S-NIC: Securing NIC-Accelerated Network Functions in the Cloud

Yang Zhou, **Mark Wilkening**, James Mickens, Minlan Yu

Harvard University

Network Functions (NFs) are popular

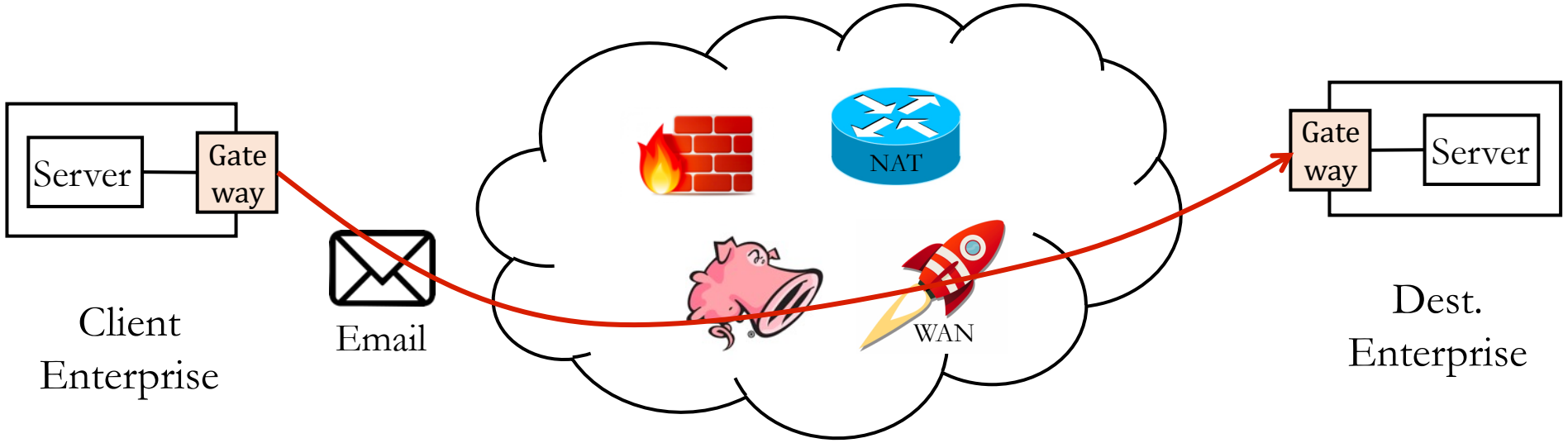
Network Functions (NFs) are critical infrastructure in networked systems.

- Firewall, NATs, WAN optimizers, deep packet inspection (DPI), etc.

Snort3 DPI

NF Outsourcing to the Cloud

- Scale on demand as the traffic load changes → Cheaper!
- Easier to manage, upgrade, and deploy. Outsourcing these tasks

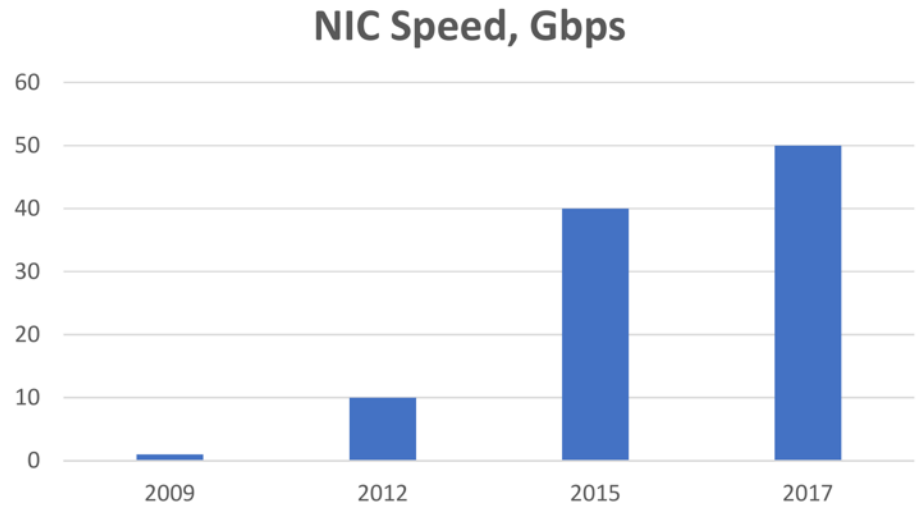


Two key requirements: Performance and Security

Requirement: High Performance

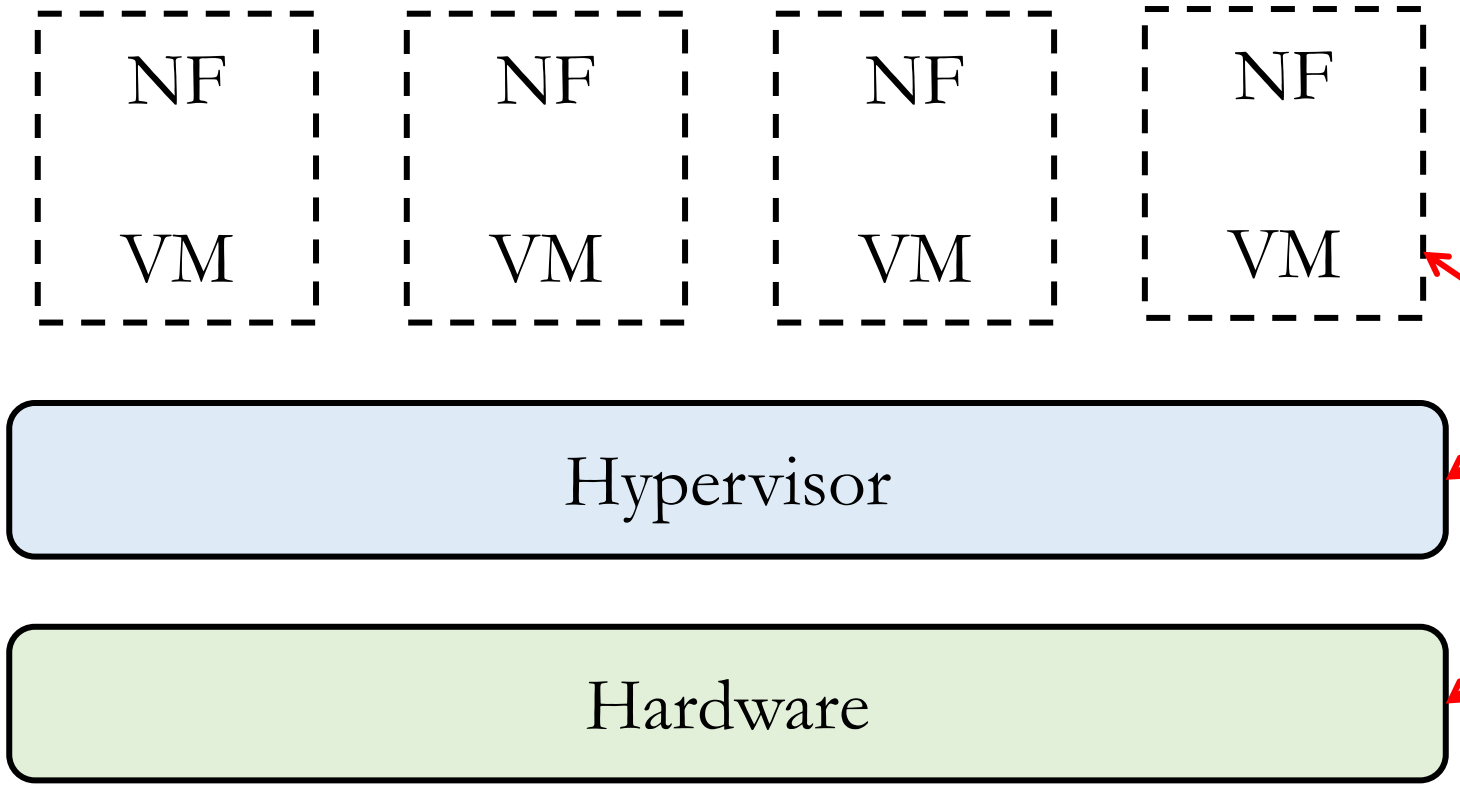
Achieving high performance is important but challenging.

