2022 OFA Virtual Workshop

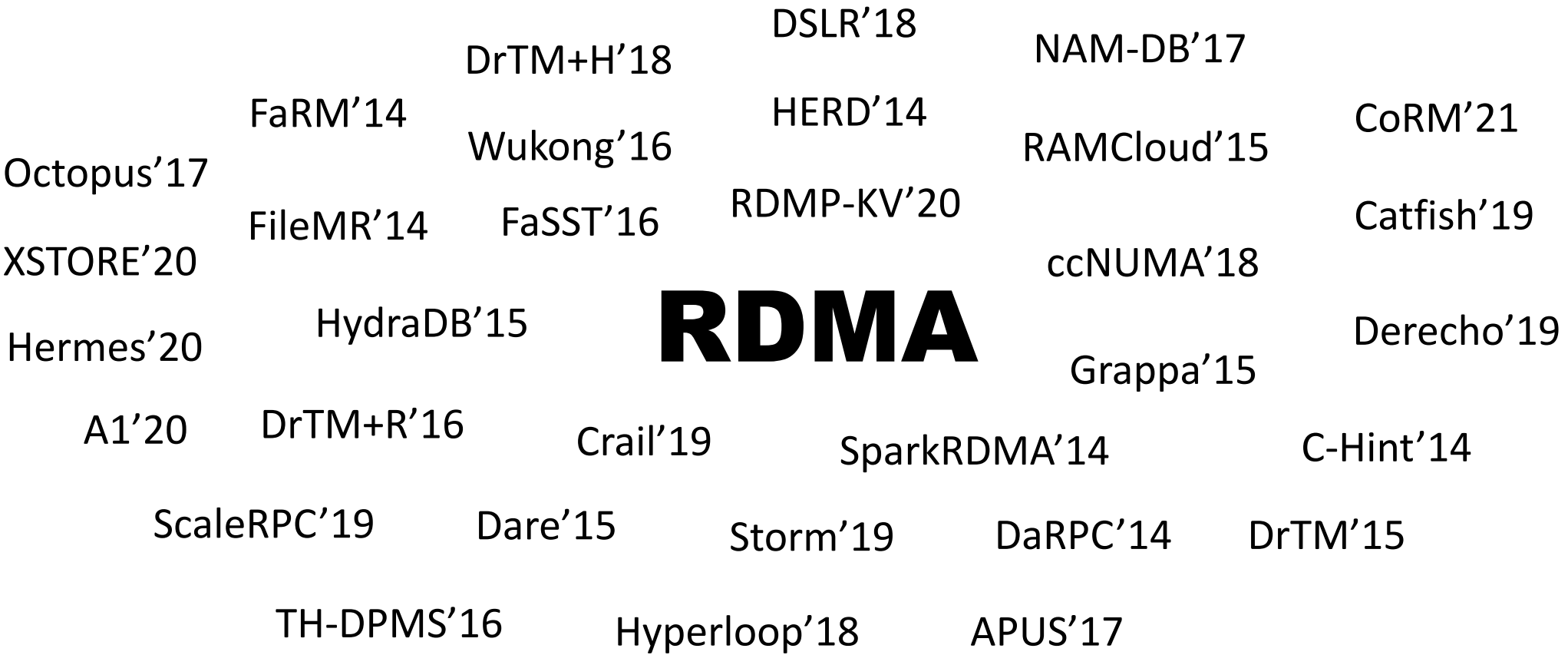
Exploiting RDMA Mistakes in NVMe-oF Storage Applications

Konstantin Taranov

ETH Zurich

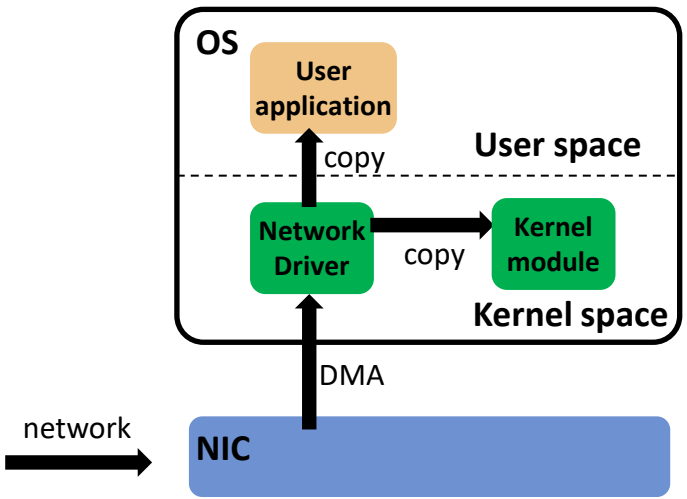
ETH zürich

RDMA is a new trend in system design



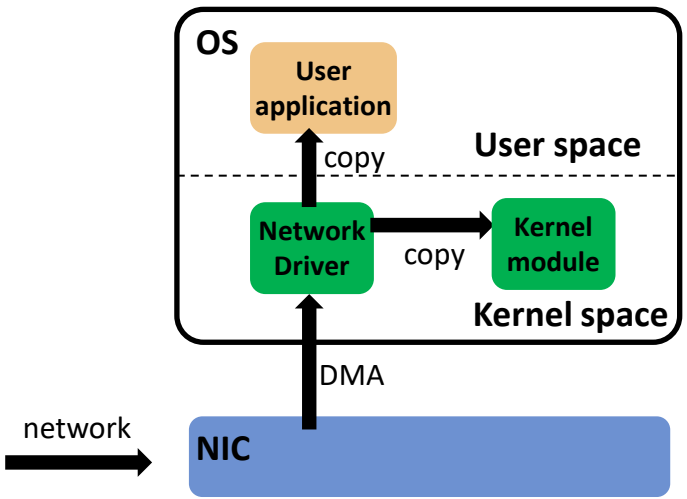
What is RDMA networking?

Socket-based networking

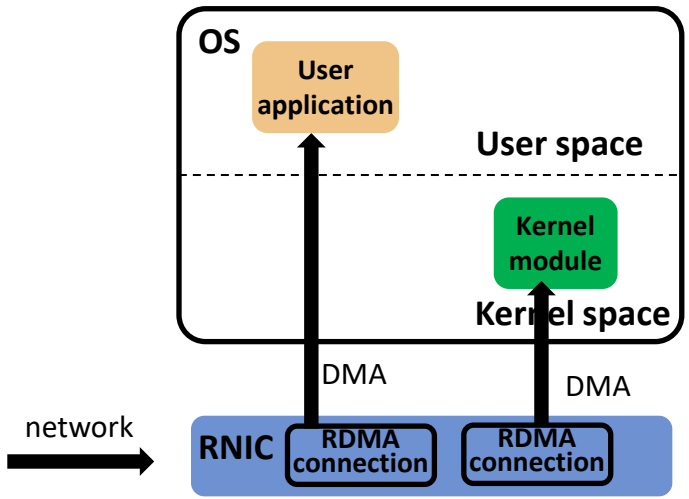


What is RDMA networking?

