Security Enablers for Future Networks

5G-ENSURE Task 3.5: Network Management and Virtualization Isolation

Felix Klaedtke
NEC Labs Europe, Heidelberg

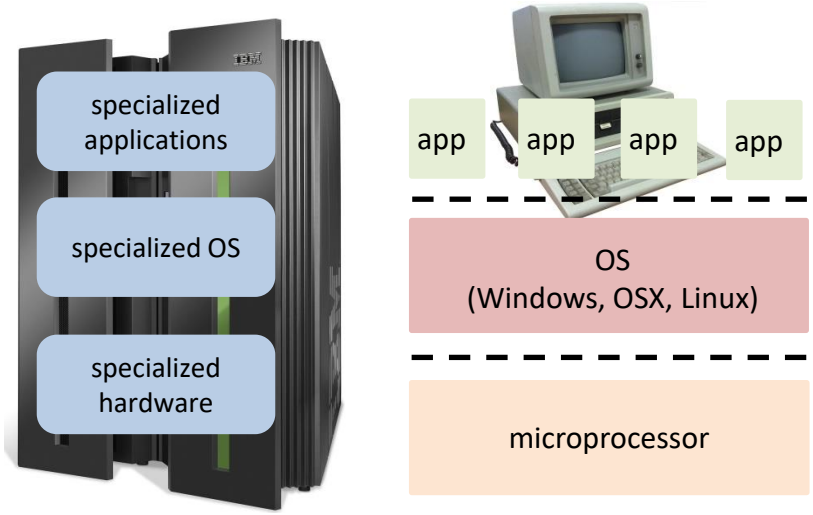
Sophia Antipolis, June 16, 2017

This work was carried out under the 5G-ENSURE project (www.5gensure.eu), which is funded by the European Union's Horizon 2020 research and innovation programme under the grant agreement number 671562. Responsibility for the information and views set out in this presentation lies entirely with the authors.

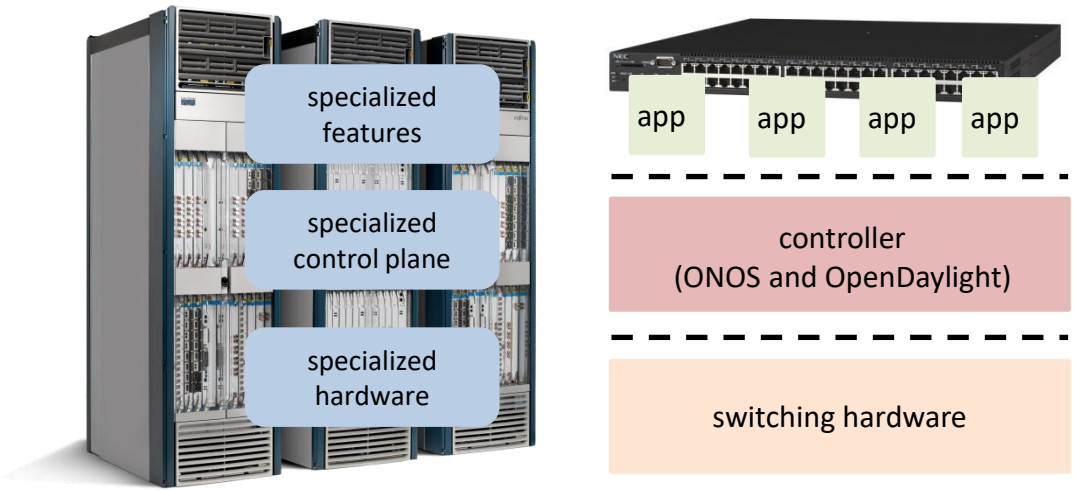
“Software is eating the world”