- High performance to keep up with link speed (10/40/100Gbps) as traffic grows.
- Microsoft AccelNet [NSDI'18]:



"We got a 50x improvement in network throughput,
but not a 50x improvement in CPU power!"

Today: Running NFs on VMs



Low performance:

Firewall = 0.66Gbps/core

(NetBricks [OSDI'16])

+

Virtualization Overhead

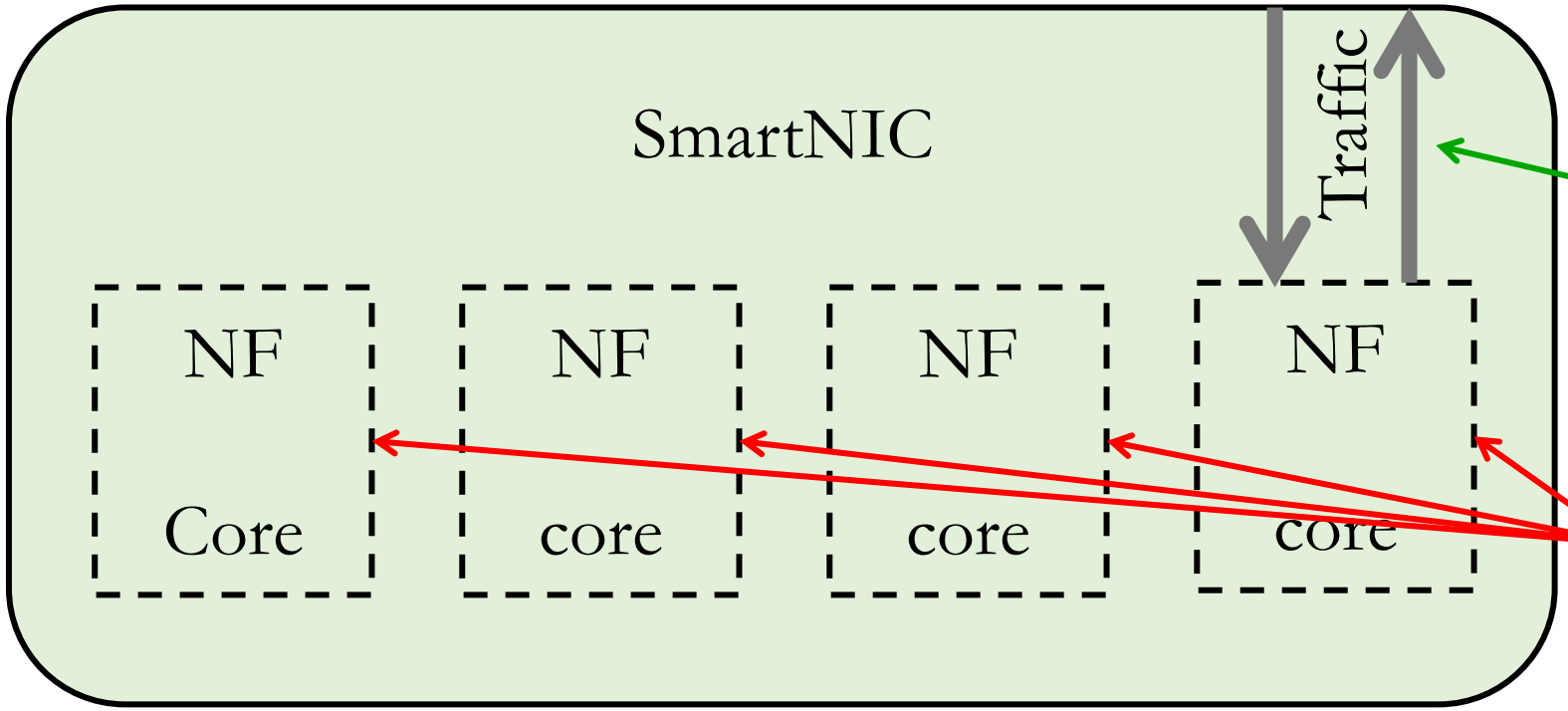
Not secure:

Even Intel SGX still suffers from

An Alternative: Running NFs on Smart NICs

Azure is offloading their full cloud storage stack to multi-core Smart NICs[1].

Available NICs: Mellanox BlueField, Broadcom Stingray, Marvell LiquidIO, etc.



