PLUS/QUIC/TCP
VPP Middlebox

Tobias Bühler

Aberdeen, 12.06.2018

measurement and architecture for a middleboxed internet

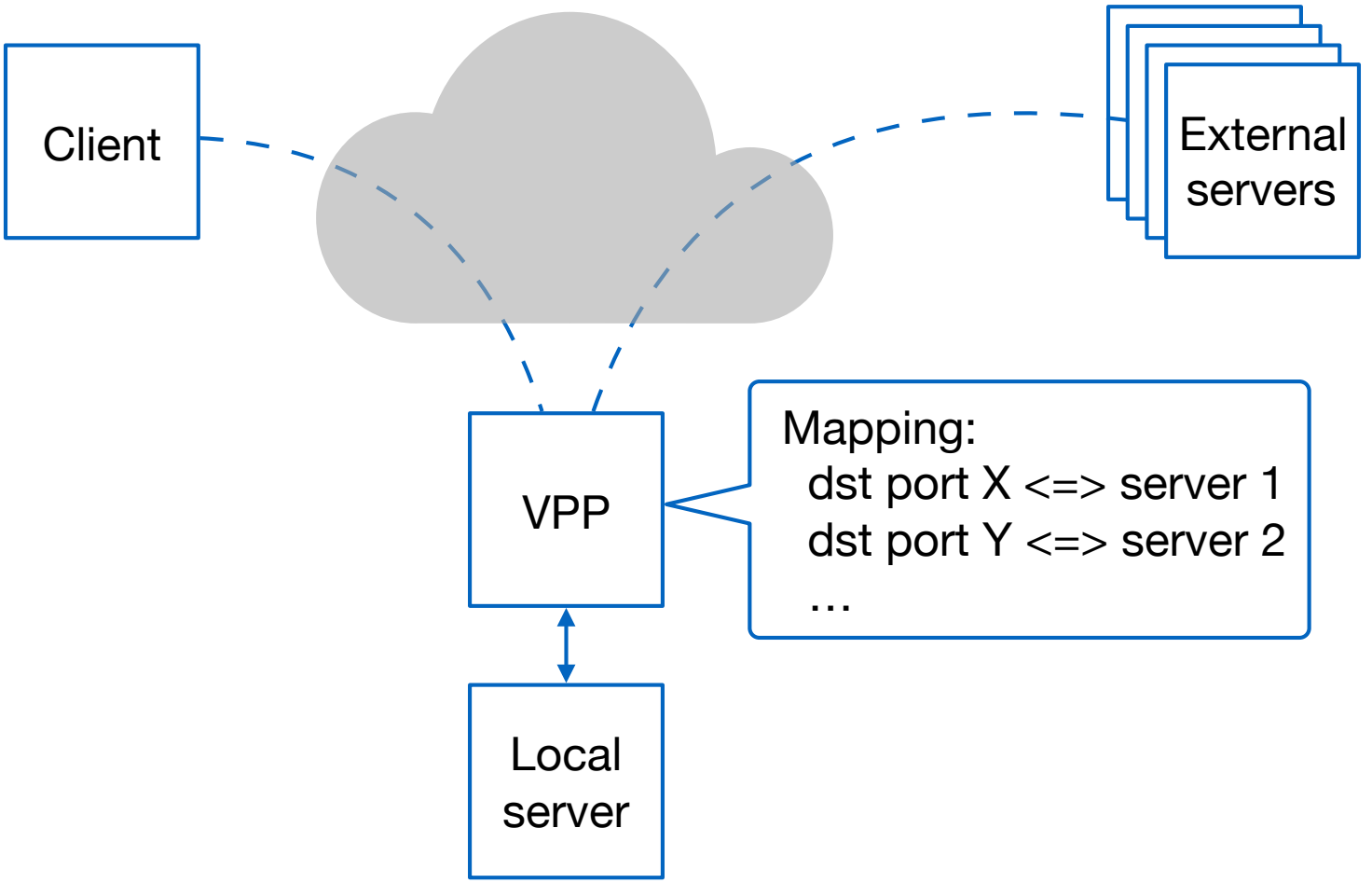
measurement | architecture | experimentation

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.

Presented during last Mami meeting

- PLUS middlebox implemented as VPP plugin
- Flows recognized by 5-tuple and CAT
- Full PLUS state machine
- RTT measurements based on PSN and PSE
- Support for extended headers
  - For example: „path accumulator“