—Marc Andreessen
The Wall Street Journal, 2011

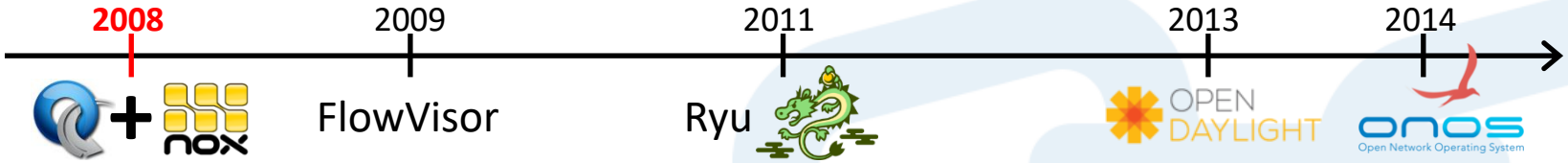
In the ‘60s and ‘70s:



Still true today (mostly):



- Since 2008: software-defined networking (SDN)



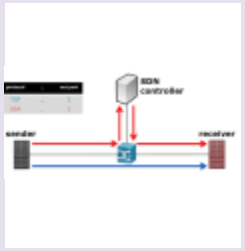
- Adoption of SDN principles in 5G (*network softwarization*)

SDN Promises

- SDN provides the means to make networks **cheaper**, **richer**, **more reliable**, ...
  - Less specialized hardware
  - Standardized APIs
  - Network abstractions
  - Virtualized network functions
  - Etc.
- ... and also **more secure**.
- However, SDN introduces new threats.
  - Software is inherently buggy
  - New attacks
  - Etc.

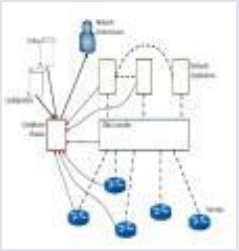
5G-ENSURE Security Enablers in Task 3.5

Fingerprinting SDN networks



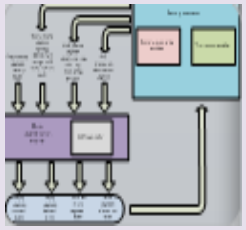
Preventing information leakage about switch configurations in SDN networks

Component interactions



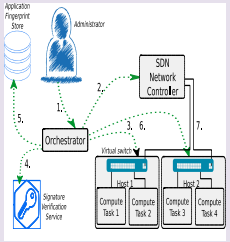
Checking policy compliance of interactions between network components

Micro-segmentation



Managing segments in a 5G network in which strict security policies can be enforced

Bootstrapping trust

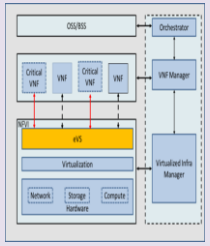


Verifying integrity of software on network edge prior to enrollment into SDN deployments

