Limits of Isolation

perspectives on device security in (5G) network slicing

Lorenzo Di Gregorio

CODE Workshop – Software-Defined Networks Security

Munich – July 11, 2018

It’s the law!

DISCLAIMER AND LEGAL INFORMATION

All opinions expressed in this document are those of the author individually and are not reflective or indicative of the opinions and positions of the author's employer.

The technology described in this document is or could be under development and is being presented solely for the purpose of soliciting feedback. The content and any information in this presentation shall in no way be regarded as a warranty or guarantee of conditions of characteristics.

This presentation reflects the current state of the subject matter and may unilaterally be changed by Intel Corporation and/or its affiliated companies (hereinafter referred to as “Intel") at any time. Unless otherwise formally agreed with Intel, Intel assumes no warranties or liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party with respect to the content and information given in this presentation.

© 2018 by Intel Deutschland GmbH

Abstract

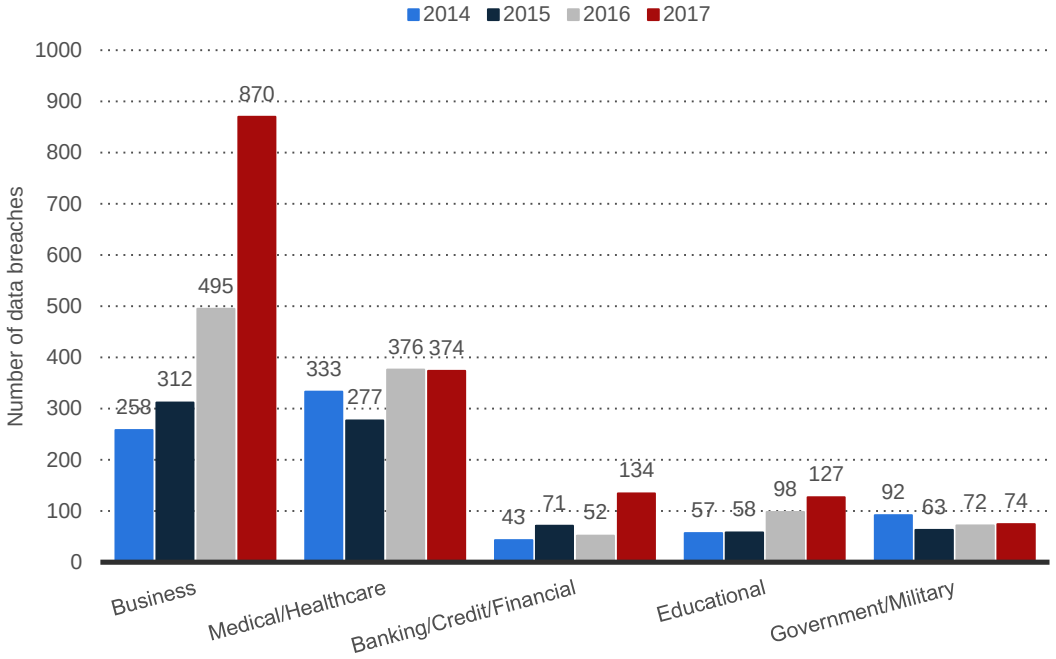
5G evangelists promote dynamic network slicing to support multi-tenancy proliferation of new services: a key requirement for these slices is isolation.

However, coexistence of slices on common hardware poses practical limits to the levels of isolation which can be achieved.

