# NetReach: Guaranteed Network Availability and Reachability to enable Resilient Networks for Embedded Systems

Tom Van Eyck, Sam Michiels, Xiaojiang Du, Danny Hughes

SysTEX 2024

KU LEUVEN DistriNet

# Introduction

- Industrial networks
  - Robots
  - Automated Guided Vehicles
  - …
- Powerful processors
- Worldwide deployment
  - No in person interventions

"AGV - Automated Guided Vehicle" by Marta Veverica is licensed under CC BY-SA 4.0.

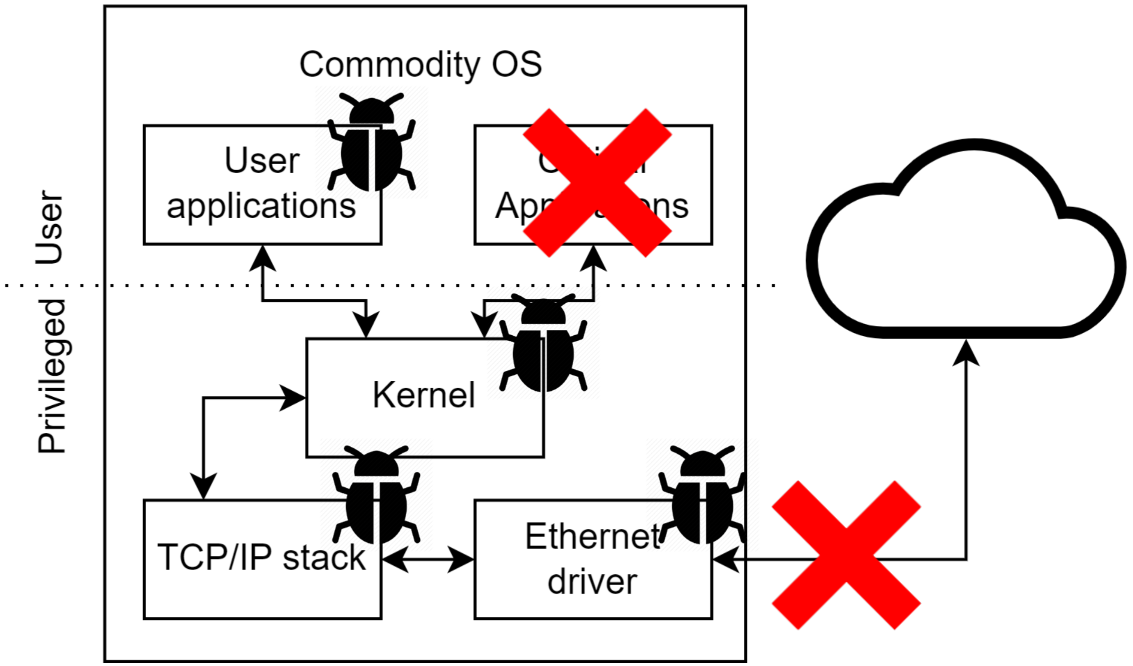
# Introduction

- Commodity OS
  - Networking
  - Remote monitoring
  - Updates
- Real-time control
  - Safety critical!
- Large attack surface

=> Problems

# Attacks on Industry

- Strong remote attacker
- Denial of Service
  - Device operation
- Large codebases
  - Ethernet driver alone: ~6k LoC
- Often untested
- Very little separation



=> High chance of failure or successfull attack

# State of the art: Availabilty with TEE

Session 5A: Control System Security CCS '21, November 15–19, 2021, Virtual Event, Republic of Korea

**Aion: Enabling Open Systems through Strong Availability Guarantees for Enclaves**

Fritz Alder
Jo Van Bulck
Frank Piessens

2022 IEEE Symposium on Security and Privacy (SP)

RT-TEE: Real-time System Availability for Cyber-physical Systems using ARM TrustZone