Access control

Controlling access of network applications to network resources

Flow control for critical VNF

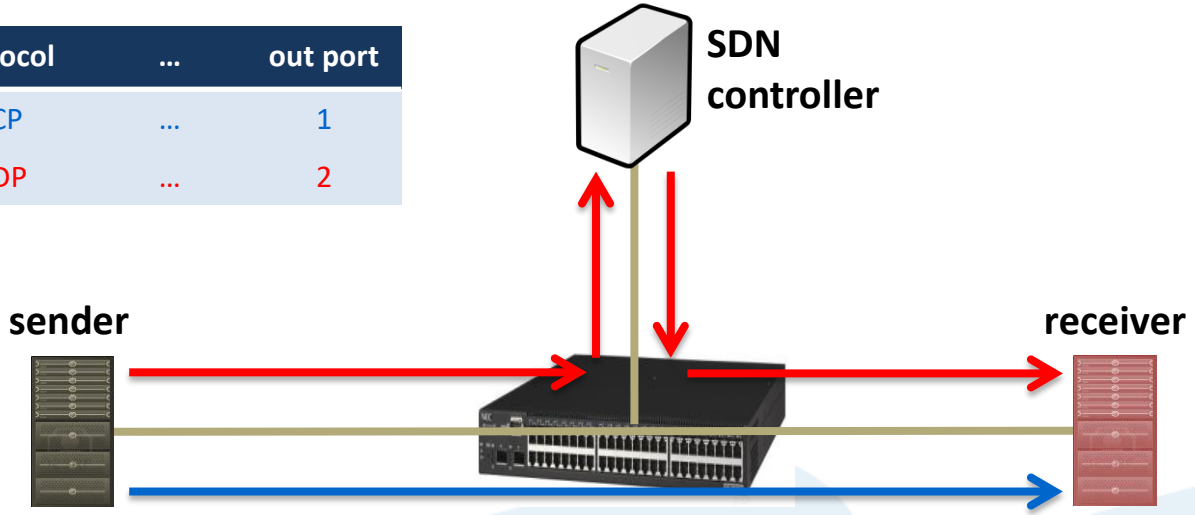


Protecting critical VNFs at runtime from malicious threats through in-network detection and mitigation

Observation and Motivation

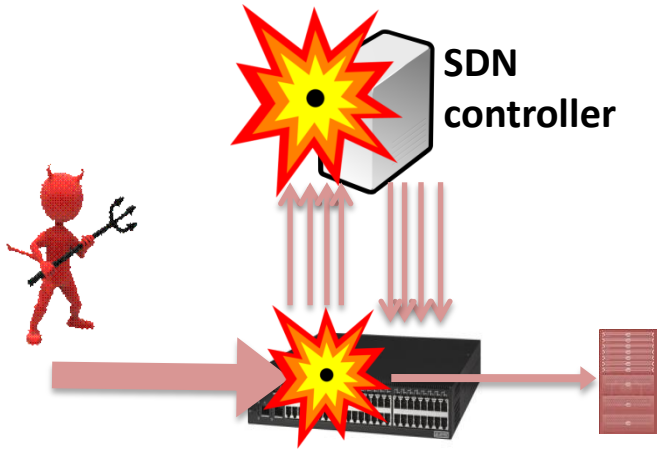
- Packets are processed much faster at the ***data plane*** than on the ***control plane***

SRC	DST	protocol	...	out port
*	R	TCP	...	1
*	R	UDP	...	2



- An attacker can measure the processing times of packets
- Information leakage about the network's control logic

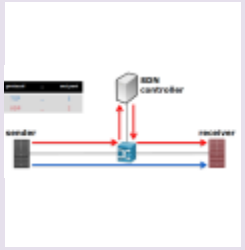
Exploitation



- Knowledge about the controller-switch interactions empowers an attacker to launch powerful DoS attacks
  - Overload the controller (e.g., too many packet-in messages)
  - Overload the switch (e.g., fill TCAMs)
  - …
- Fingerprinting the network can also be exploited for rule scanning
- Our experiments provide evidence that fingerprinting an SDN network is feasible!
  - High accuracy (>95%)
  - Even passive attackers can fingerprint SDN networks
  - Number of hardware switches has minor impact
  - Feasible even in the presence of software switches
  - Countermeasure through obfuscation: mimic controller-switch interactions for flows that were interactive for some time

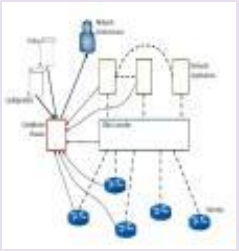
5G-ENSURE Security Enablers in Task 3.5

Fingerprinting SDN networks



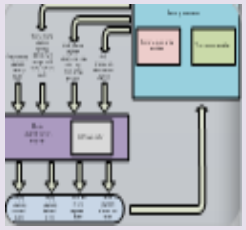
Preventing information leakage about switch configurations in SDN networks

Component interactions



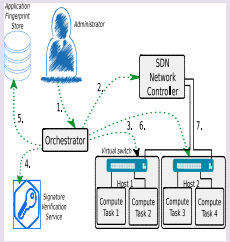
Checking policy compliance of interactions between network components

Micro-segmentation



Managing segments in a 5G network in which strict security policies can be enforced

Bootstrapping trust

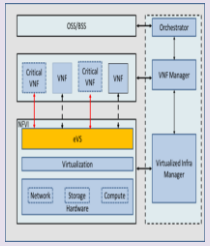


Verifying integrity of software on network edge prior to enrollment into SDN deployments