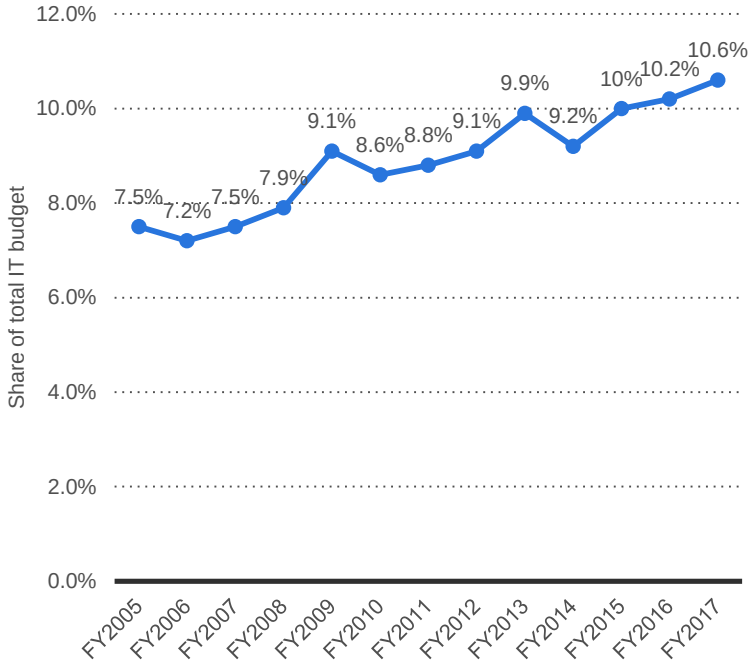
This talk shall quickly give some perspectives on security challenges onto the development of new devices in this novel evolving ecosystem.

Security attacks target predominantly business

product budgets for security must remain below 10%



Source: Identity Theft Resource Center; CyberScout ID 273572



Source: Ponemon Institute; Thales Group ID 536764

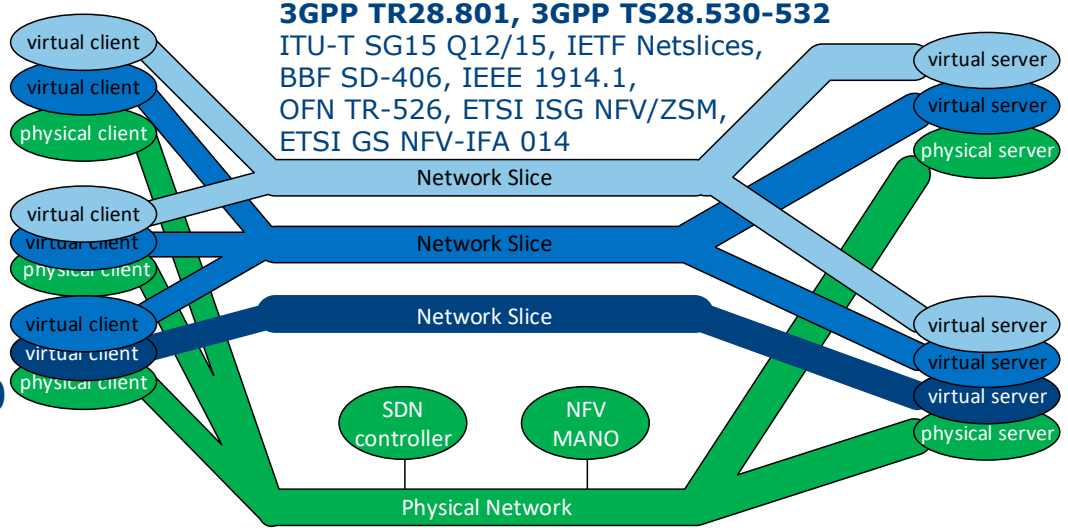
Network Slicing – a new network virtualization in 5G

"Welcome back, my friends / to the show that never ends"

Network Slice: managed group of subsets of resources, typically via SDN/NFV

Slice Instance: activated slice created as logical virtual network from a template

Tenant: user of instances with some management capabilities



Standardization

3GPP Release 15 on 2018-06-19

On security, see TR 33.811 V1.0.0

"Study on security aspects of 5G network slicing management"

Some general principles on SDN/NFV attack surfaces

timing

under best effort for low latency, traffic timing must leak some information: concurrent users, cached content, network topology, amplification sources ...