Jinwen Wang, Ao Li, Haoran Li, Chenyang Lu, Ning Zhang

**Mr-TEE: Practical Trusted Execution of Mixed-Criticality Code**

Tom Van Eyck
tom.vaneyck@kuleuven.be
DistriNet, KU Leuven
Leuven, Belgium

Hamdi Trimech
hamdi.trimech@kuleuven.be
DistriNet, KU Leuven
Leuven, Belgium

Sam Michiels
sam.michiels@kuleuven.be
DistriNet, KU Leuven
Leuven, Belgium

Danny Hughes
danny.hughes@kuleuven.be
DistriNet, KU Leuven
Leuven, Belgium

Majid Salehi
majid.salehi@nokia-bell-labs.com
Nokia Bell Labs
Antwerp, Belgium

Hassan Janjuaa
hassaan.janjua@gmail.com
DistriNet, KU Leuven
Leuven, Belgium

Thanh-Liem Ta
ta.thanh-liem.sup@hotmail.com
Polytech Sorbonne
Paris, France

**ABSTRACT**

Industry 4.0 is increasingly using commodity hardware and software in place of dedicated control systems to lower costs and increase flexibility. However, this means that critical control code must compete for resources with an increasingly complex software stack that exposes a new attack surface. The Mixed Criticality Trusted Execution Environment (Mr-TEE) tackles this problem at its root, by delivering availability for safety-critical control code, while running untrusted applications in a minimally modified Linux stack. This is achieved by providing a real-time scheduler and novel peripheral sharing system in the Secure World. Mr-TEE hence provides the *best of both worlds* for Industry 4.0 developers, ensuring the trusted execution of time-sensitive control applications, while minimizing design effort and restrictions for untrusted applications. Evaluation on an Arm TrustZone-enabled Cortex MCU demonstrates a worst-case overhead of 1.88% to support real-time scheduling in the TEE.

**CCS CONCEPTS**

• **Computer systems organization** → *Embedded systems*; **Real-time operating systems**; **Availability**; • **Security and privacy** → **Operating systems security**.

**KEYWORDS**

Cyber-Physical Systems, Mixed-Criticality, Trusted Execution Environments, Arm TrustZone

**ACM Reference Format:**
Tom Van Eyck, Hamdi Trimech, Sam Michiels, Danny Hughes, Majid Salehi, Hassan Janjuaa, and Thanh-Liem Ta. 2023. Mr-TEE: Practical Trusted Execution of Mixed-Criticality Code. In *Middleware Industrial Track '23: Proceedings of the 24th International Middleware Conference Industrial Track (Middleware '23), December 11–15, 2023, Bologna, Italy.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3626562.3626831

*Middleware '23, December 11–15, 2023, Bologna, Italy*
© 2023 Association for Computing Machinery.
ACM ISBN 979-8-4007-0427-7/23/12...$15.00
https://doi.org/10.1145/3626562.3626831

**1 INTRODUCTION**

Until recently, industrial control was implemented using reliable yet inflexible Programmable Logic Controllers (PLCs), which were rarely connected to the Internet. However, this approach is being challenged by Industry 4.0, which connects industrial assets such as machines, robots, and production lines with real-time analytics and control to improve efficiency, safety and quality. Additionally, these assets are inherently a Cyber Physical System (CPS), which interacts continuously with the physical world and are hence bound by stringent timing and safety requirements. This makes it crucial that all required system resources are made available to critical tasks in a real-time manner and protected from any undue interference. Extensive conversations with our industry partners in the context of industrial/applied research projects (TACOS[1], TRUSTI[2] and ERATOSTHENES[3]) identified support for deterministic *1 ms* control loops as a key requirement.

Connecting CPS to the Internet is best achieved by reusing reputable commodity OS components and applications because of their ease of implementation and well-tested nature. Unfortunately, these significantly increase the attack surface due to their large code base, exposing vulnerabilities [5, 15, 19] which range from nuisances [20] to life-threatening malfunctions [18]. This poses the urgent question: *How can strong trust in time-sensitive cyber-physical control functionality be preserved when using a commodity Linux OS?*

