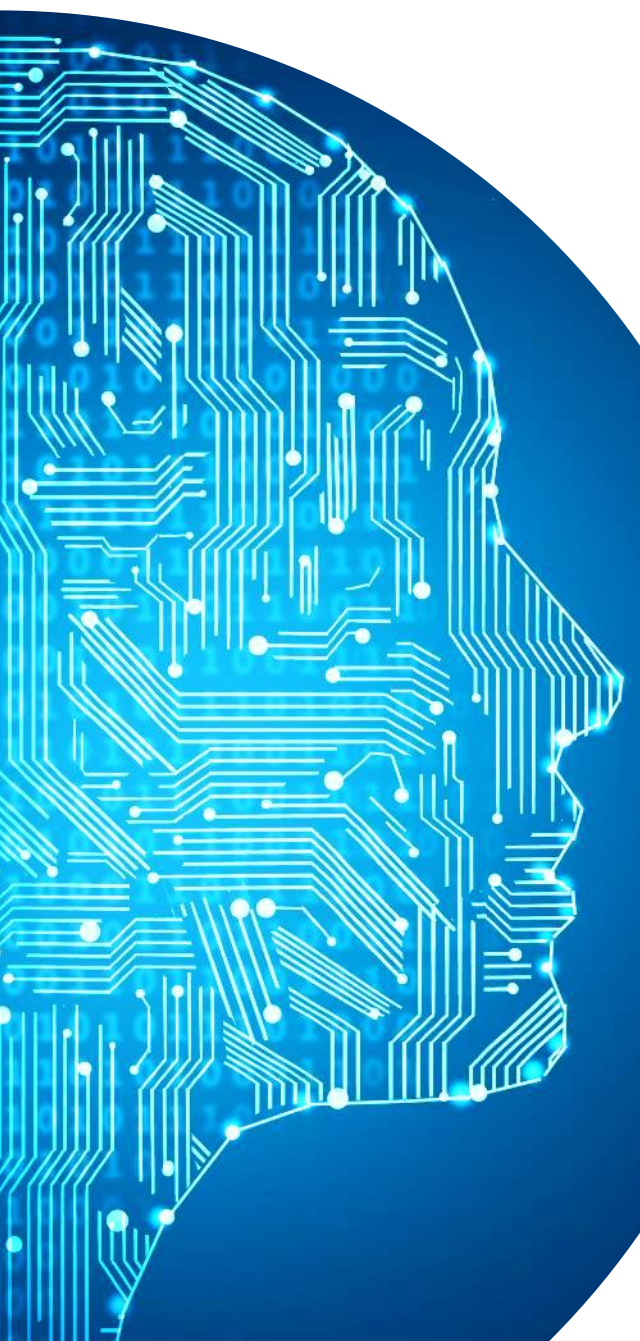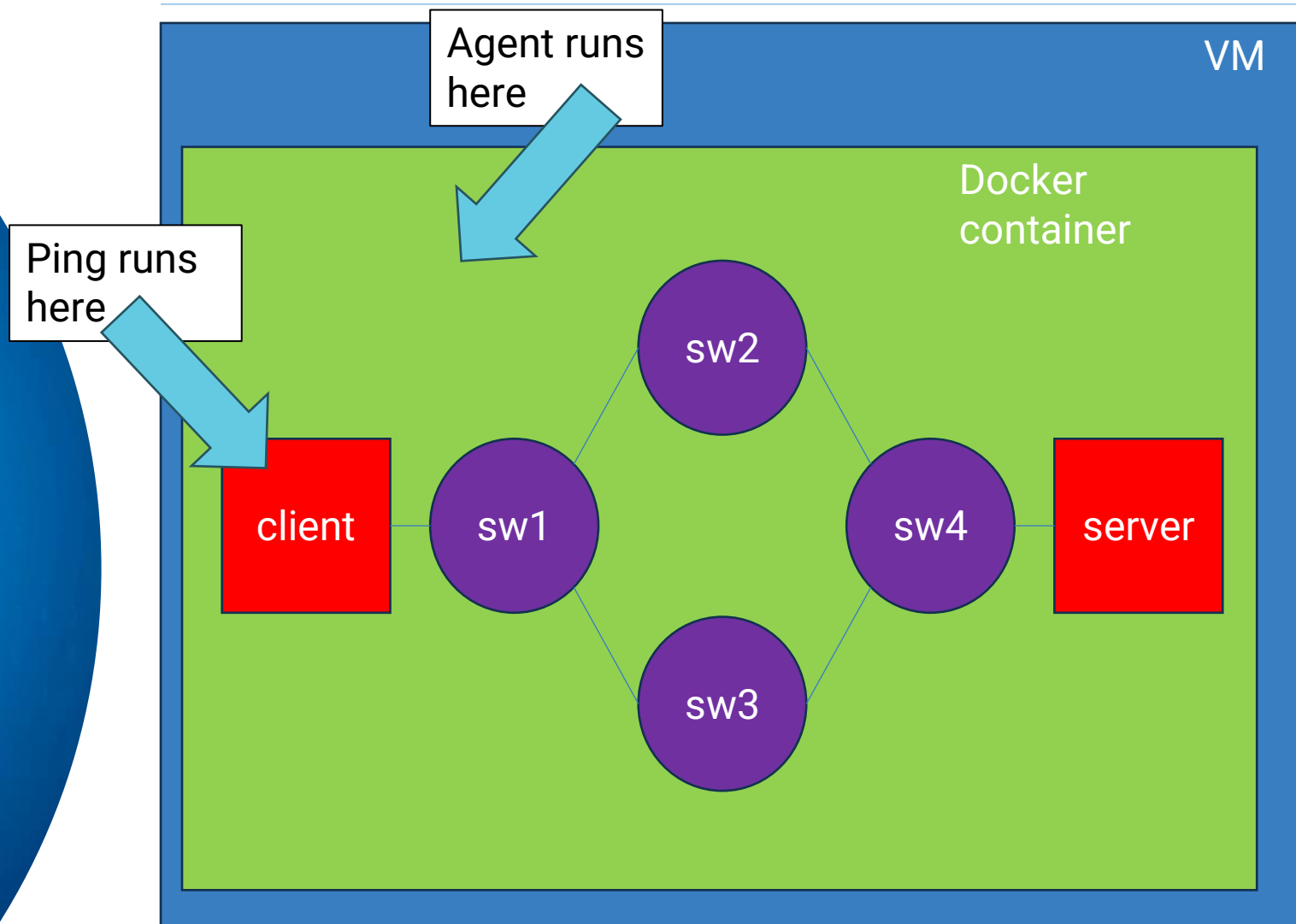
Mininet container