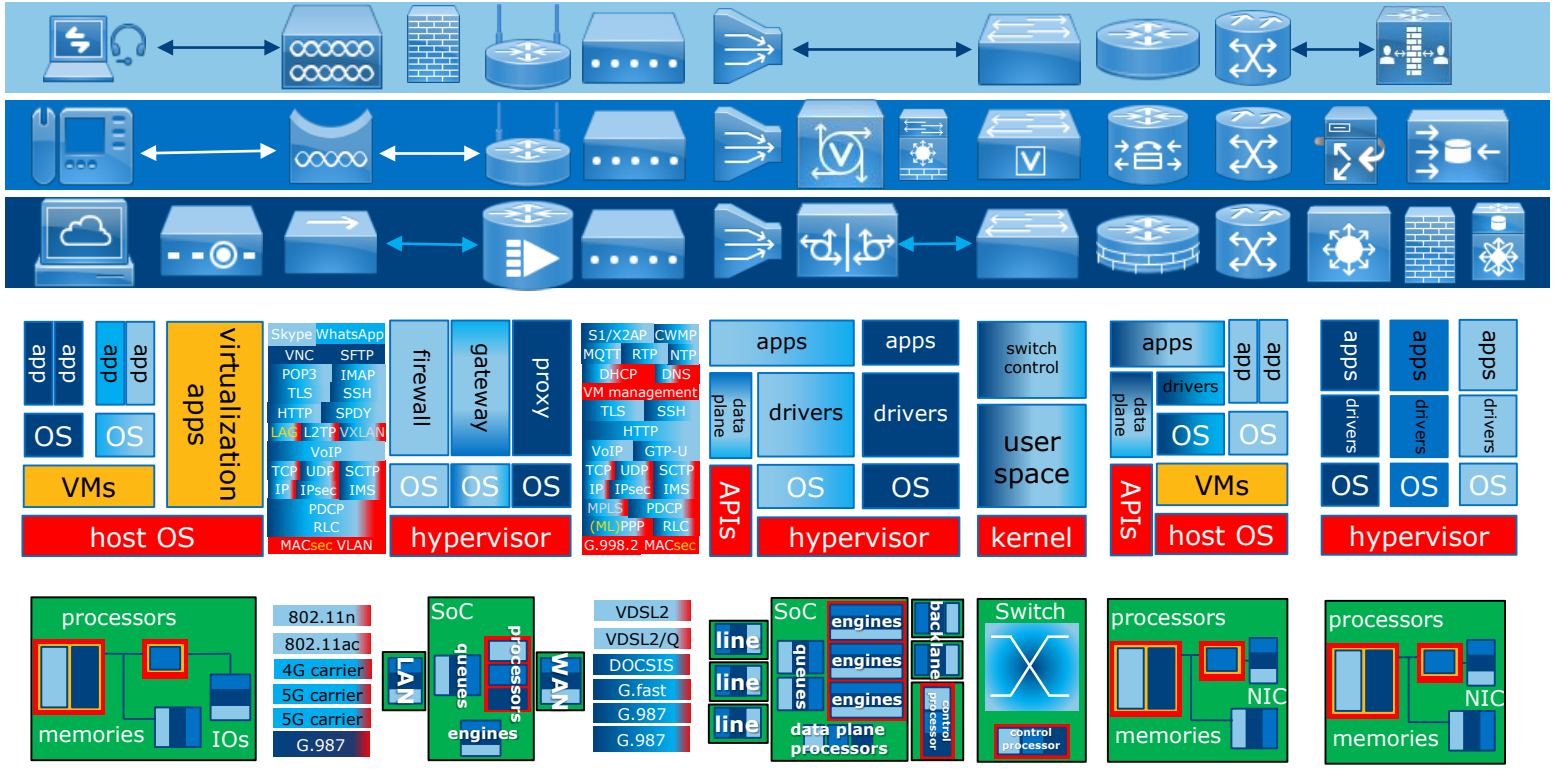
coherency

propagation delay of provisioning commands might lead to illegal intermediate network states and make connectivity appear across isolated domains

segmentation

multiple isolated trust domains might sustain a common domain in an upper layer, and conversely, so that vertical pivoting (e.g. APIs) turns horizontal (slices)