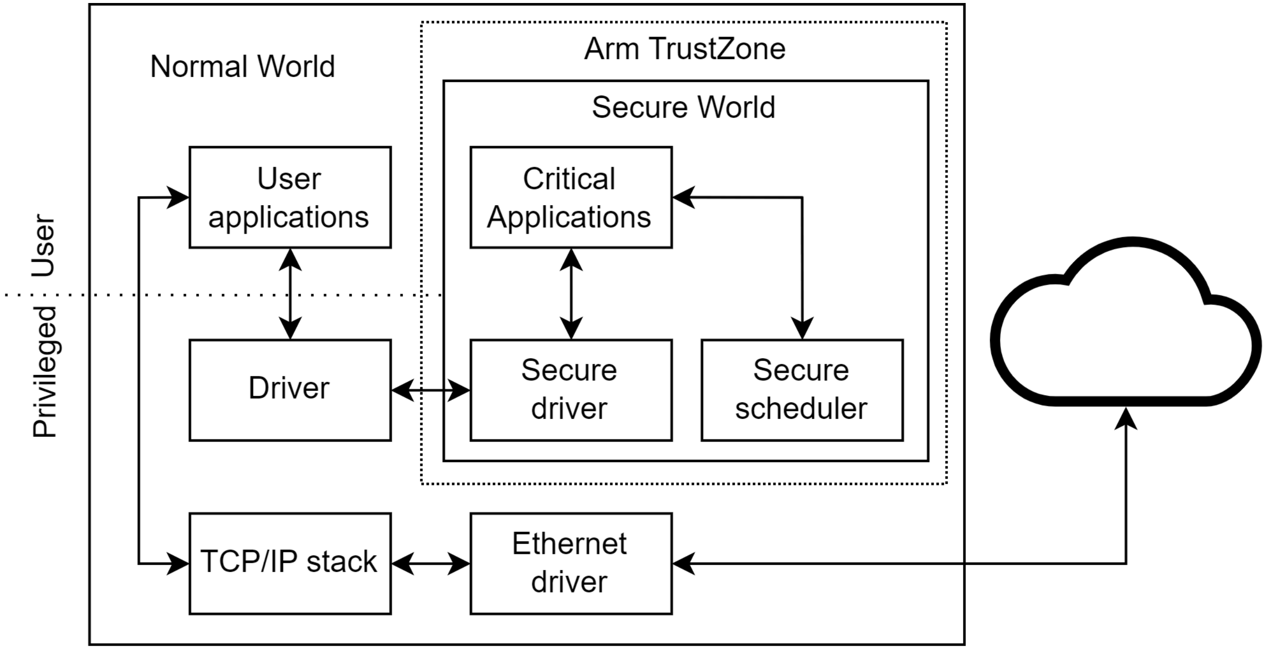
As prior research has shown, attackers can cause a catastrophic accident by delaying or altering sensor readings [27]. The end goal is thus to prevent attackers from tampering with the schedule of real-time tasks or overloading resources. To achieve this, it is necessary to prevent unauthorized reconfiguration of the system, and to deny unauthorized tasks from accessing certain critical peripherals. This must be achieved while preserving access to standard peripherals from the non-secure OS in order to prevent extensive re-engineering efforts.

This paper proposes Mr-TEE, which guarantees using a hardware-enabled *Trusted Execution Environment (TEE)* (specifically Arm TrustZone) to host and schedule safety-critical code, while maintaining a strong separation from the Linux OS, which executes outside of the TEE. Mr-TEE also provides mediated access to peripherals via a *Shared Secure Peripheral* Framework. This prevents untrusted applications from performing Denial of Service (DoS) attacks. In contrast to prior research in this area [29], Mr-TEE does not require

[1] https://distrinet.cs.kuleuven.be/research/projects/TACOS
[2] https://distrinet.cs.kuleuven.be/research/projects/TRUSTI
[3] https://eratosthenes-project.eu/

22

- Availability of Critical Code
- Mr-TEE [1] on Arm TrustZone:



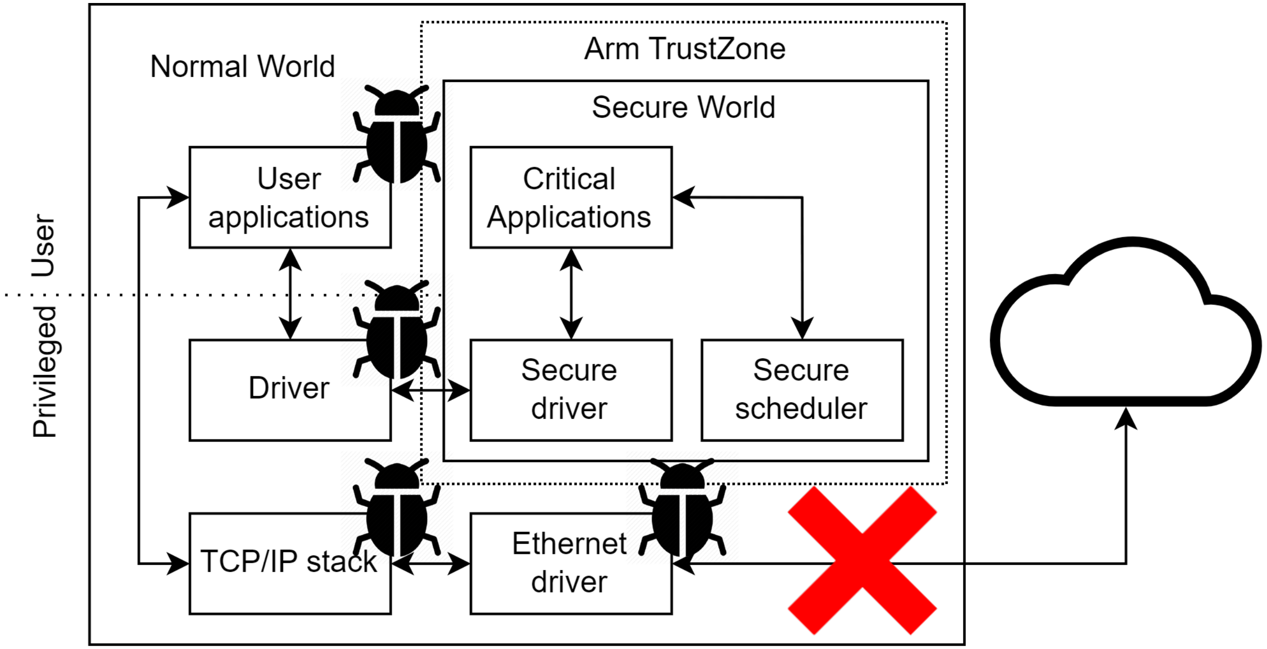
[1] T. Van Eyck et al., Mr-TEE: Practical Trusted Execution of Mixed-Criticality Code. 2023. doi: 10.1145/3626562.3626831.