Good efficiency:

- Cost-efficient RISC cores
- ASIC-based accelerators
- No PCIe traversing

Bad Security:

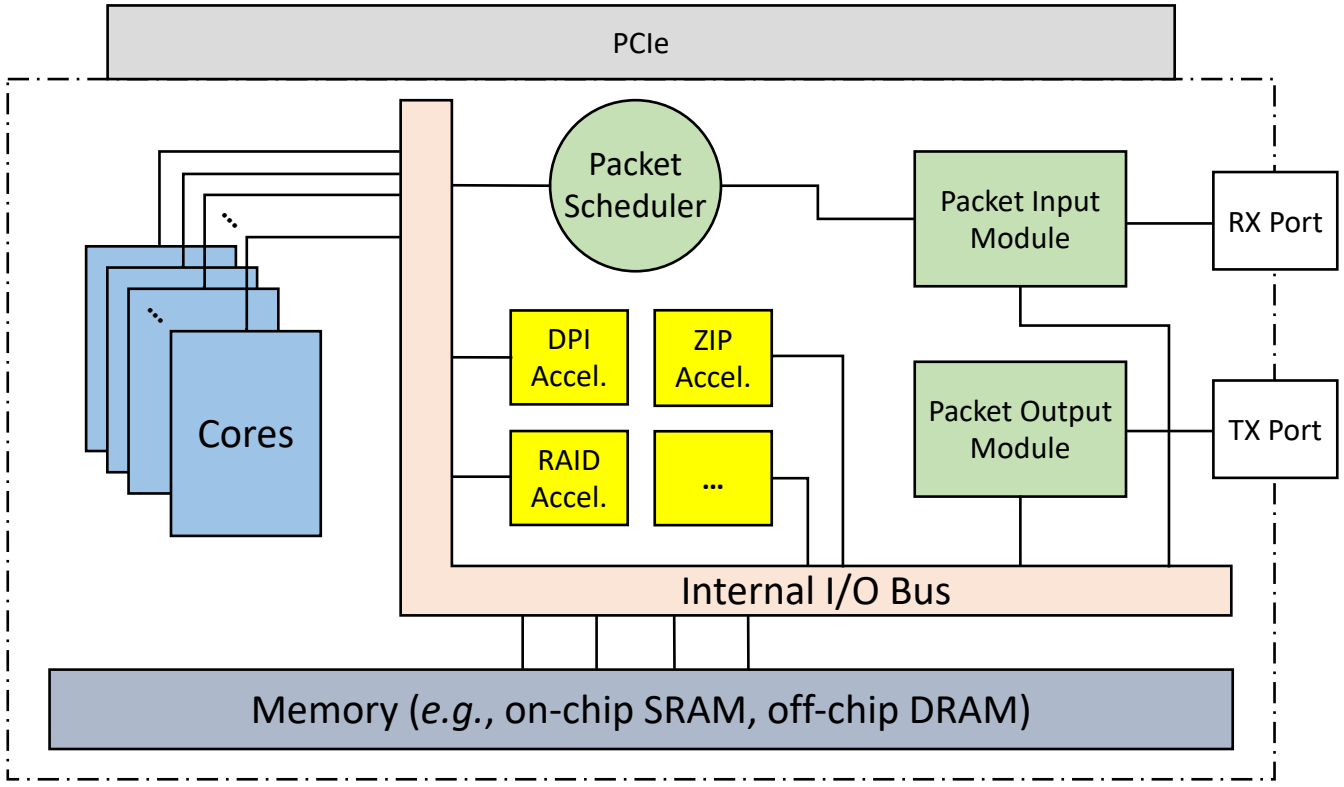
- No mem. isolation
- No perf. isolation

[1] Li, Huaicheng, et al. "LeapIO: Efficient and Portable Virtual NVMe Storage on ARM SoCs." ASPLOS 2020.

S-NIC

Redesign Smart NICs to provide high performance and strong security for network functions