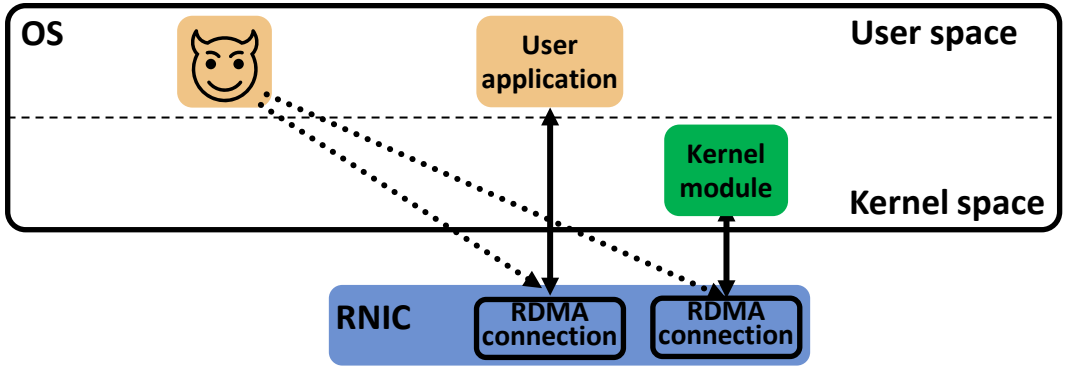
Socket-based networking



RDMA networking



Security of RDMA – NeVerMore's injection



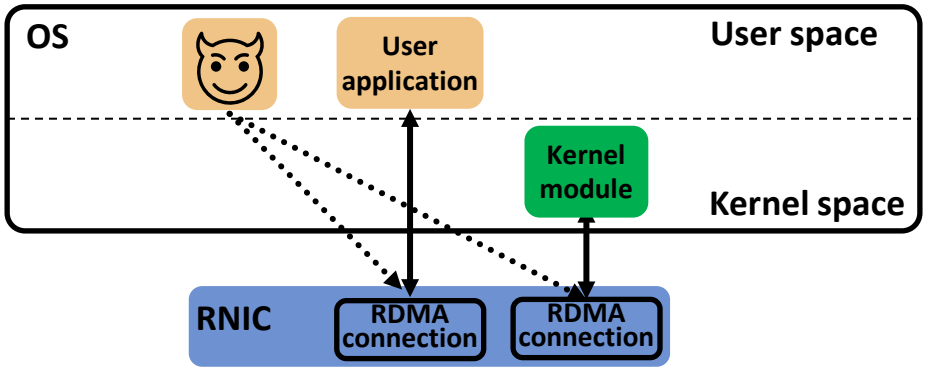
Injection without administrative privileges

Injection into any local RDMA connection

Injection into all IB-based protocols (including RoCE)

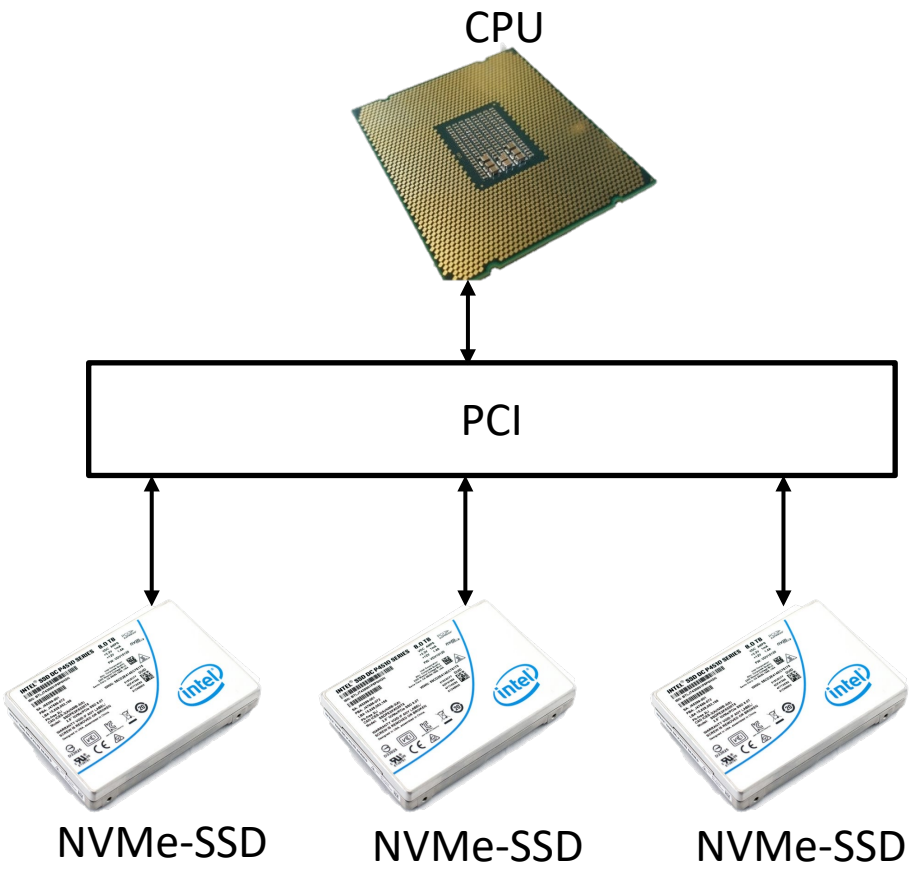
Taranov et al.: NeVerMore: Exploiting RDMA Mistakes in NVMe-oF Storage Applications. 2022. arXiv:2202.08080