# Mr-TEE: Practical Mixed-Criticality

- Real-time scheduler in TEE
  - Minimal implementation
- Secure sharing of peripherals
  - Interrupt passing
- Reboot of Linux

- No network availability

=> Costly manual intervention



# NetReach

Guaranteed Network Availability and Reachability

KU LEUVEN DistriNet

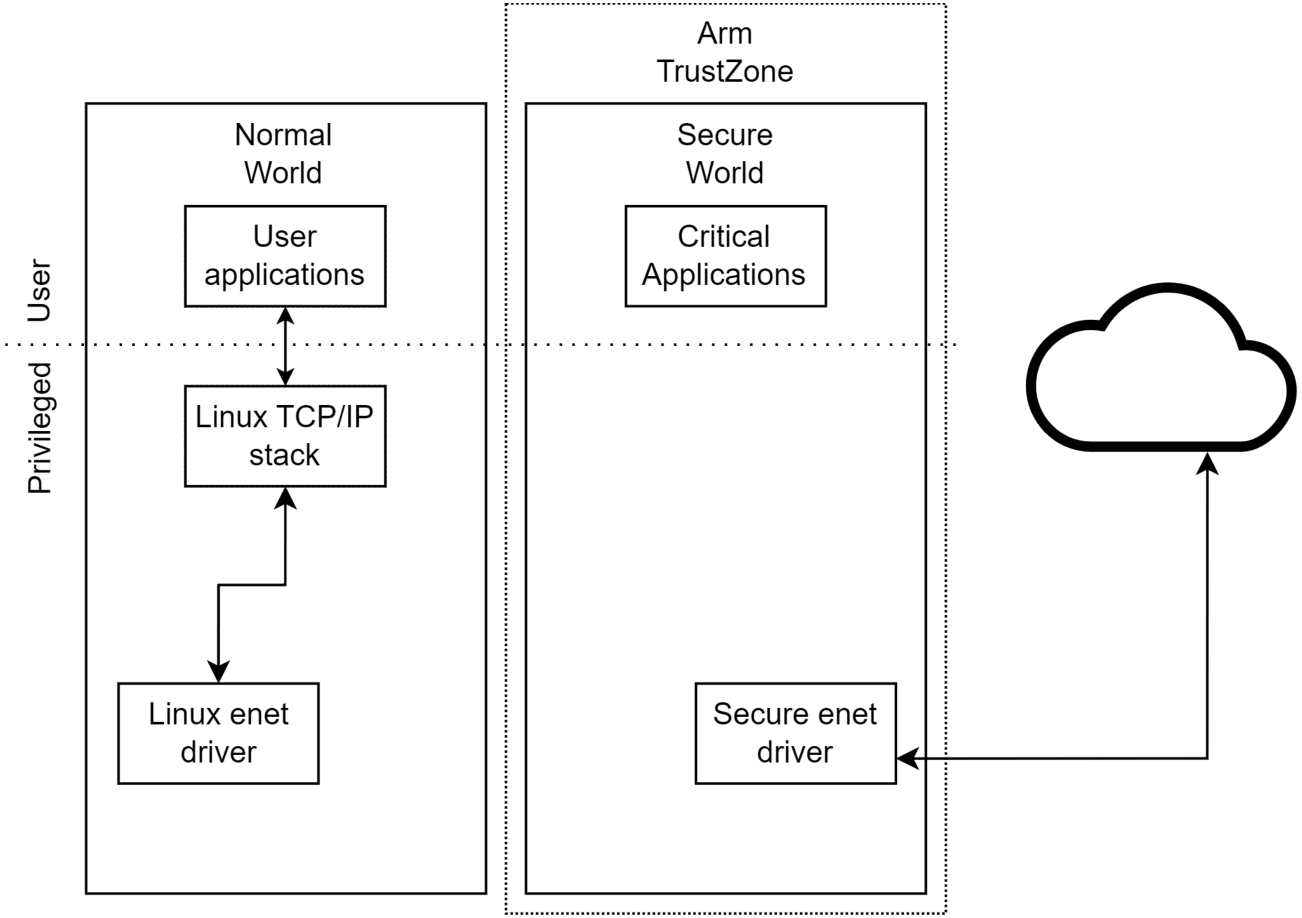
# NetReach

- First step towards resilient networks
  1. Always available network peripheral
  2. Always reachable backup network

- Requirements
  1. Protect peripheral from DoS by NW
  2. Provide backup network
  3. Minimize TCB