The illusion of fully isolated network slices



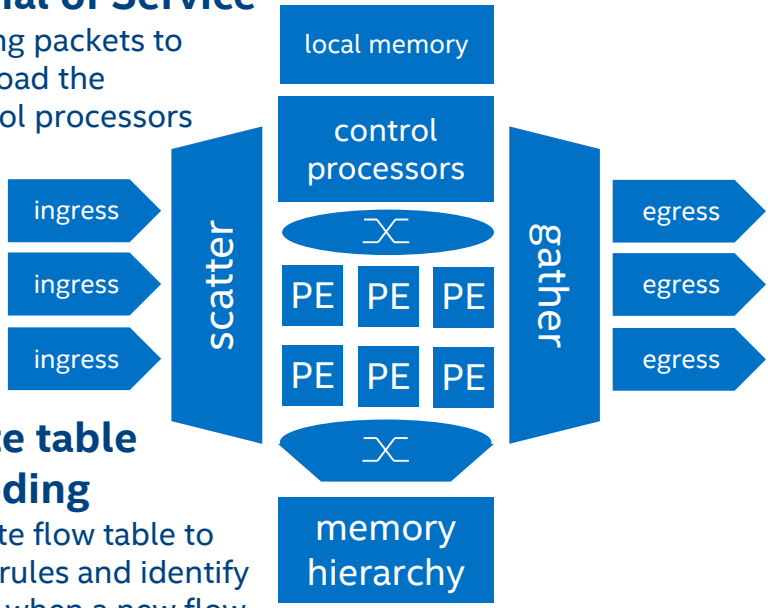
"False face must hide what the false heart doth know"

Examples of timing attacks within devices

data within one device could be exposed among slices

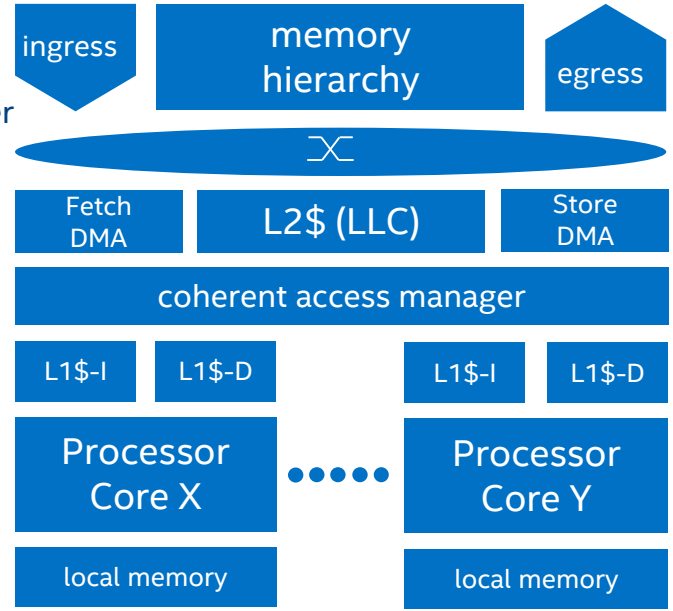
Denial of Service
forging packets to overload the control processors

State table flooding
pollute flow table to evict rules and identify them when a new flow hits with high latency



Caching
Invalidate+Transfer
Prime+Probe
Flush+Reload
Evict+Reload
Flush+Flush
Evict+Time
Cache Games
CacheBleed
Meltdown
Spectre