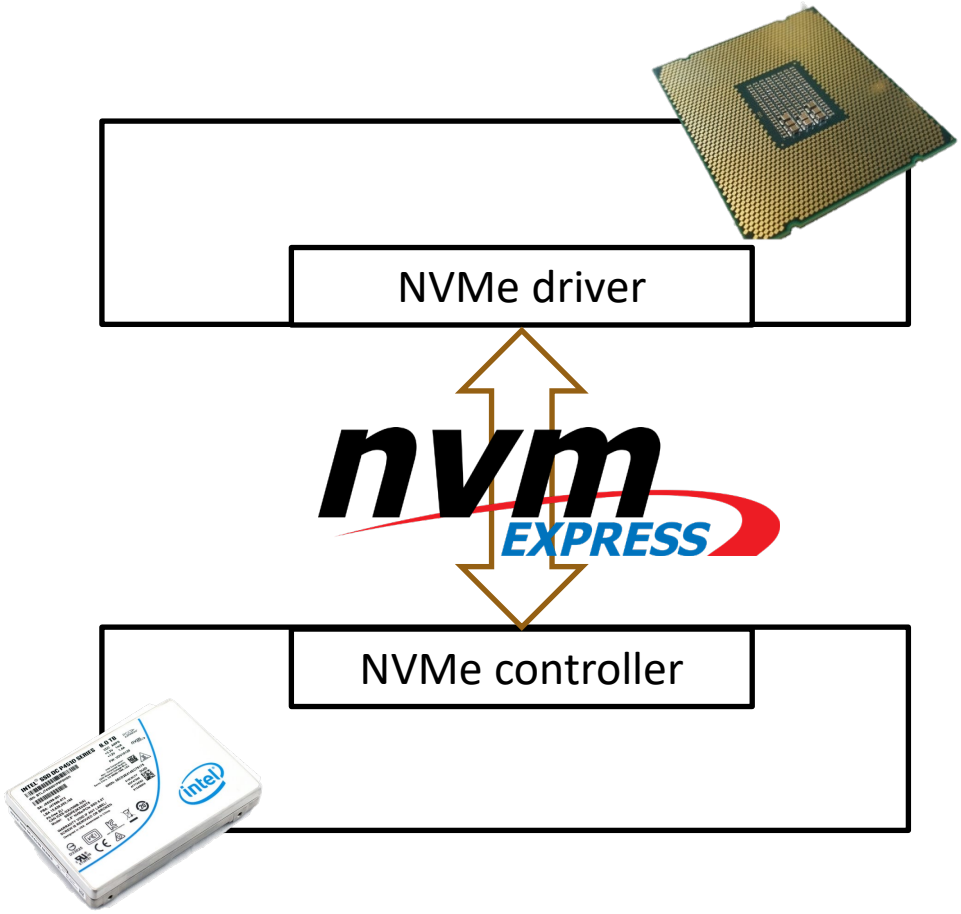
Implications of the attack



- A local user can manipulate any local RDMA connection
- A local user can manipulate RDMA-enabled kernel modules
- A local user can bypass security mechanisms of the OS and directly access the affected kernel module
- It is especially dangerous for the NVMe-oF protocol, relying on RDMA to access remote NVMe SSD

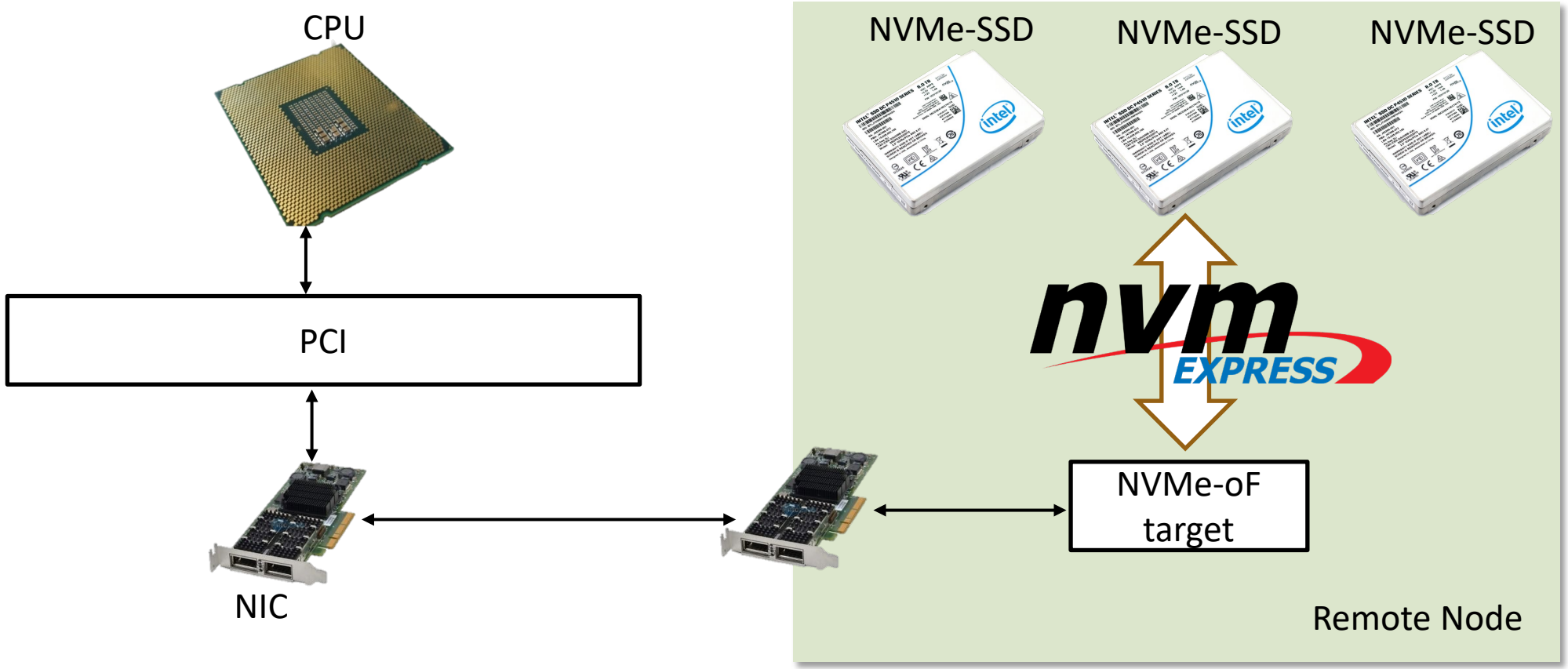
Non-Volatile Memory Express (NVMe)