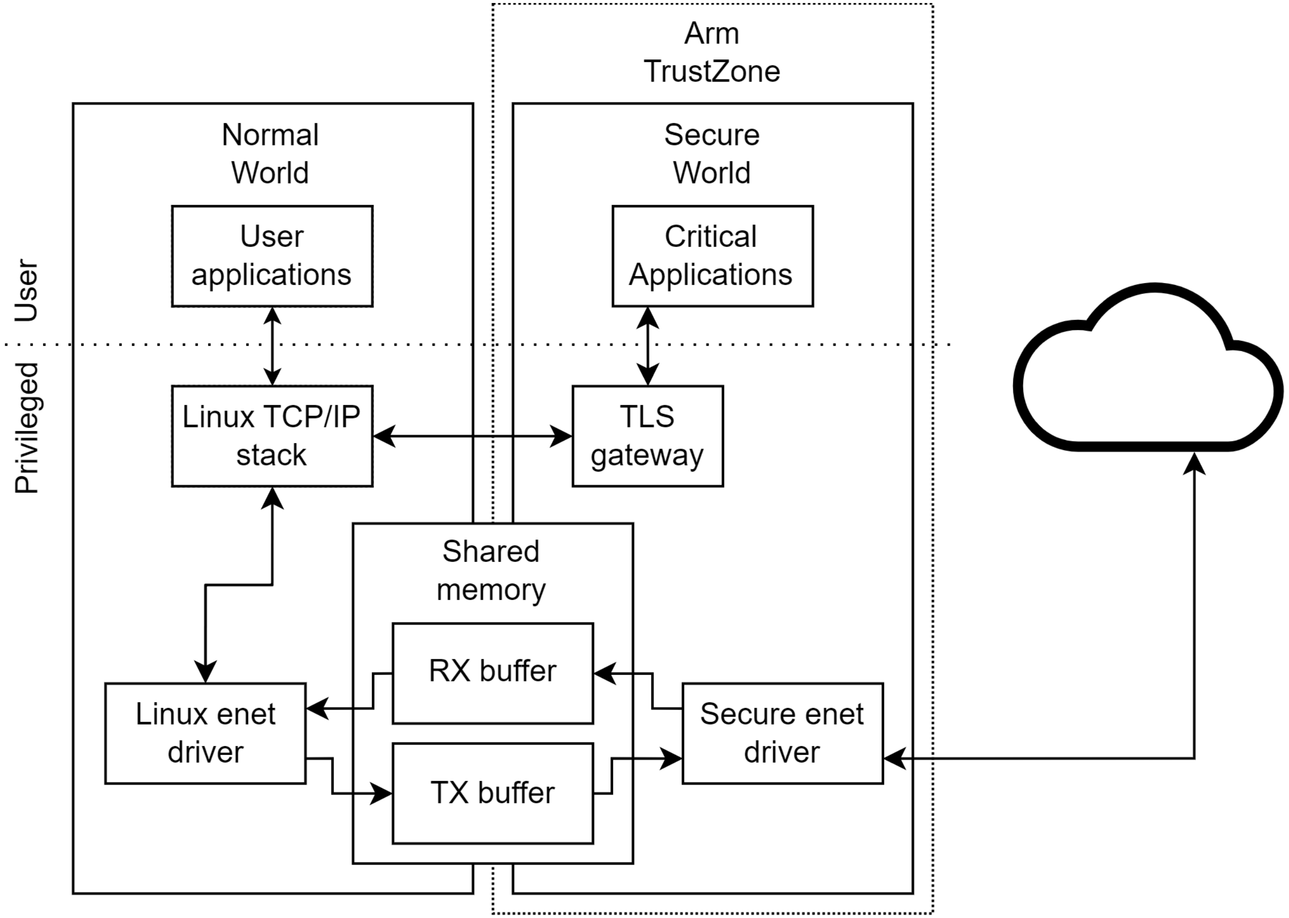
# The network peripheral

- Availability
  - Assign memory to SW
  - Assign interrupts to SW
  - Minimal driver in SW



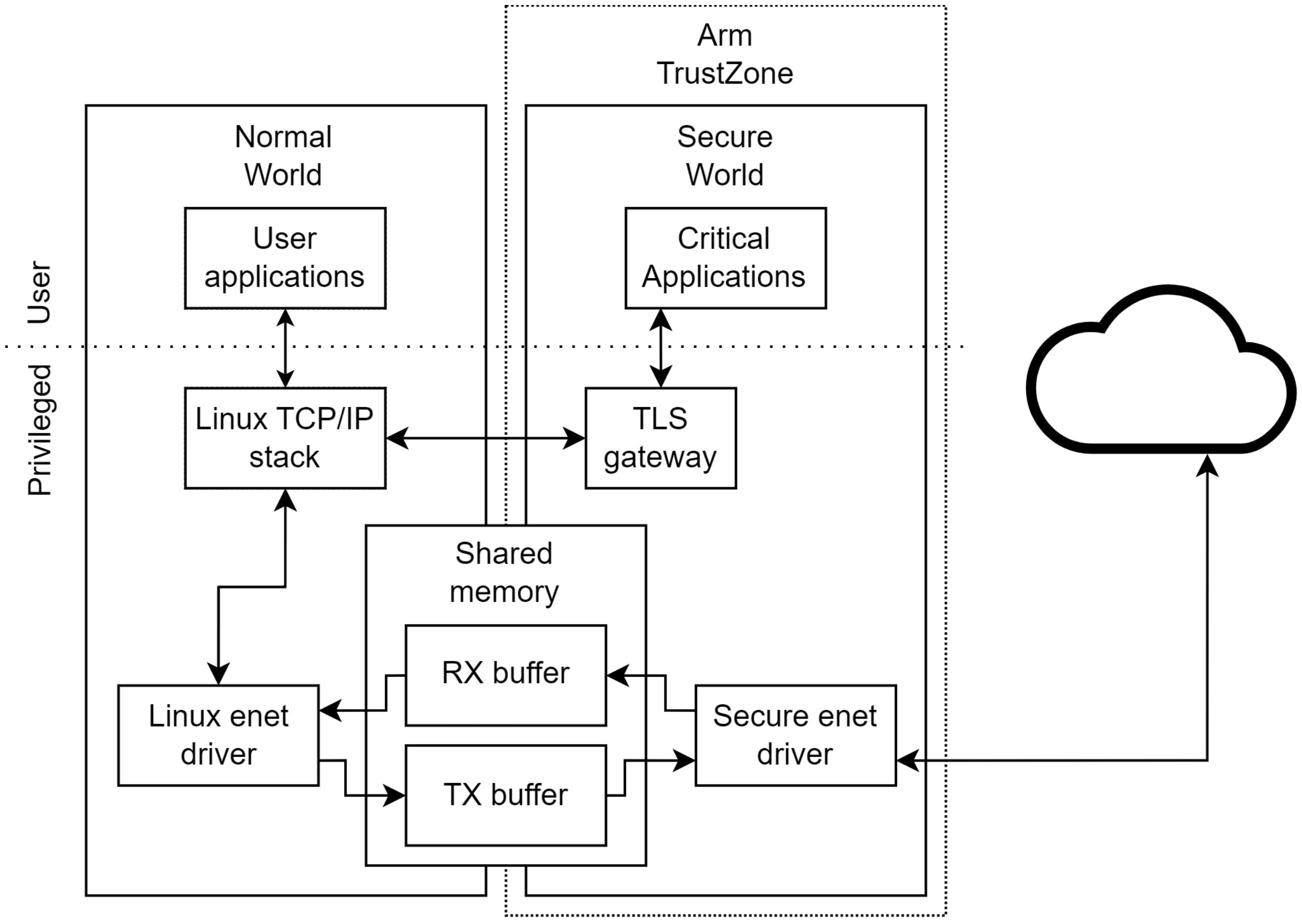
# The network peripheral

- Availability
  - Assign memory to SW
  - Assign interrupts to SW
  - Minimal driver in SW
- Sharing access with NW
  - Buffers in shared memory
  - Using Linux network stack