Access control

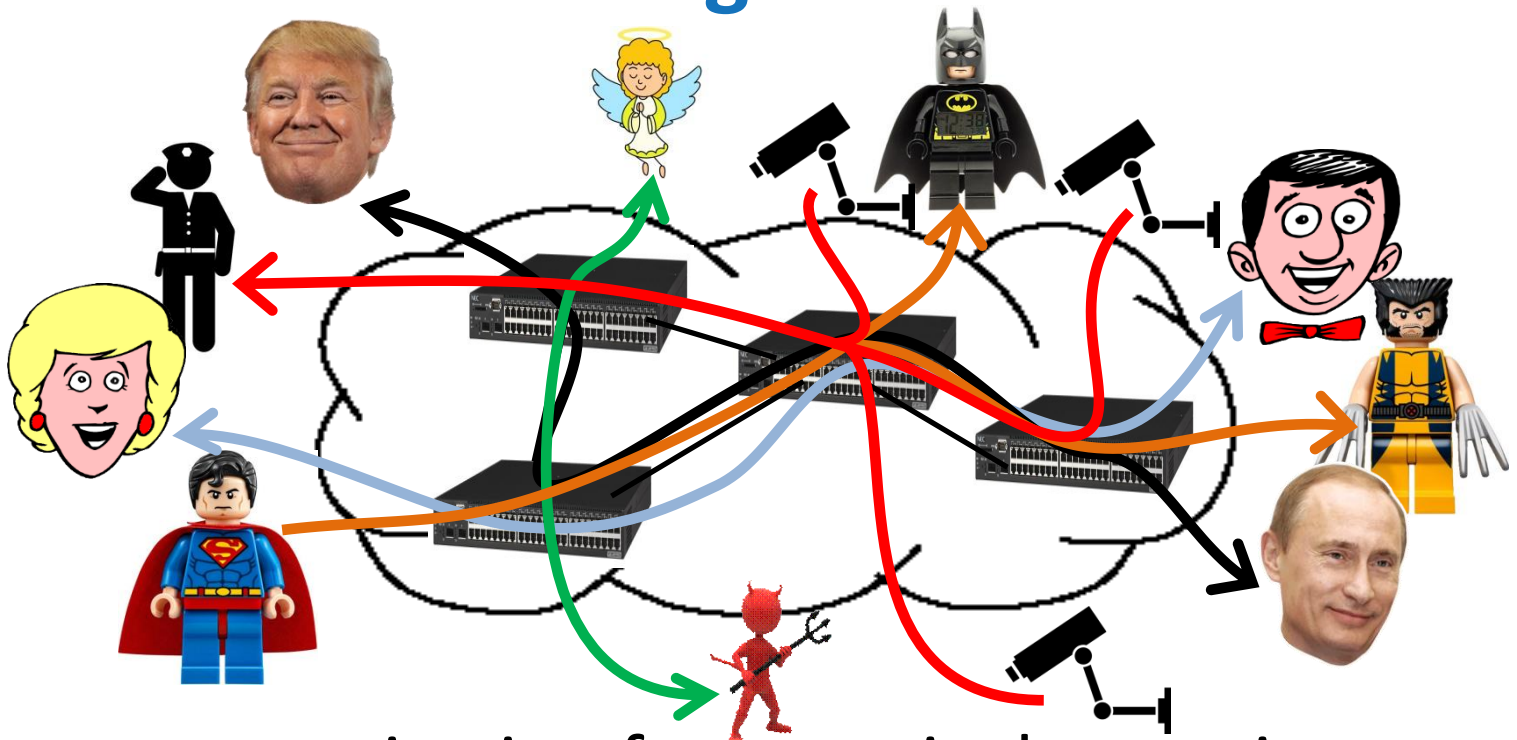
Controlling access of network applications to network resources

Flow control for critical VNF



Protecting critical VNFs at runtime from malicious threats through in-network detection and mitigation