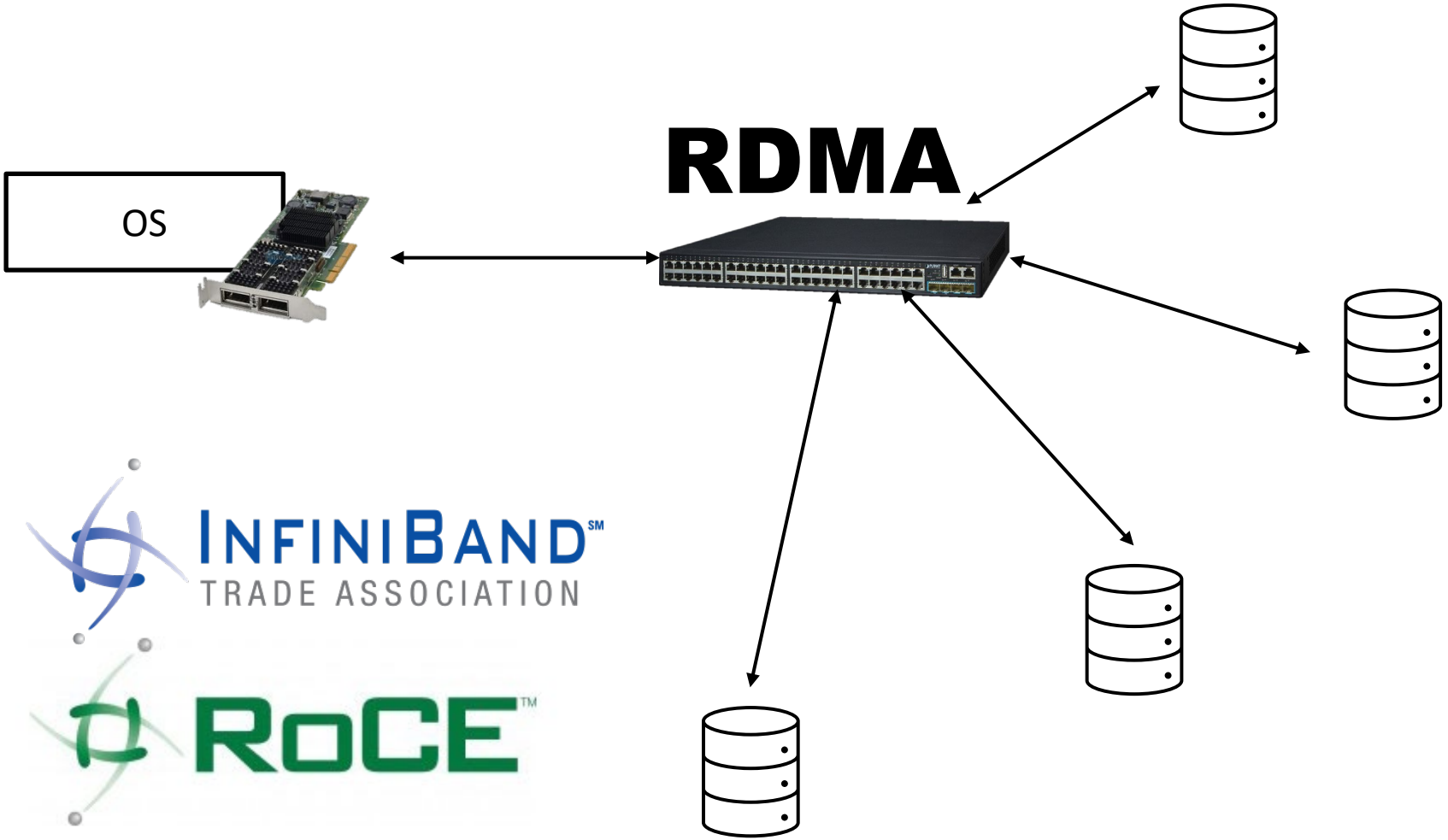


designed for performance – lower latency, higher bandwidth, lower CPU utilization etc.

NVMe over Fabrics (NVMe-oF)



Storage disaggregation over RDMA-capable networks

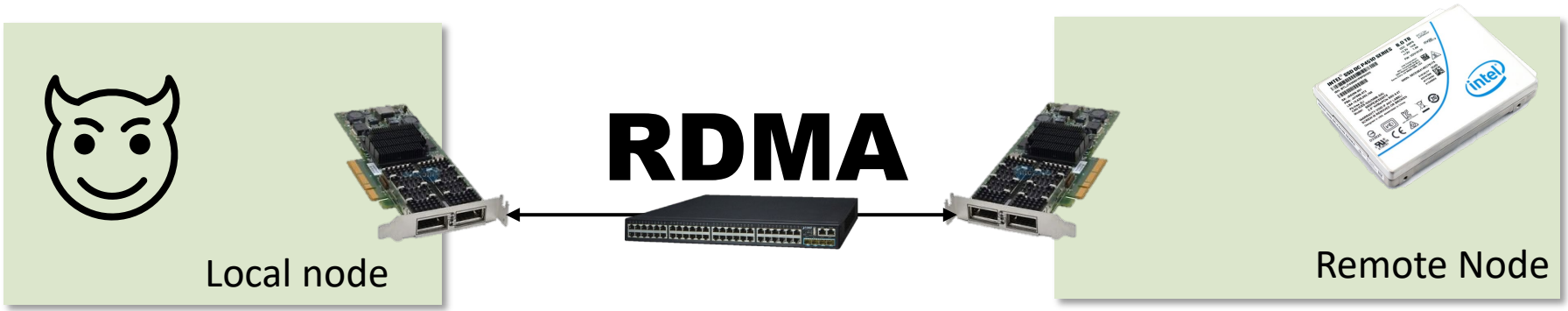


INFINIBAND℠
TRADE ASSOCIATION

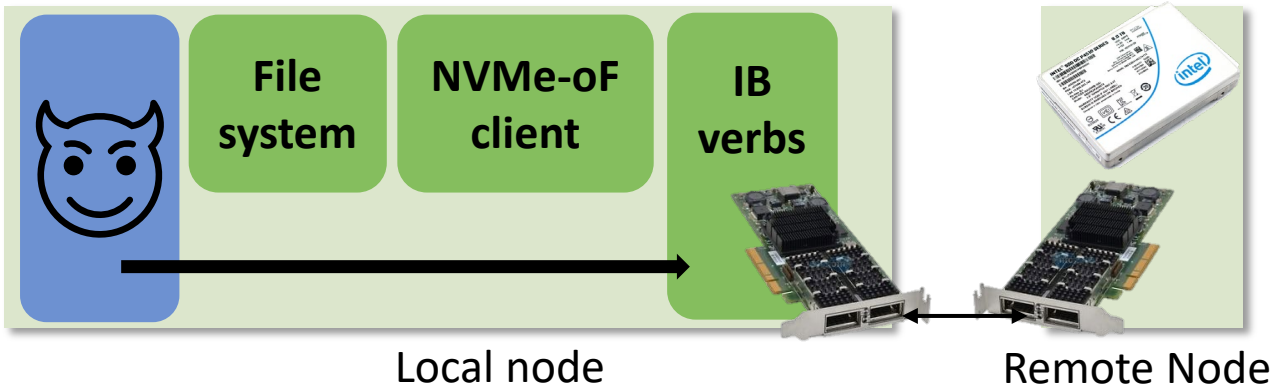
RoCE™

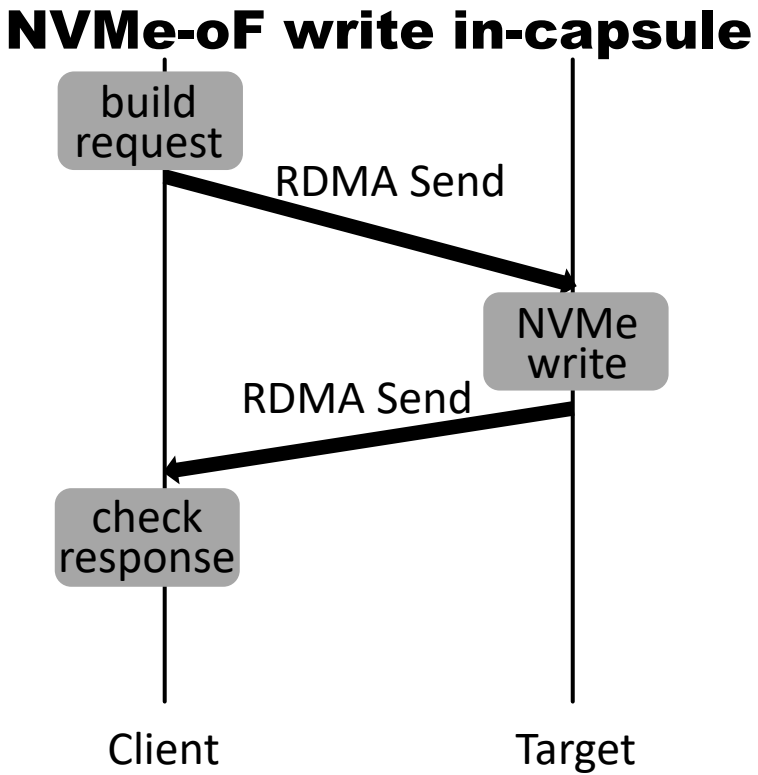
Threat models

- **Model TLU – The attacker is at a local node. It does not have root privileges.**