Background on (Multi-Core) Smart NICs



Memory: DRAM, SRAM

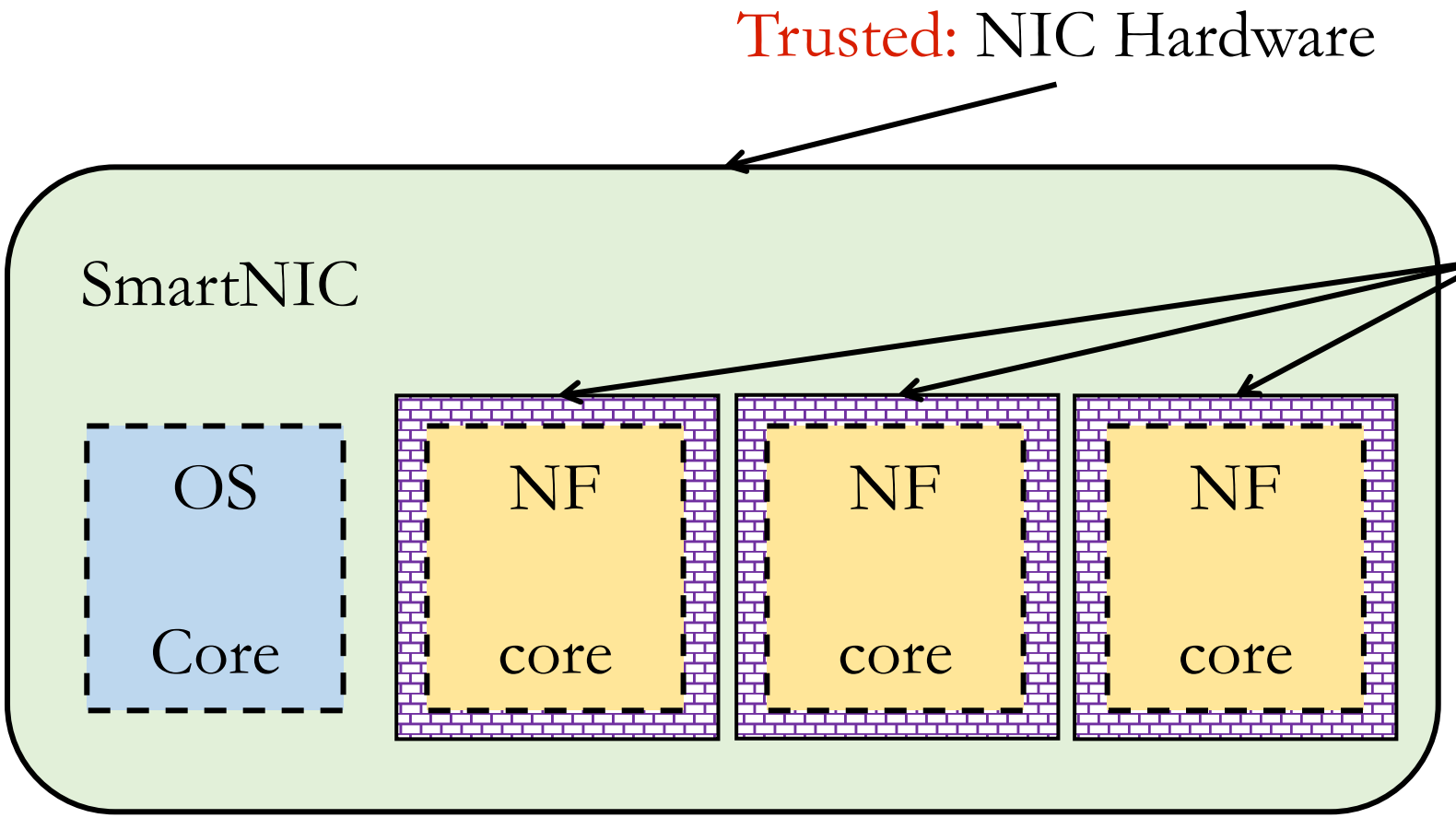
Hardware accelerators

Packet ingress and egress

IO bus

Smart NICs are not designed for shared environment

Threat Model



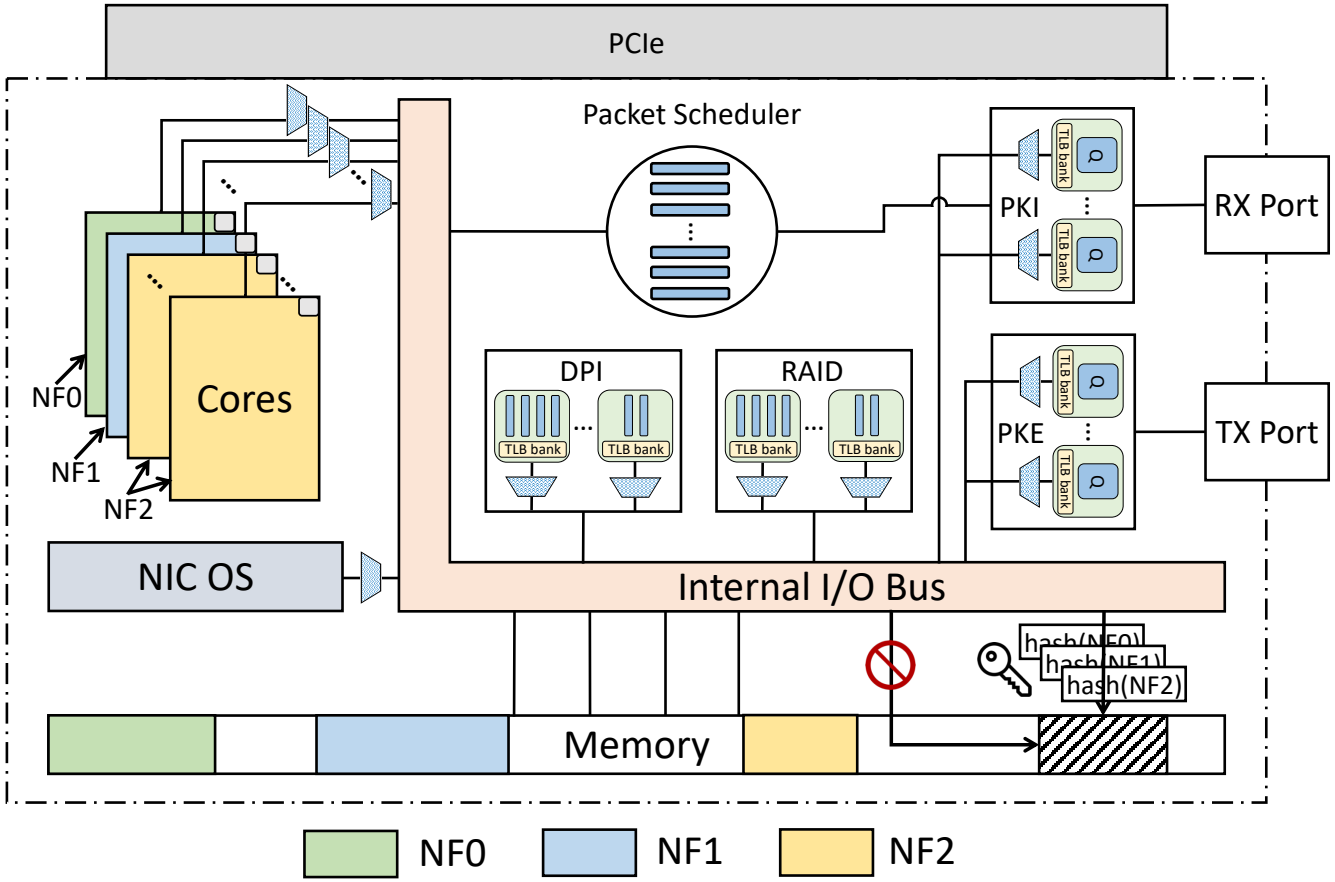
Goals:

1) Isolation
2) Side-channel-free (shared HW states)

Out of scope:

1) Physical possession attacks
2) DoS attacks from Cloud

S-NIC: Key Designs



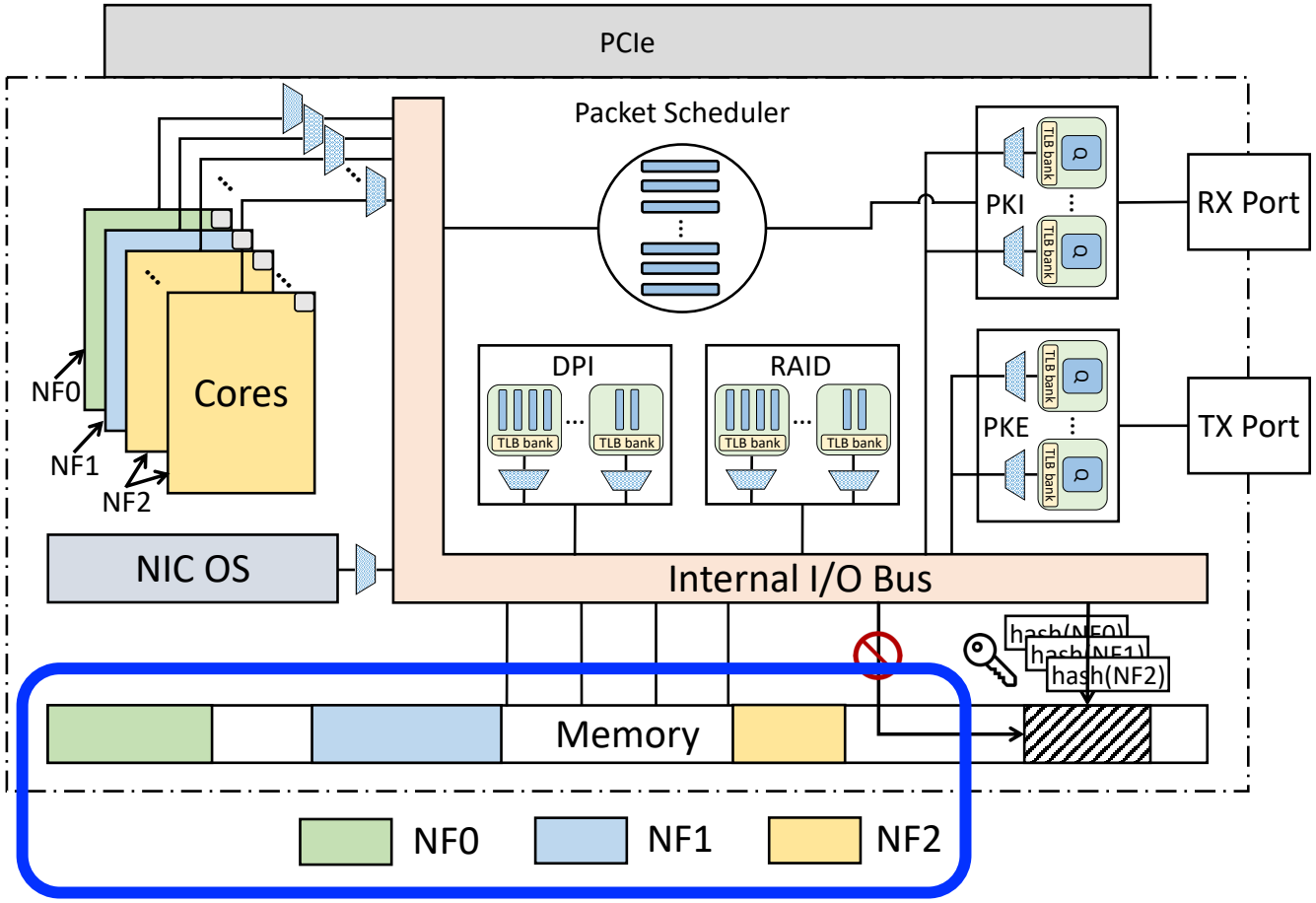
Single-owner RAM Semantics

Bus Arbitration

Accelerator Virtualization

Packet IO Virtualization

S-NIC: Key Designs



Single-owner RAM Semantics

Bus Arbitration

Accelerator Virtualization

Packet IO Virtualization

Design: Single-Owner RAM Semantics

Goals:

- Prevent NFs from accessing other NF's memory.
- Prevent OS from accessing arbitrary memory.
- Prevent side channels in cache.

Solution – single-owner RAM semantics

- A RAM region exclusively belongs to either a **running** network function, or to the management OS.

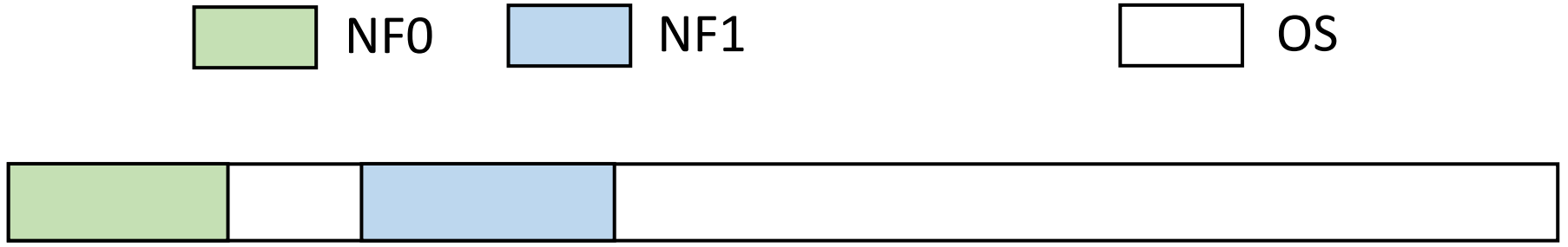
Preventing OS from modifying Translations

No (modifyable) page table for programmable cores.

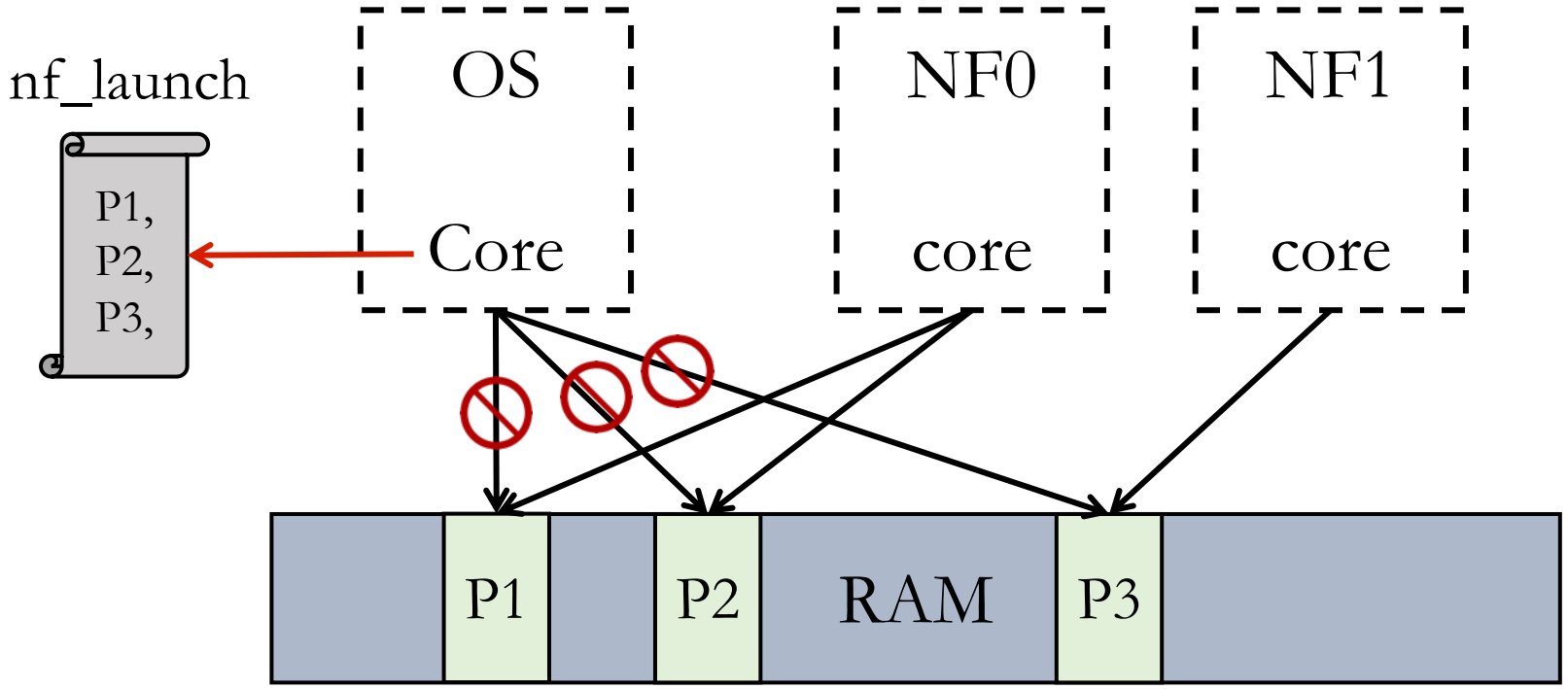
- Preventing NFs and OS from manipulating the mapping.
- As the cost of constraining the memory an NF can use.
- NF profiling shows: common NFs require < 70MB memory