VM

Docker container

client — sw1 — sw2 — sw4 — server

sw3
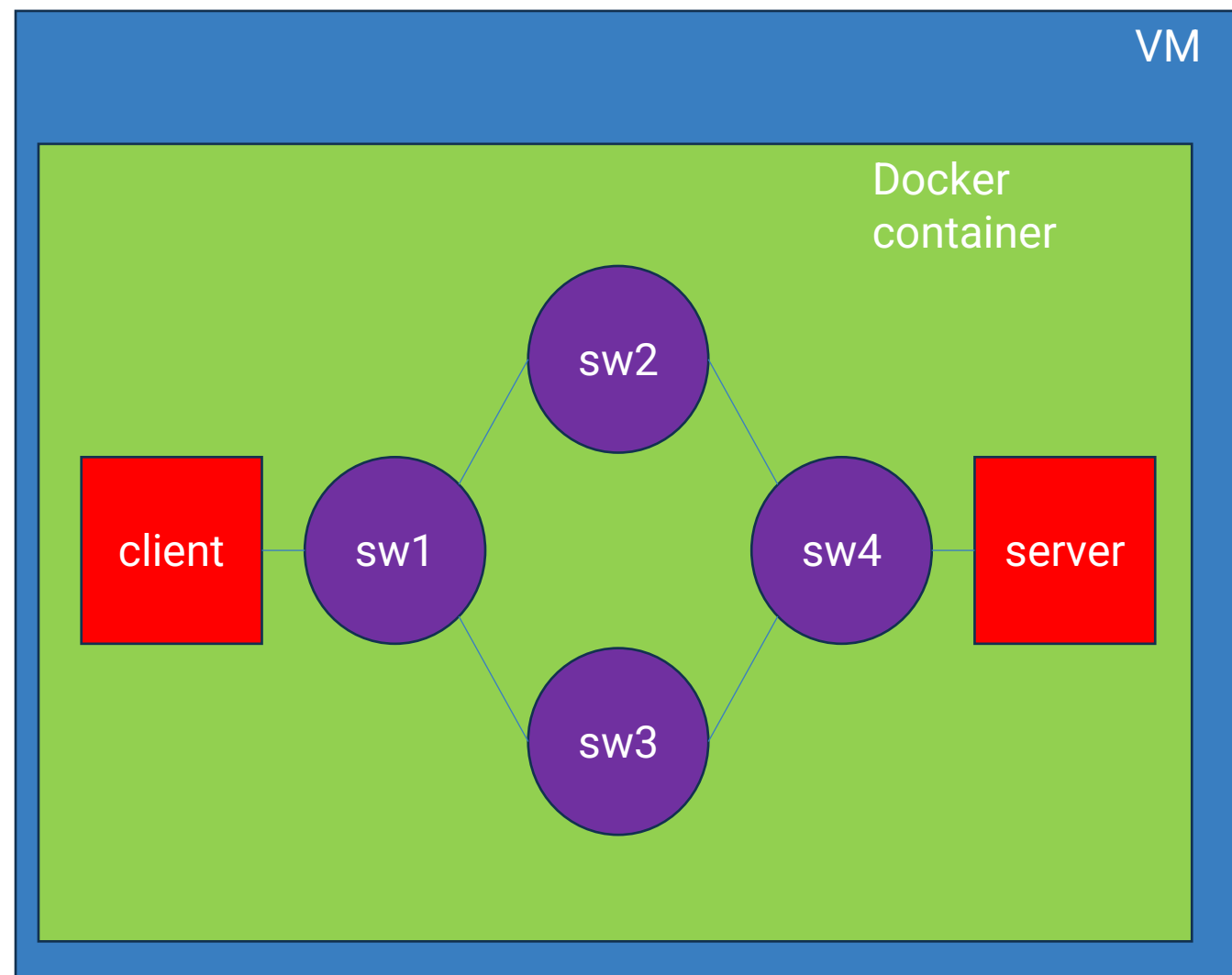
Mininet container

Mininet hosts and switches are just processes in different network namespaces

Mininet container

Communication through sockets

◉ The communication between them is done through unix sockets, with a shared file in the docker container

◉ By moving the file to a mounted volume, we can have the agent running in the VM

Program's structure

- **agent.rs**
  - Thread A: Listens to Context for service events and registers a corresponding KPI
  - Thread B: Waits for values from ping.rs and creates KPI values from them

- **ping.rs**
  - Runs a periodic ping and sends the collected value to the agent

Objective 1: Create a new monitoring probe in python

```
cd ~/controller/proto
```

Step 1 (ping)

◉ First, let's try to write the ping script in python
(Use python 2 syntax!)

◉ To begin, we can just ping 8.8.8.8 and print the value

◉ Then we will implement the socket communication
(exercise 3) and ping the correct IP (exercise 4)

Step 2a (agent)

- Try and implement the first Thread (A) that listens to Context events for service creation

  - Hint: Check context protobuf to see what method to use

- How are events differentiated?

  - Hint: Check context proto to see what kind of events we may have

- What should happen for each case?

Step 3 (agent)