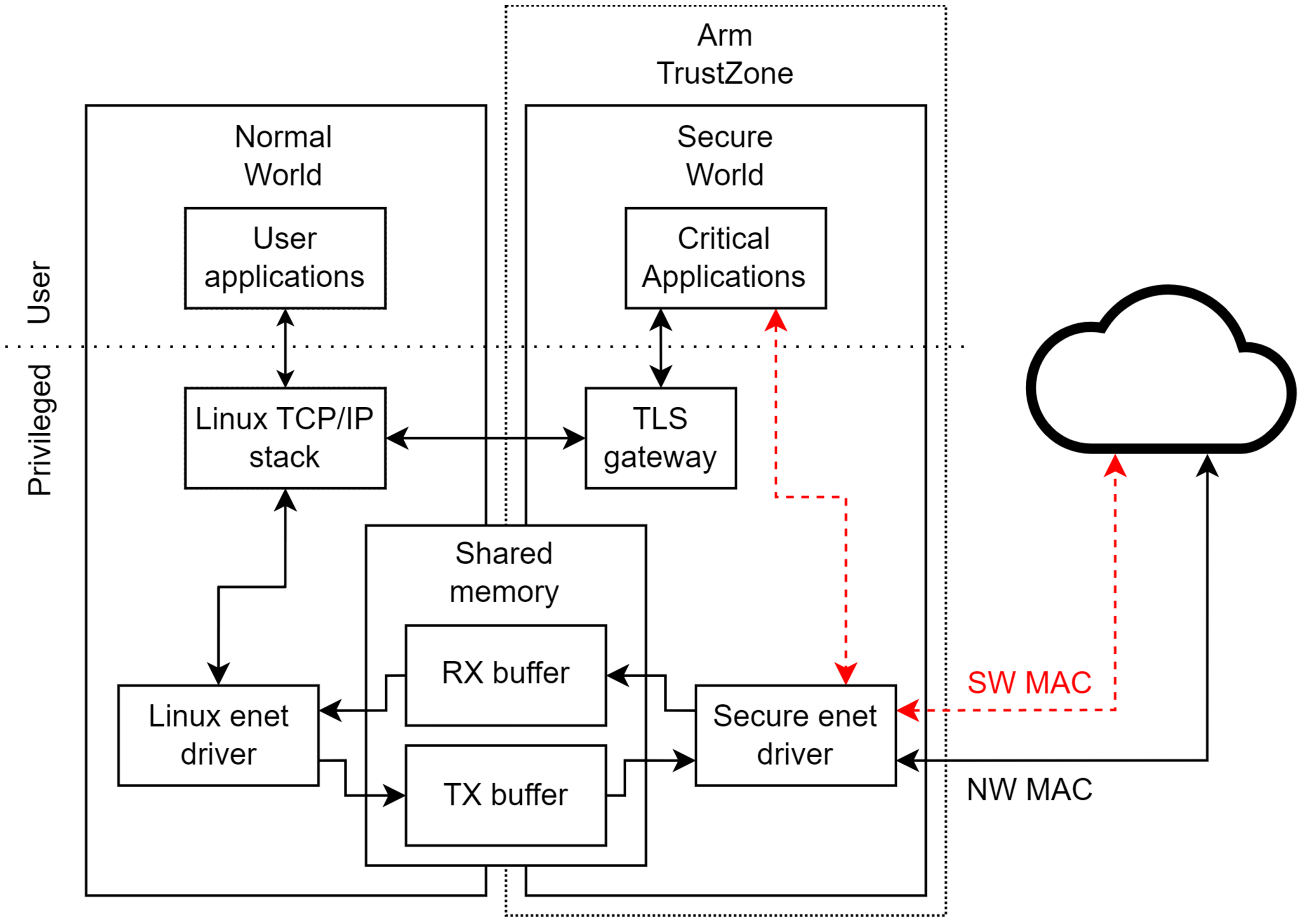
# The network peripheral

- Availability
  - Assign memory to SW
  - Assign interrupts to SW
  - Minimal driver in SW
- Sharing access with NW
  - Buffers in shared memory
  - Using Linux network stack
- Interrupt driven operation
  - Avoids overhead
  - Intelligent priorities