nf_launch installs TLBs with memory mapping on each programmable core.

- Once nf_launch completes, the hardware sets the TLBs to read-only
- Common NFs require < 40 TLB entries (2MB page)



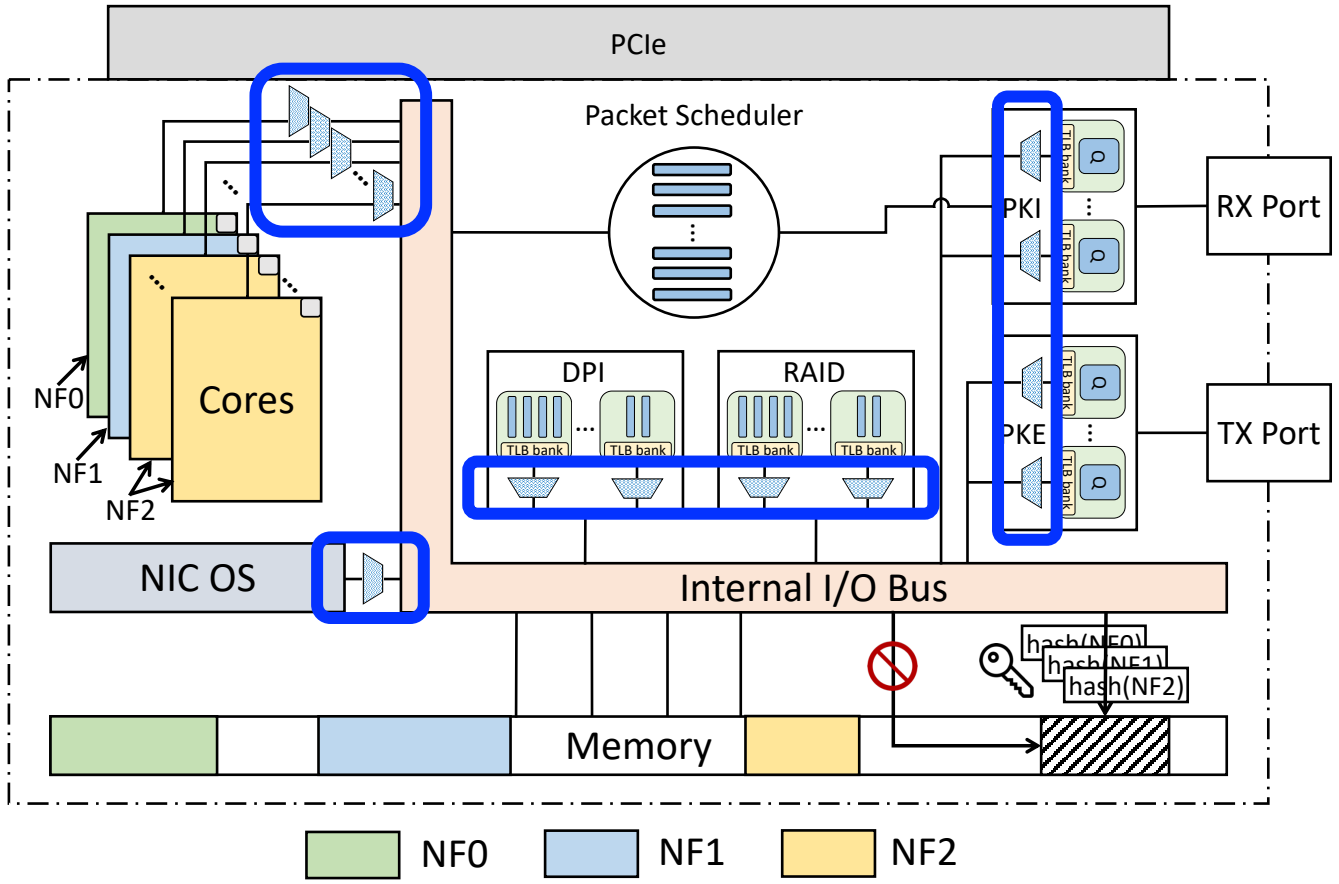
Preventing OS using Blacklist Page Table



Use trusted hardware instruction nf_launch:

Blacklist NF RAM pages and configurations on the OS core

S-NIC: Key Designs



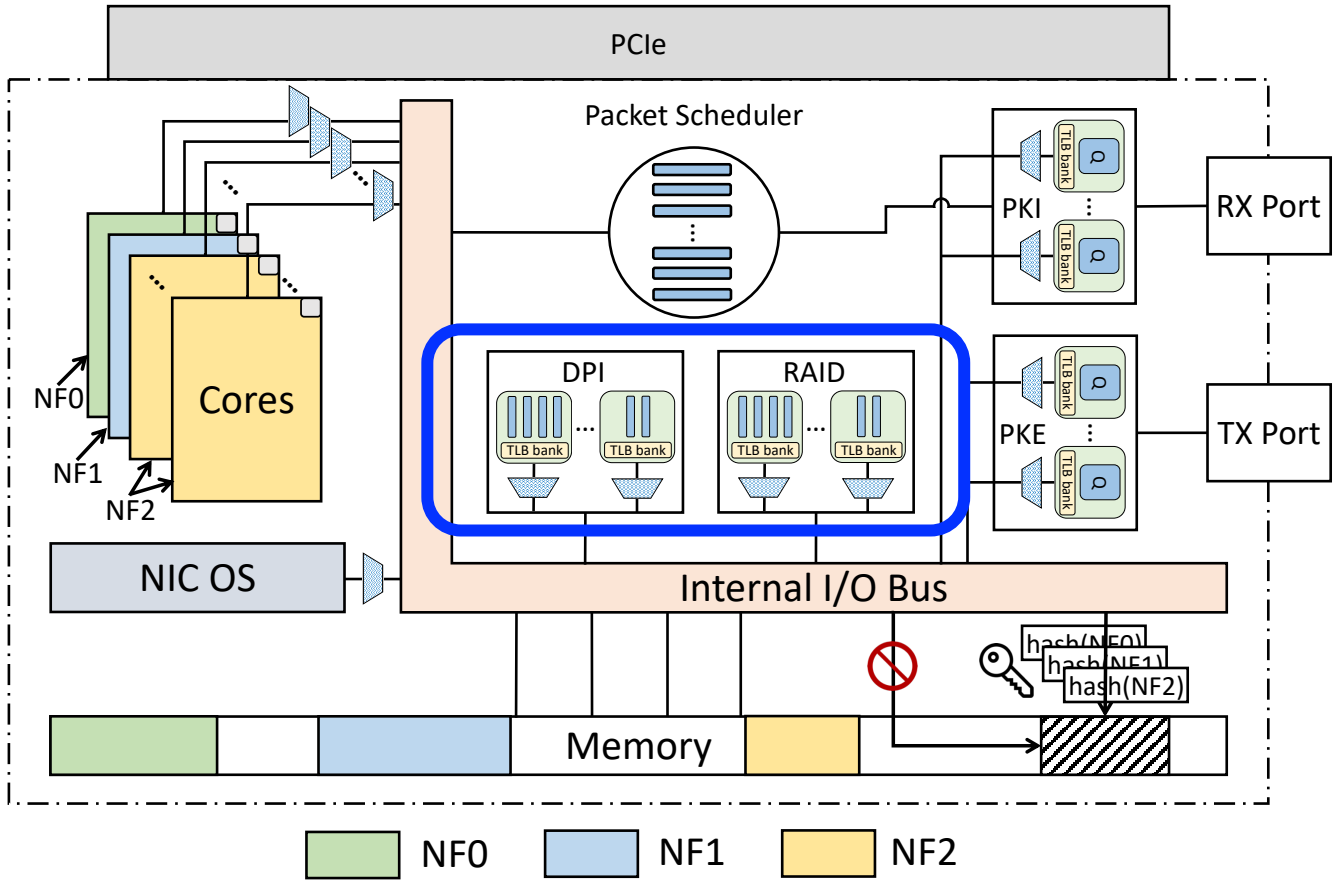
Single-owner RAM Semantics

Bus Arbitration

Accelerator Virtualization

Packet IO Virtualization

S-NIC: Key Designs



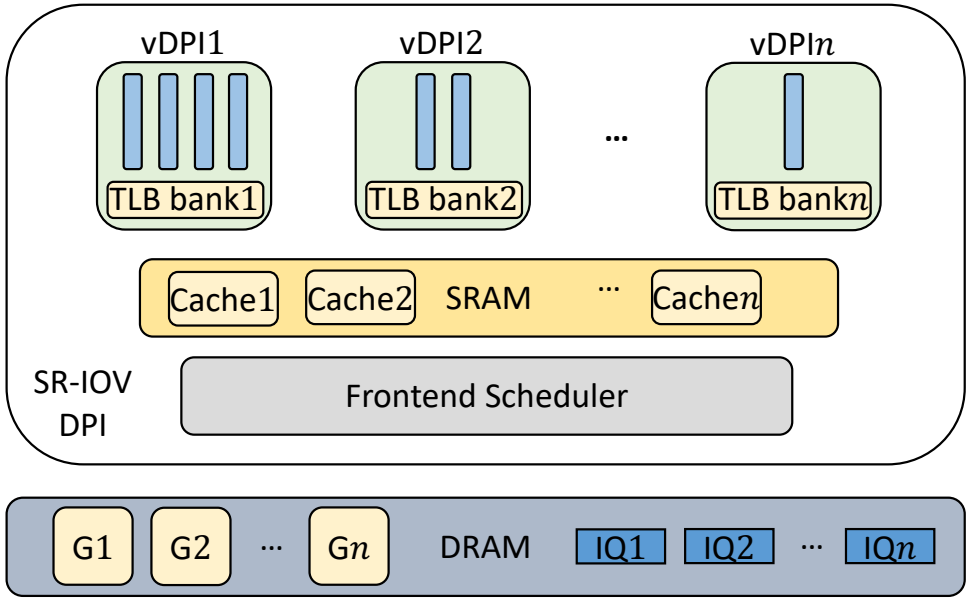
Single-owner RAM Semantics

Bus Arbitration

Accelerator Virtualization

Packet IO Virtualization

A. Clustering Threads with TLB Banks

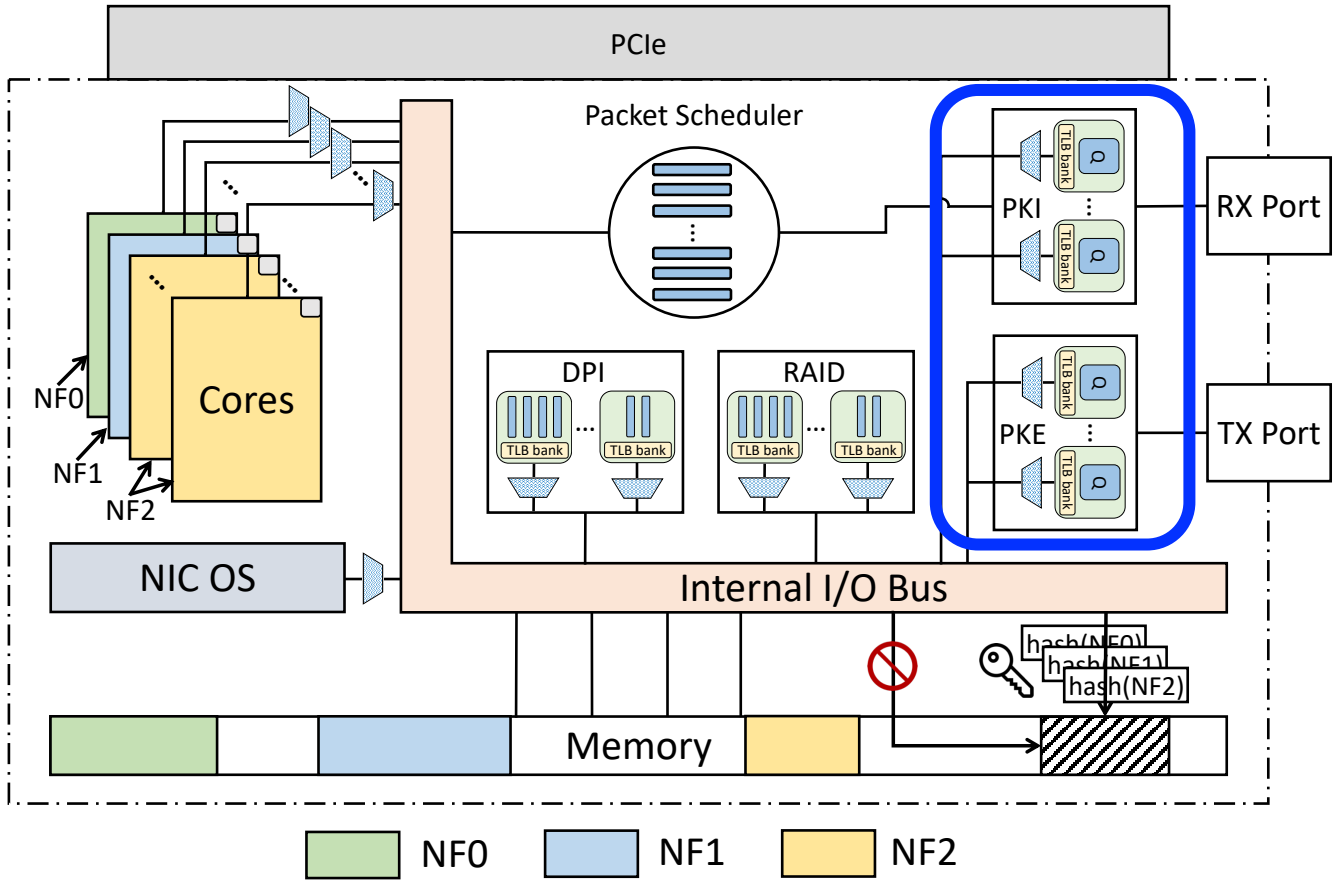


S-NIC partitions threads and RAM into multiple clusters (vDPIs)

- Avoid context switching to get high performance.

Each vDPI has a TLB banks with appropriate memory mappings.

S-NIC: Key Designs



Single-owner RAM Semantics

Bus Arbitration

Accelerator Virtualization

Packet IO Virtualization

Evaluation: Hardware Cost (TLB entries)

- Use standard open source NFs with real traffic setting
  - Take the largest memory, accelerator, virtual packet pipeline requirements
- Use the McPAT modeling framework to generate hardware cost estimates,
- Compare to an ARM Cortex-A9 multicore processor (a relatively small processor).

	TLBs for programmable cores	TLBs for virtualized accelerators	TLBs for packet IO