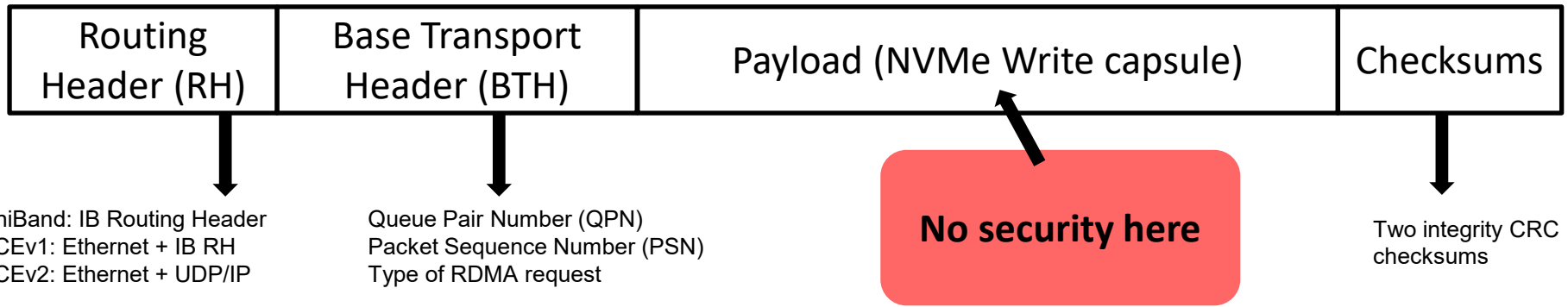
Towards injection of NVMe-oF write

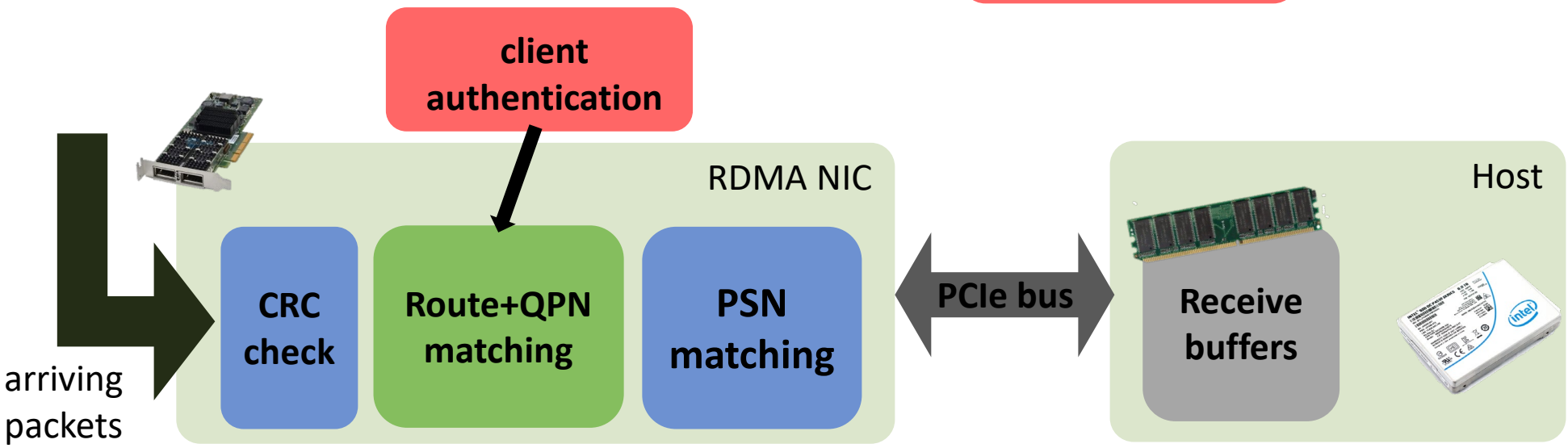




- **Setting TLU:**
  - The kernel mounts a remote NVMe SSD and installs a file system on it
  - The attacker is a user of the machine which uses the NVMe-oF client

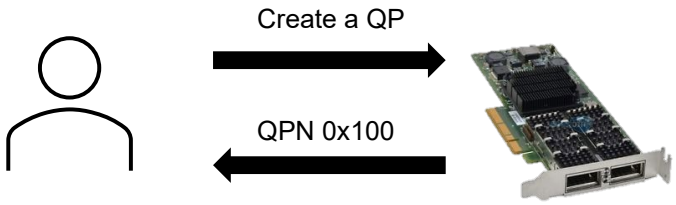
Background: RDMA send packet format and packet processing

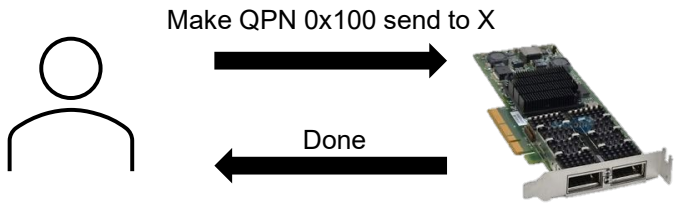




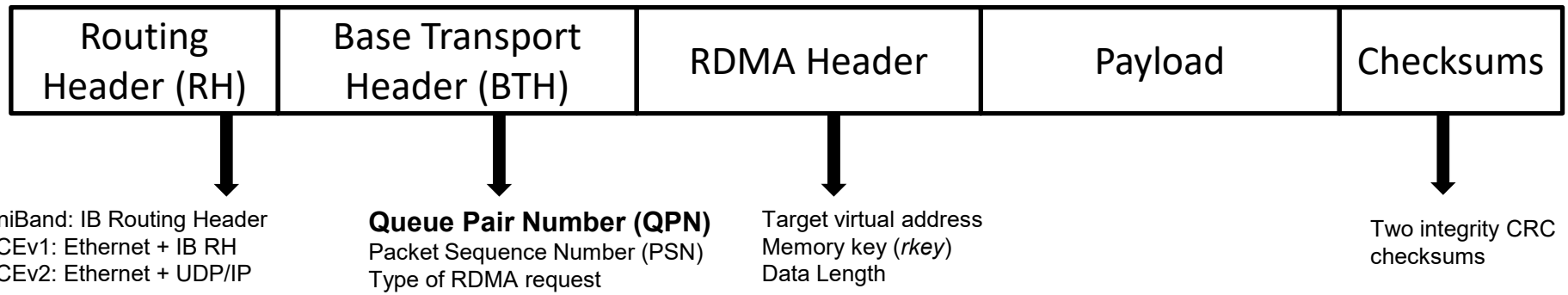
Fundamental vulnerabilities in InfiniBand-based protocols