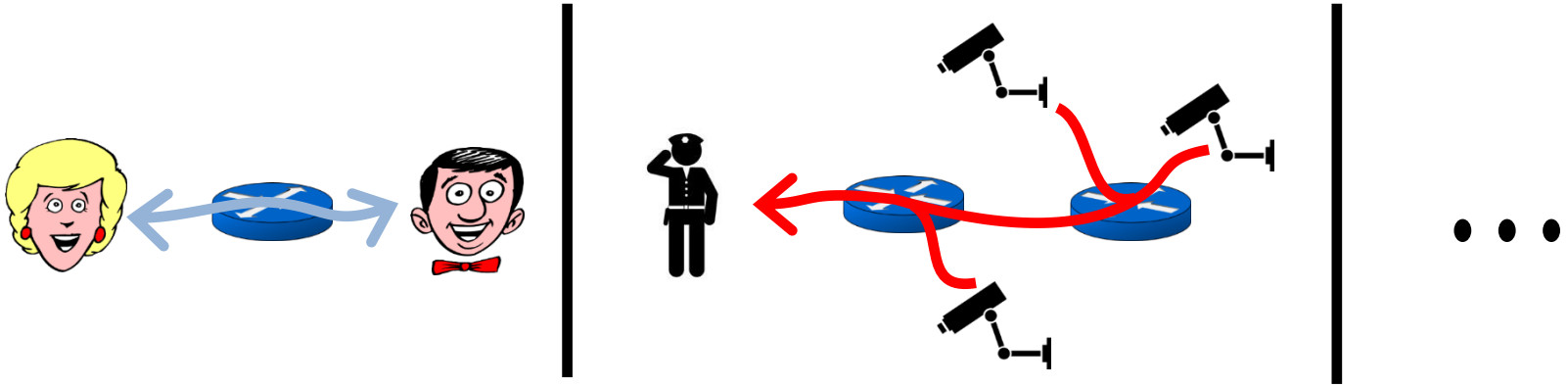
Micro-segmentation



Group communication for a particular service, user group, ...

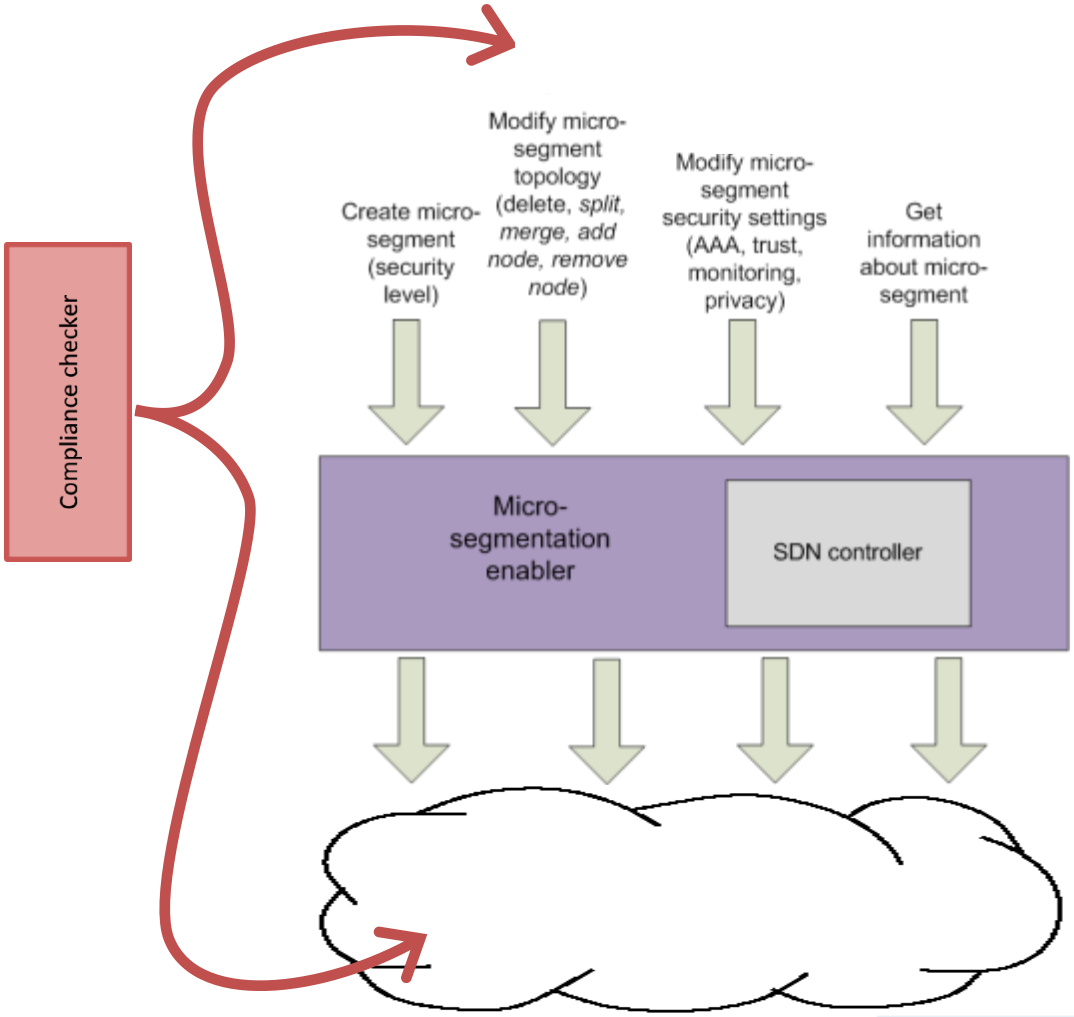
- Segments are highlevel, isolated, and customizable
- Segments share some of the physical infrastructure