# The backup network

- Separate ip & mac address
  => Ensures reachability
- Reduced capabilities
  - Smaller attack surface
  - Local network only
  - ~400 LoC

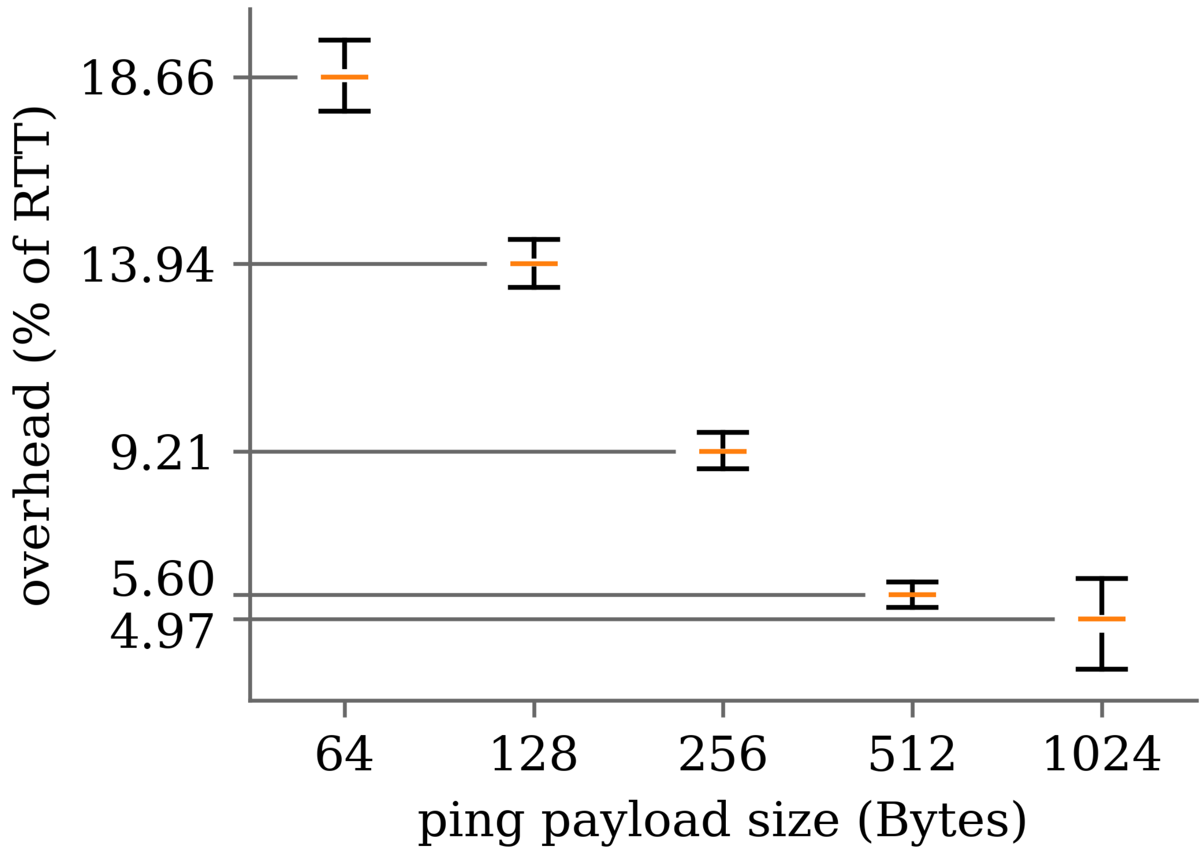


# Evaluation

- Proof of Concept
  - BD-SL-i.MX6
  - SPI Ethernet controller
  - Mr-TEE & OP-TEE OS
- Measured RTT using ping

# Evaluation

- Overhead
  - 64 bytes: 18,66%
  - 1024 bytes: 4,97%
  - Typical packet size: ~100B and ~2kB
- TCB size Driver
  - 418 LoC (0,1% of OP-TEE OS) <-> ~6k LoC Linux driver



# What’s next?

- Monitoring the state of the device remotely
- Controlling the device
- Recovering the device in case of attack
- Formal verification

# NetReach: Guaranteed Network Availability and Reachability to enable Resilient Networks for Embedded Systems

Tom Van Eyck, Sam Michiels, Xiaojiang Du, Danny Hughes

**Contact**
tom.vaneyck@kuleuven.be

**Source code**
https://gitlab.com/distrinet-netreach/documentation

With the support of VLAIO.
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101020416.