- **1) The IBV user space library allows to create any RDMA connection with no sudo:**
  - A user can manually create a QP and add to it routing, PSNs, destination QPN

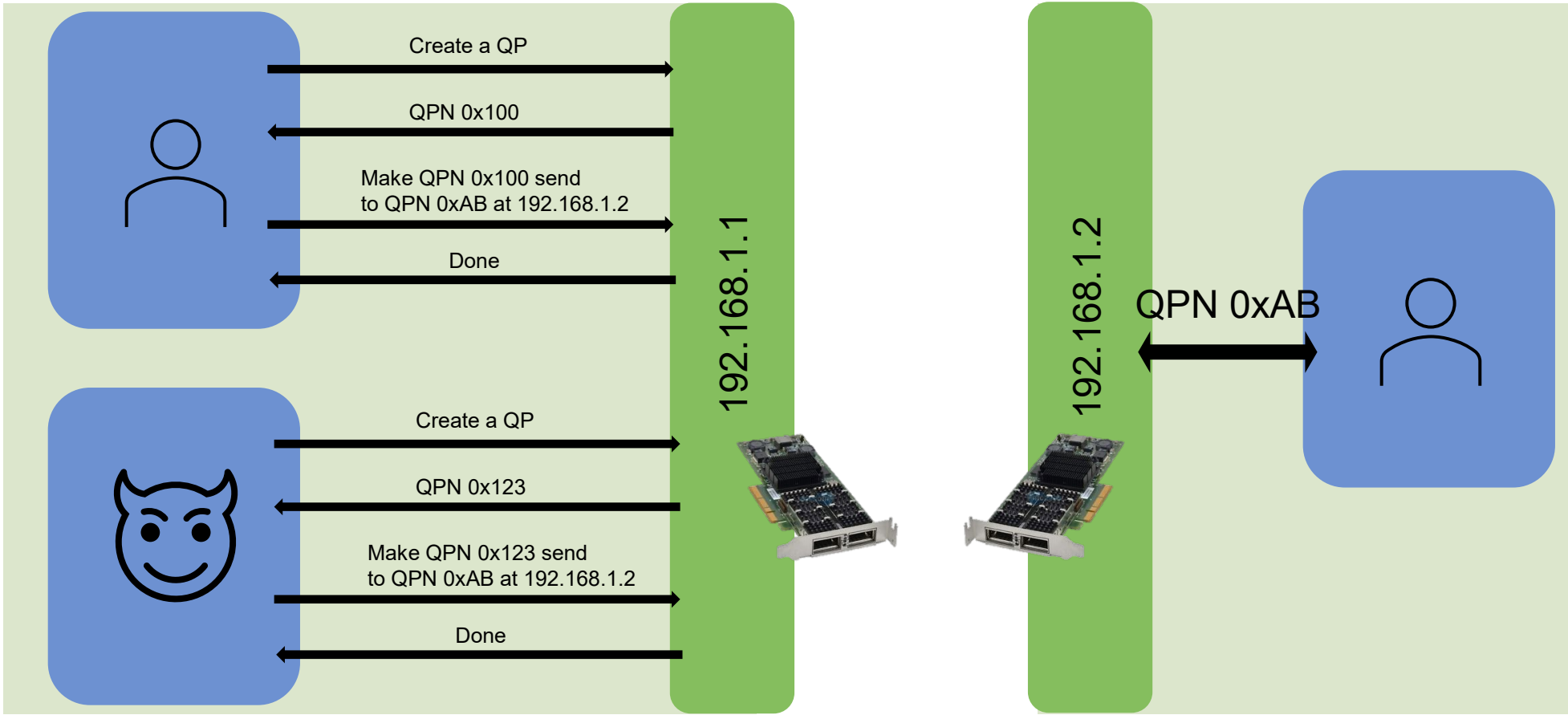




- **2) The Base Transport Header does not include source QPN**



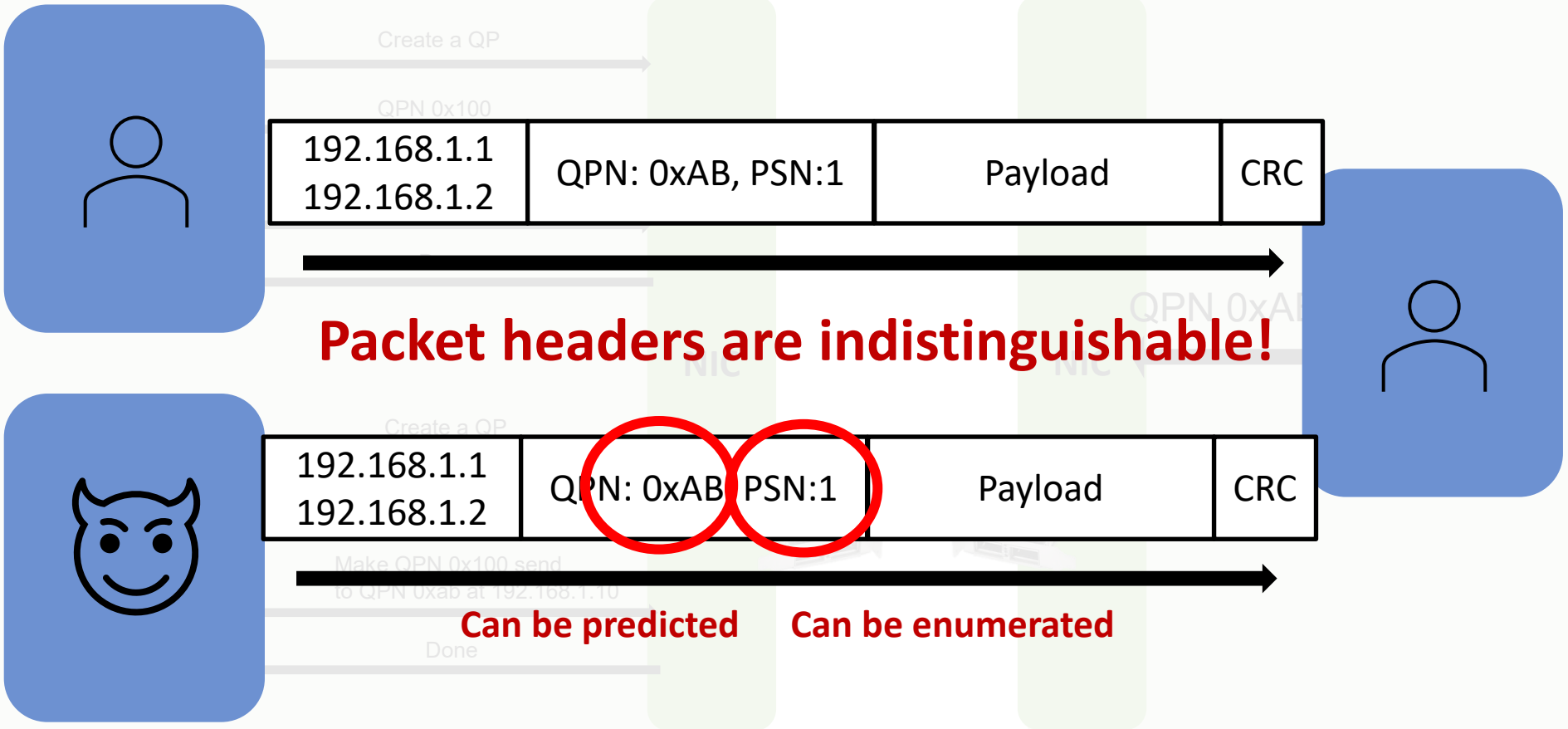
Packet forging with no root



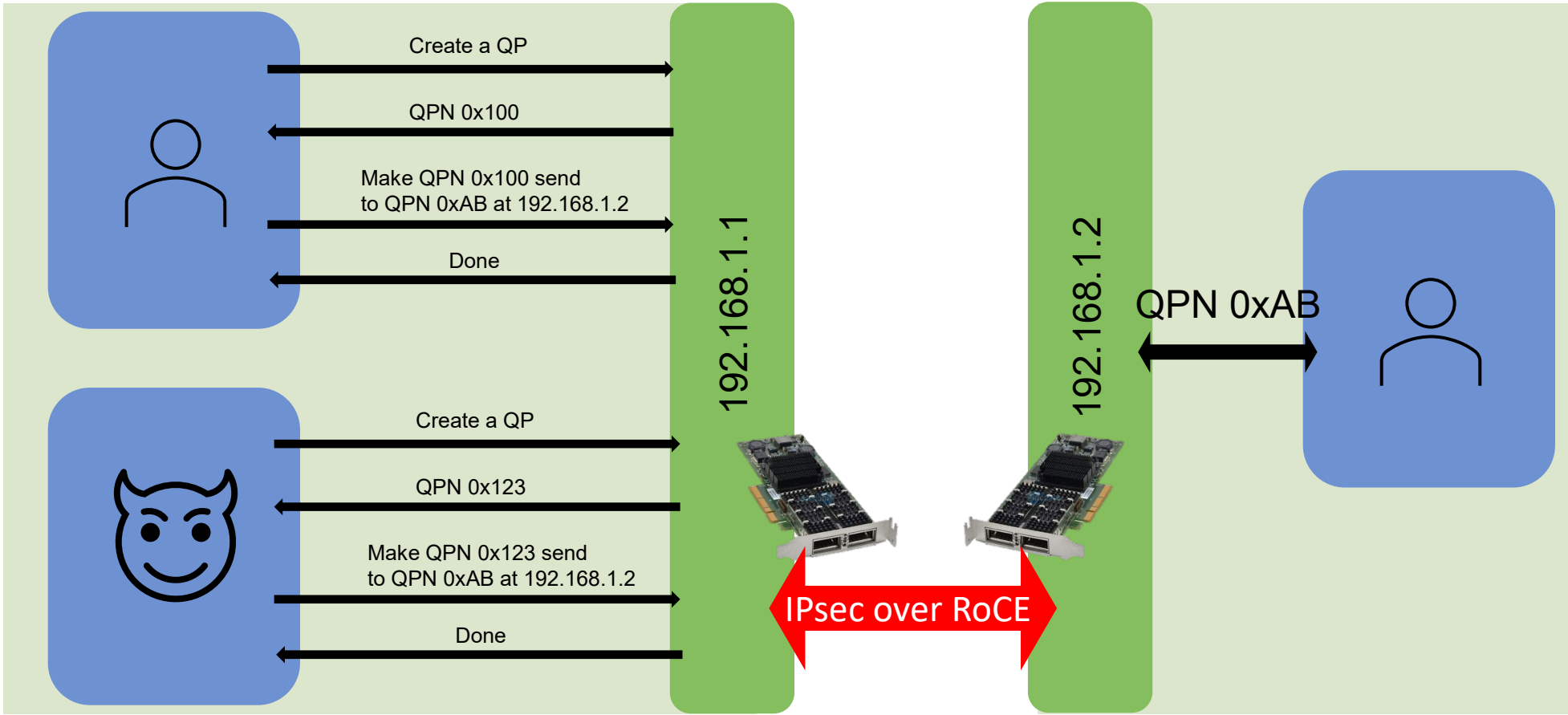
A local node

A remote node

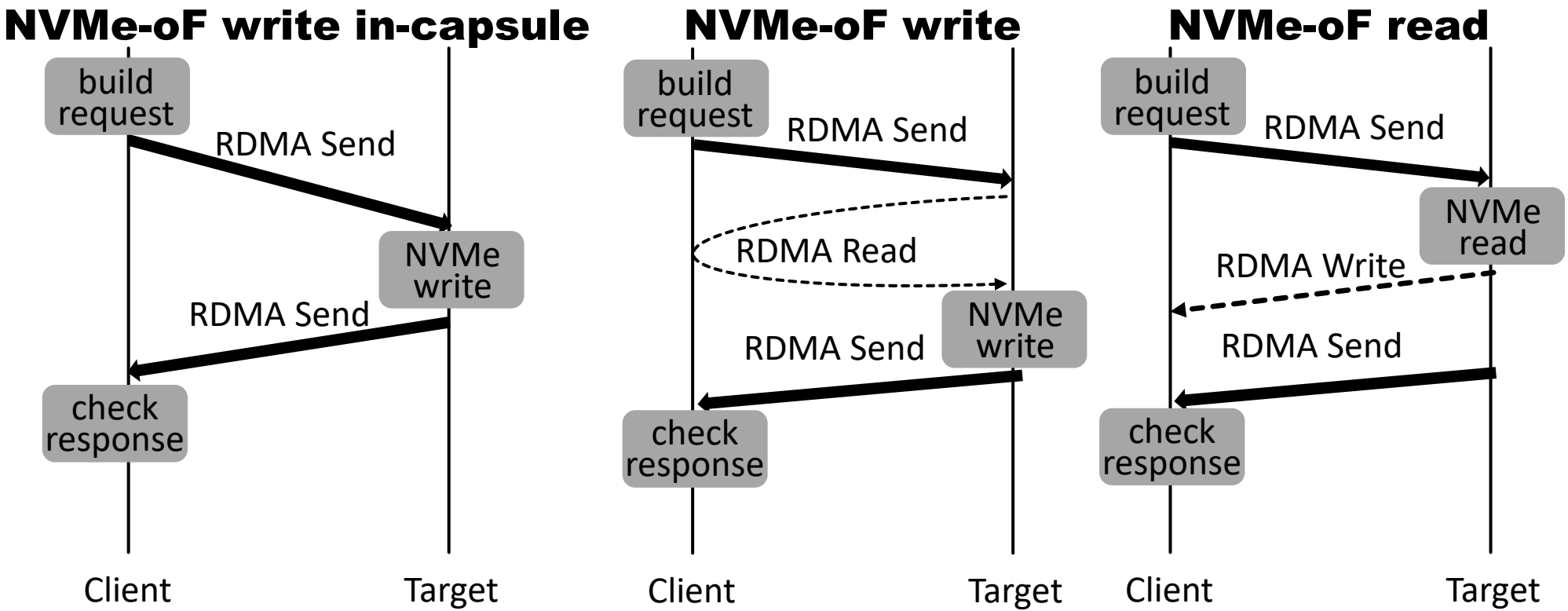
Packet forging with no root



Packet forging with no root – IPsec over RoCE is also vulnerable



NVMe-oF protocol



Takeaway for NVMe-oF

- ***Storage Performance Development Kit* (*SPDK*)**

Implemented in user space

- **Linux Kernel modules**
  - Client: nvme-rdma
  - Target: nvmet-rdma

Implemented in kernel space

- Existing security mechanisms in the NVMe-oF protocol:
  - In-band security – For client/target authentication at connection establishment
  - IPsec – To prevent injection into the secure link

	Threat Model TLU			Threat Model TRA			
Attack	None	In-band	IPsec	None	In-band	IPsec	Effect
Spoof NVMe-oF request	Yes	Yes	Yes	Yes	Yes	No	Execution of falsified request
Spoof NVMe-oF response	Yes	Yes	Yes	Yes	Yes	No	Early termination
Corrupt memory	Yes[1]	Yes[1]	Yes[1]	Yes[1]	Yes[1]	No	Use of falsified data

More in the paper

Packet forging with root – TRA model

- **We need to have correct routing header**
  - T1. The attacker had the same routing header as the victim, because they were at the same machine