Processing
Montgomery's reduction and ladder attacks



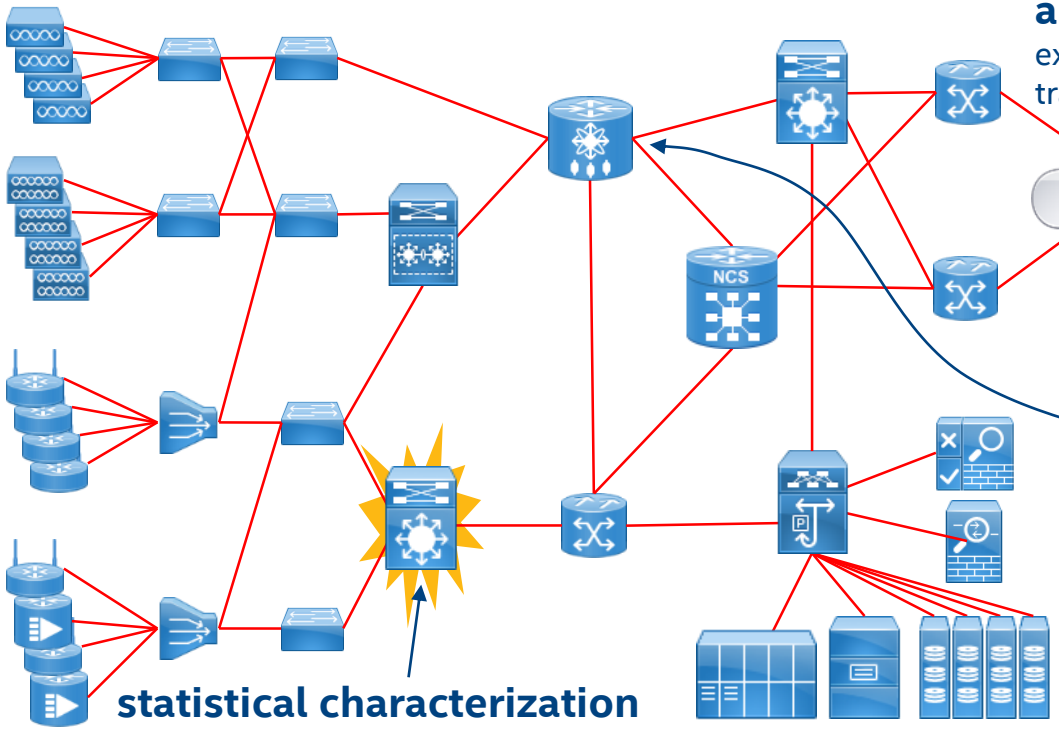
Data Plane Offloading

Symmetric Multiprocessing

Examples of timing attacks in networks

traffic in one slice is characterizable from another slice

network topology inference
measure end-to-end latencies to identify major network structures



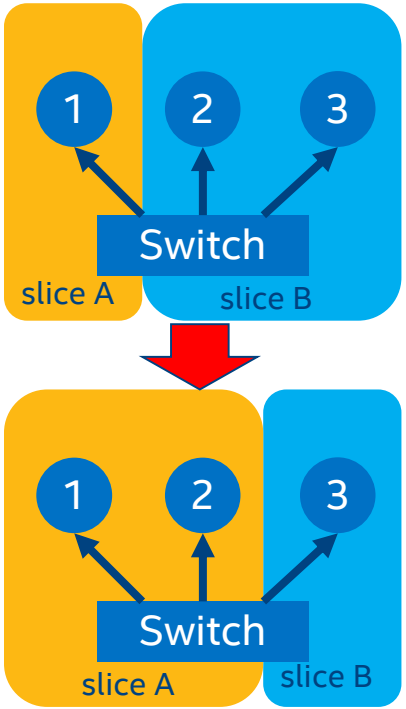
amplification attack
exploit some VNF to create high traffic (no DoS) toward a target

flooding attack
congestion and flapping routes cause temporary divertion of user traffic through other penetrated nodes

statistical characterization
estimate numer of users, accessed content (cache latencíes) and services (headers)

Need for "happens-before" in provisioning updates

propagation delays open leaks or cause DoS across slices



Load balancing example

"change you can't believe in"

Slice A sees high traffic, we must move ② from B to A

Reconfigure ② first

Some traffic in B dropped, balancing is faulty → DoS attack is possible

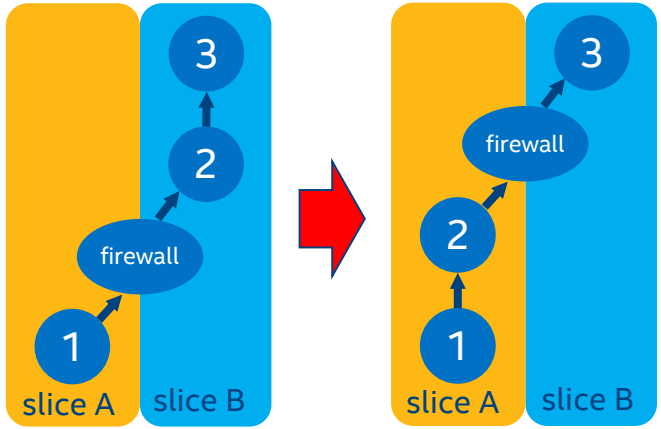
Reconfigure Switch first

Slice A leaks into slice B, illegal connection → security breach

Fence and drain ②, then reconfigure and reopen

OK, but high latency in balancing

Adapted from "Consistent updates for software-defined networks: change you can believe in!" In Proceedings of the 10th ACM Workshop on Hot Topics in Networks (HotNets-X)



Partitioning example

A VNF on ② must move into slice A.