Chip area (mm^2)	0.163	0.091	0.074
Power (W)	0.088	0.044	0.034

8.89% chip area and **11.45%** power.

Evaluation: Performance

Gem5 models Marvell LiquidIO NIC cores, cache, memory, and bus with:

- Static partitioning for the cache.
- Temporal partitioning for the bus.

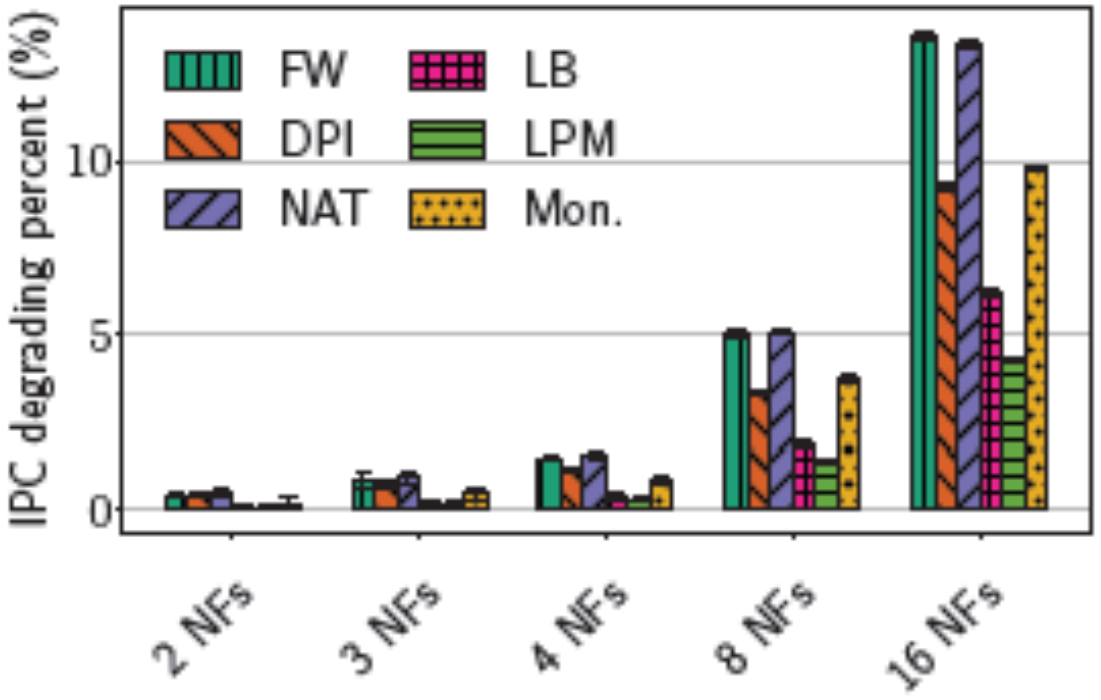
Runs six NFs

- Firewall (FW), DPI, NAT, Load Balancer (LB), LPM, Monitor (Mon)

We measured the instructions-per-cycle (IPC) degradation.

- Directly reflecting function throughput reduction.

Evaluation: Performance



Cache partitioning, and bus arbitration reduce function throughput by less than 13.71% in the worst case.

Conclusion

Outsource NFs to the cloud requires high performance and strong security, and offloading to smart NIC is an attractive option.

S-NIC redesigns smart NICs to provide strong isolation with modest hardware overheads, while preserving performance benefits of smart NICs.

S-NIC achieves both by

- enforcing single-owner semantics for on-NIC RAM and caches,
- pervasively virtualizing hardware accelerators and packet IO,
- providing dedicated bus bandwidth for each NF

Thanks!

Q&A