Micro-segments



Video here

Micro-segmentation Enabler and Others

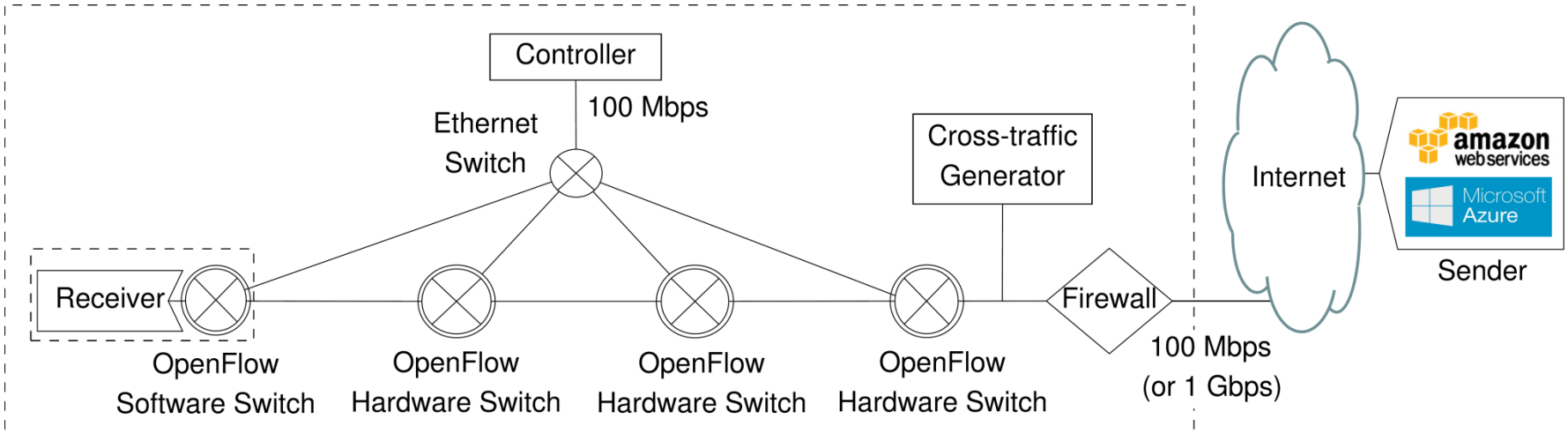


References

- www.5gensure.eu
- H. Cui, G.O. Karame, F. Klaedtke, and R. Bifulco
  On the fingerprinting of software-define networks
  IEEE Transactions on Information Forensics and Security, 2016
- O. Mämmelä, J. Hiltunen, J. Suomalainen, K. Ahola, P. Mannersalo, and J. Vehkaperä
  Towards micro-segmentation in 5G network security
  EuCNC workshop, 2016
- D. Basin, F. Klaedtke, and E. Zalinescu
  Runtime verification of temporal properties over out-of-order data streams
  29th Conference on Computer Aided Verification (CAV), 2017 (to appear)