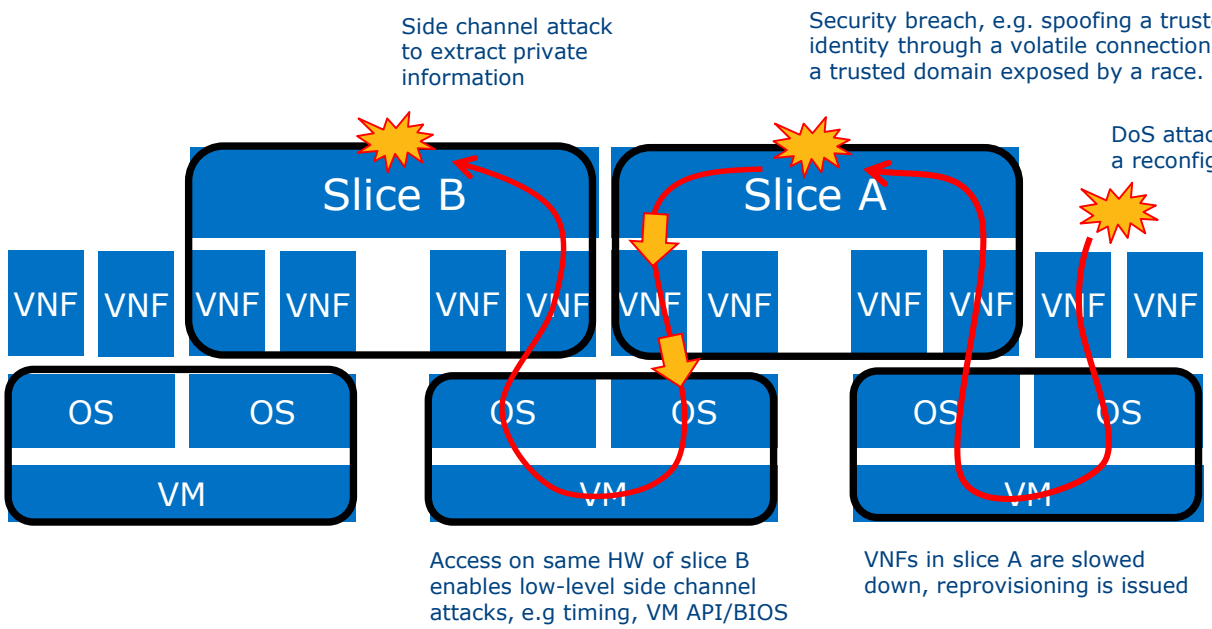
No secure sequence of switch operations exists without bypassing the firewall.

Traffic must be fenced and drained

Adapted from "Safe Update of Hybrid SDN Networks" In IEEE/ACM Trans. Netw. 25, 3 (June 2017)

Concept for horizontal move of vertical pivoting

a breach in one API enables a breach into another slice



Races

change or delay the order of events

TOCTOU
(time of check to time of use)

privilege escalation, delayed authorization

Topology poisoning

spoof identity into trust area when connection appears

Typical vulnerabilities

non-initialzed storage, unchecked inputs, side-channel attacks

Prevention in a nutshell

Timing decorrelation

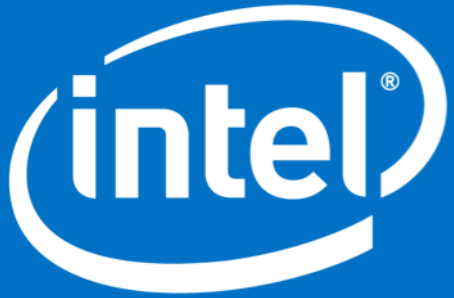
Present outputs and side effects in uncorrelated (constant) time, e.g. queuing
Suppress side effects by partitioning/fencing resources, e.g. cache, bandwidth
Implement graceful degradation and failsafe measures for overload conditions

Race prevention

Obvious: fuzz testing agaist DoS, taint analyses and other hardening measures
Large state tables for coexistence of tagged transitions, preventing illegal states
Components for model checking, with precedence rules and deadlock recovery

Segregation

Conventional: atomic access, fast admission control, trusted boot and execution
Coexistence of many nested trust domains on device, tracking source identities



Intel Communication and Devices Group