192.168.1.1 192.168.1.2	QPN: 0xAB, PSN:1	Payload	CRC

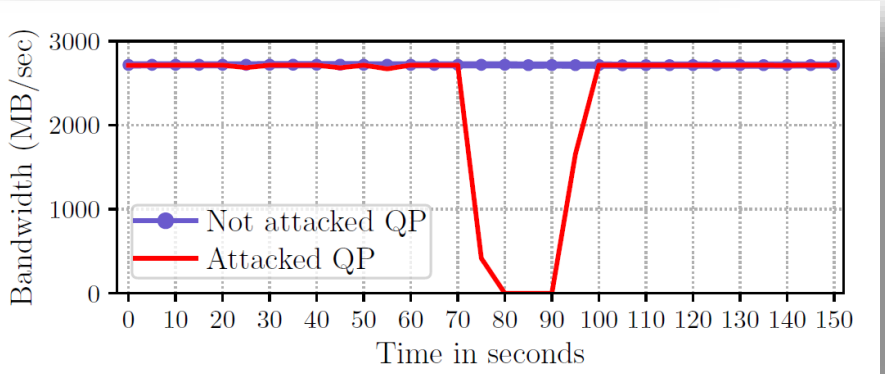
  - T2. We need to change the address of the NIC
    RoCE: sudo ifconfig IP DEVICE up
    IB: managed by a subnet manager that is running on a switch or another machine
  - T2. We need to change the address of the NIC for IB bypassing the subnet manager
    ibportstate is a tool to change a configuration of a device directly
    IB : sudo ibportstate -D 0 1 smlid 0x1 lid 0x4 active -d

The change is temporal but it lasts enough to perform an injection!

Attack	Threat Model TLU			Threat Model TRA			Effect
	None	In-band	IPsec	None	In-Band	IPsec	
Spoof NVMe-oF request	Yes	Yes	Yes	Yes	Yes	No	Execution of falsified request
Spoof NVMe-oF response	Yes	Yes	Yes	Yes	Yes	No	Early termination
Corrupt memory	Yes[1]	Yes[1]	Yes[1]	Yes[1]	Yes[1]	No	Use of falsified data
Exhaust QPNs	Yes[2]	Yes[2]	Yes[2]	No	No	No	Connection failure
Spoof CNPs	No[3]	No[3]	No[3]	Yes	Yes	No	Connection slowdown
Spoof RDMA-CM disconnect	Yes	Yes	Yes	Yes	Yes	Yes	Disconnection
Spoof invalid packet [26]	Yes	Yes	Yes	Yes	Yes	No	Disconnection

Feasibility of the attacks on NVMe-oF

[1] Linux kernel uses fast memory registrations with invalidation, which increases the complexity of the attack.
[2] Can be mitigated with RDMA Controller [19].
[3] Injection of CNPs is possible only for RoCE with administrative permissions.



RDMA-CM threats

- **We can forge RDMA-CM disconnect requests with no sudo**
- **We need to guess:**