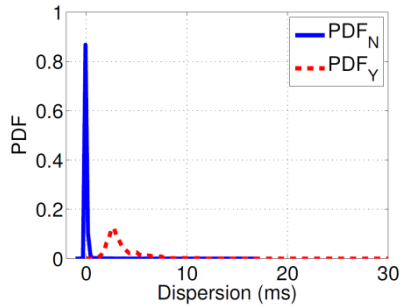
ADDITIONAL DETAILS

Experiments

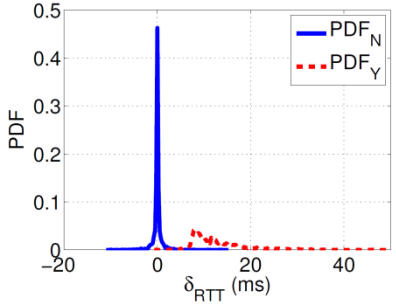


- Probe: sender → Internet → firewall → switches → receiver
  - Multiple sender locations around the globe
  - Measurements conducted over several months
- Time-based features measured at the sender
  1. Dispersion
  2. Round-trip time (RTT)

Results



(a) $k = 3$ hardware switches



(a) $k = 3$ hardware switches, time span 1 second

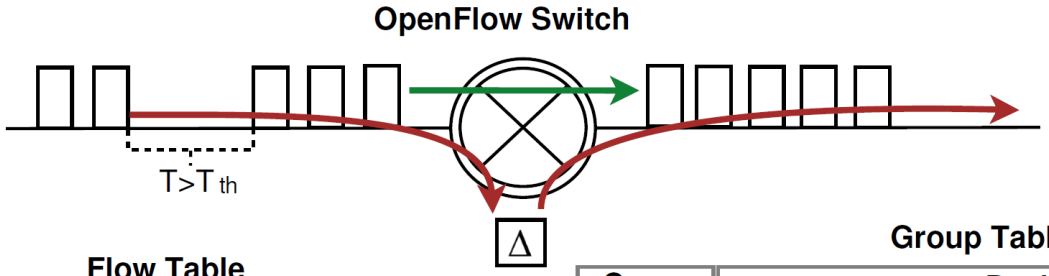
- PDF_N: packet does not trigger rule installation
- PDF_Y: packet triggers rule installation
- Distributions (PDF_N and PDF_Y) differ significantly

- Dispersion:
  - Stable over time
  - Less affected by network size
- Delta-RTT:
  - Less stable over time
  - Can be extracted from passive measurements

- Experiments provide evidence that fingerprinting an SDN network is feasible
  - With high accuracy (>95%)
  - Even passive attackers can fingerprint SDN networks
  - Number of hardware switches has minor impact
  - Fingerprinting remains feasible even in the presence of software switches

Countermeasure

- Control plane cannot be made significantly faster (ns instead of ms)
- Make processing times for packets indistinguishable
  - Delay matching packets at a switch before forwarding them
  - Severely harms network performance
- Delay the first few packets of “old” flows
  - The delay can be determined from our observations
  - Obscure attacker whether additional delay is caused by “controller-switch” interaction or our countermeasure
  - No overhead for control plane, minor impact on network performance, and effective



Flow Table

Match set	Action set
Destination IP = 1.2.3.4	Group (3)
*	To Controller
....	....

Group Table

Group Identifier	Buckets	
	Description	Action set
3	Active flows bucket	Forward to port 2
	Inactive flows bucket	Delay and forward to port 2
	....	....