◉ When we get an event for service creation, we should create a KPI to Monitoring

◉ We should find the appropriate method and create the appropriate object

  ◉ <u>Hint:</u> check monitoring protobuf!

Step 4 (agent and ping)

◉ Now it's time to implement the second Thread (B) that listens for values and sends them to monitoring

◉ First implement the socket part in both agent and ping to make sure that you can get a value on the agent

◉ Then implement sending the KPI value to Monitoring

　◉ Hint: protobuf files!

Objective 2: Measure packet loss, report to TFS, and create policy
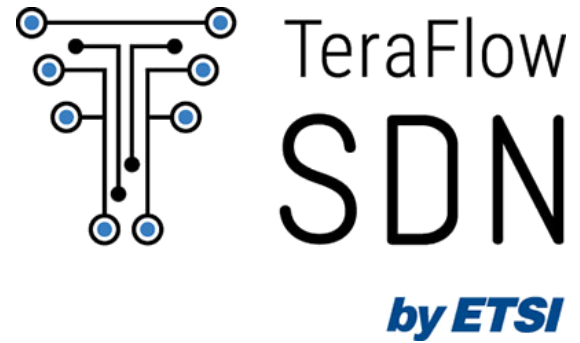
```
cd ~/controller/proto
```

Loss Ratio probe

◉ Now that we have created our Latency probe in Python, we can easily change it to measure other things

◉ For this exercise we are going to try and measure both Latency and Packet Loss

◉ Then we are going to change the Policy to use Packet Loss instead of Latency

TeraFlow
SDN

Loss Ratio probe

◉ To measure the Loss Ratio we can create a moving ratio, based on whether the ping failed or not

◉ Change the ping program to implement the above idea

New Policy

◉ To create a new Policy, we should first get the correct KPI_id from Monitoring

◉ Then, change the Policy message accordingly

◉ To insert Packet Loss we can again use:

◉ `tc qdisc add dev <switch-iface> root netem loss <x>%`

Thank you!

TFSsupport@etsi.org