  - QPN
  - RDMA-CM IDs of the initiator and the target
- **RDMA-CM IDs are predictable**

```
//on loading the RDMA-CM kernel module
uint32_t seed = get_random_bytes(sizeof(uint32_t));
uint32_t local_id = 0;
//to assign hidden identifier
uint32_t get_id(){
    return seed ^ (local_id++);
}
```

Figure 5: A pseudo-code showing how RDMA-CM kernel modules assigns identifiers to connections.

Mitigations for IB-based protocols – prevent injection under TLU

- **Change packet format**
  - Add source QPN
- **Inform users to employ application-layer authentication**
  - As we propose for NVMe-oF in this work [1]
- **Implement secure transport**
  - As we propose in sRDMA [2]
- **Deploy infrastructure to detect the injection under TLU**
  - As proposed in Bedrock [3]

[1] Taranov et al.: NeVerMore: Exploiting RDMA Mistakes in NVMe-oF Storage Applications. 2022. arXiv:2202.08080
[2] Taranov et al.: sRDMA - Efficient NIC-based Authentication and Encryption for Remote Direct Memory Access. Usenix ATC 2020.
[3] Jiarong Xing et al., Bedrock: Programmable Network Support for Secure RDMA Systems. Usenix Security 2022

Summary

- Vulnerabilities in the packet format allows spoofing RDMA packets with no root
- The injection allows to manipulate local RDMA-enabled kernel modules from the user space
- NVMe-oF security is not sufficient for insecure RDMA interconnects
- RDMA requires a secure transport

Contact information:
Konstantin Taranov
konstantin.taranov@inf.ethz.ch