Better setup



Better setup - tradeoffs

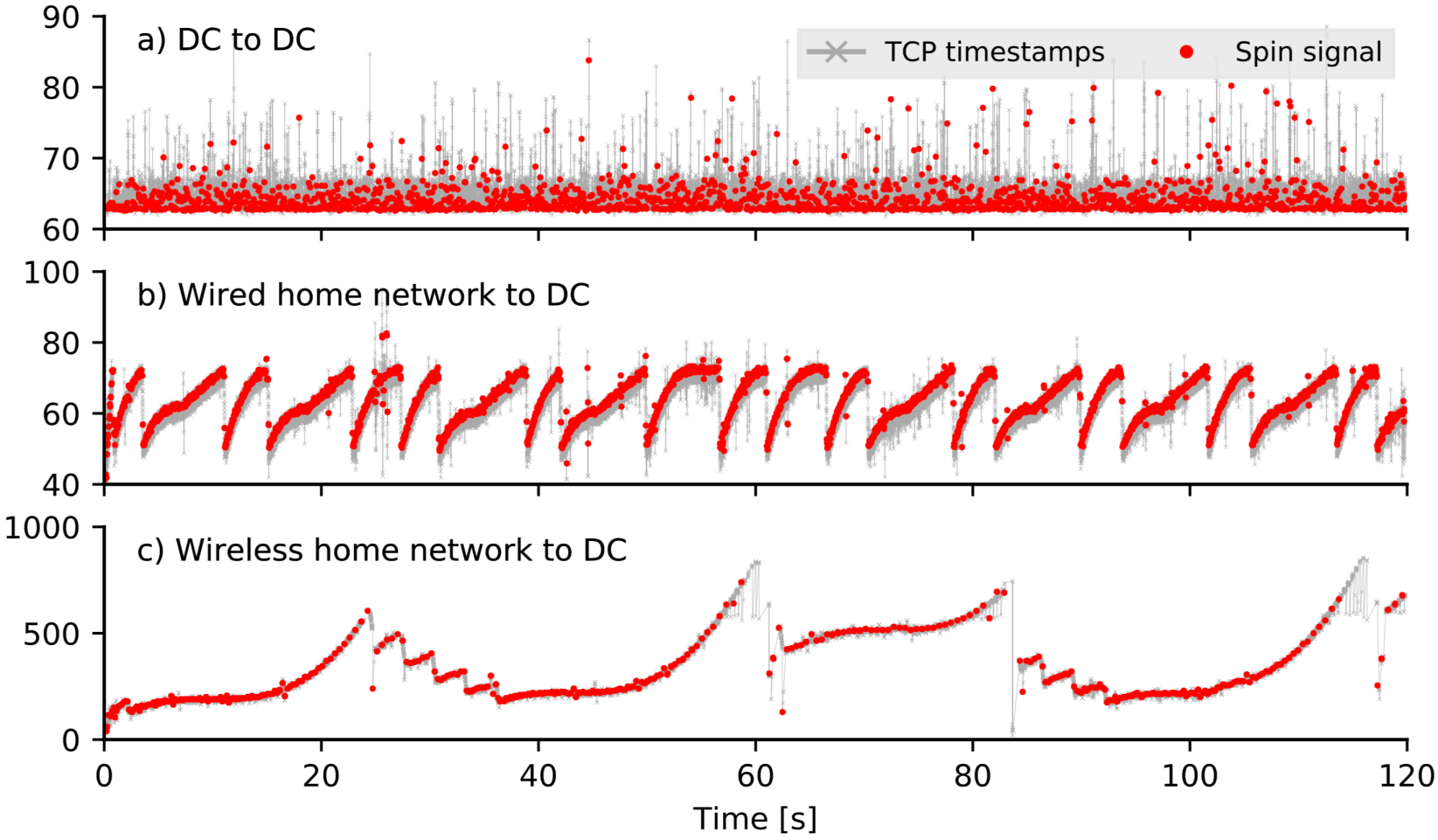
- For flows towards external servers:
  - VPP acts as a “true“ middlebox
  - There are two paths which could change/drop packets

- For flows towards the local server:
  - There are no uncontrollable path influences between the VPP box and the server
  - One half of the RTT estimation is always very short

Support for TCP and QUIC measurements

L4	Detection	State	Data	Measurement	Mod.	Forwarding
TCP		5-tuple	TS and VEC	RTT TS, RTT VEC		New dst IP based on dst port Save src IP for reverse packets Update checksums
UDP	QUIC port		Spin bit and VEC	RTT spin, RTT VEC, …		
	PLUS magic	5-tuple & CAT	PSN and PSE	RTT PSN/ PES	Extended headers	

Used for our spin bit IMC paper



Future implementation improvements

- In the process of abstracting key-parts of the pipeline
- Distribution into multiple VPP nodes
- Better CLI interface to easily:
  - Add new measurement servers
  - Track specific flows
- Customizable measurement output

Planned experiment setup

- Docker container deployed on MONROE
- ZHAW PLUS-go code as client/server
- Sending traffic towards the server behind our VPP box
- Results/measurements collected from the MONROE nodes, PLUS server and the VPP box

RTT measurements

- RTT measurements (using PSN/PSE) with comparison to estimations based on:
  - TCP timestamps
  - (TCP spin bit/VEC)
  - QUIC spin bit/VEC
  - PLUS spin bit/VEC
    - Currently not implemented, but could be added as extended header

Differences in RTT estimations

- **Spin bit**: max one RTT estimate per RTT
- **VEC**: max one RTT estimate per RTT
  but better guarantees on validity, reordering detection, …
- **TCP timestamps**: up to one estimate per packet,
  can be reduced to one estimate per RTT
- **PLUS PSN/PSE**: up to one estimate per packet
  PSN is guaranteed to increase with each packet

- **State**: PSN/PSE >= TS > VEC > spin bit

Additional experiments

- Various extended PLUS headers
- Support for moving (MONROE) end points
- …

VPP performance evaluation

- Per-packet processing time
- Memory usage during measurement campaign
- Use similar metrics as Korian/Justin (VPP-MB)
- Could be interesting numbers for the final demo
  - To discuss