# The Security MicroVisor

## A Comprehensive Security Architecture for Multi-Application IoT Systems

…or how I learned to stop worrying and love virtualization…

**Danny Hughes, KU Leuven**

VersaSense

DistriNet

iMinds

KU LEUVEN

# IoT Devices and Applications

- Our work focuses on low-cost embedded IoT devices.
- IETF Class-1 device has 10KB RAM, 100KB ROM and wireless networking.
- Today's devices achieve 10 years lifetime on a single battery charge. Standards are progressing.

# IoT Devices and Applications

Precision farming and food production

Energy monitoring and control

Long-distance pipeline monitoring

*VersaSense uses the generic MicroPnP platform to support multiple applications.*

# Towards Fog Computing

- Applications and infrastructure may be provided by **different stakeholders**.
- Infrastructure is re-tasked to support new applications & increase return on investment.
- The ‘fog computing’ vision is essential to support city-scale deployments:
  - **No single player can provide everything**.

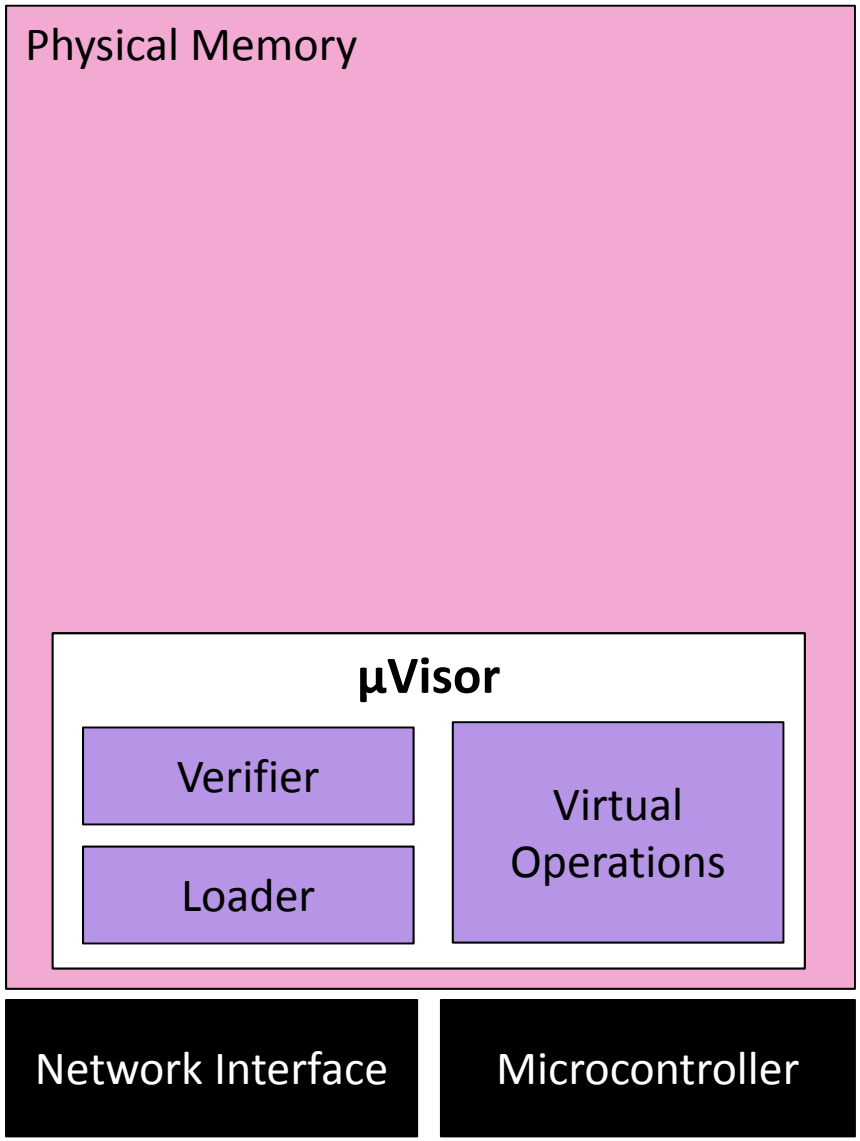
# The Problem of Malware

- IoT devices are **very easy to hijack** due to a lack of memory protection:
  - Over-the-air program installation.
  - Full mem
  - Interrupt
    over the network
- **Malware** i
  pp scenarios:
  - Concurrent applications must be isolated to protect both data and application logic from spying.
  - Applications must be completely removed when infrastructure is re-tasked.

Is it possible to fix this problem in software for all IoT devices?

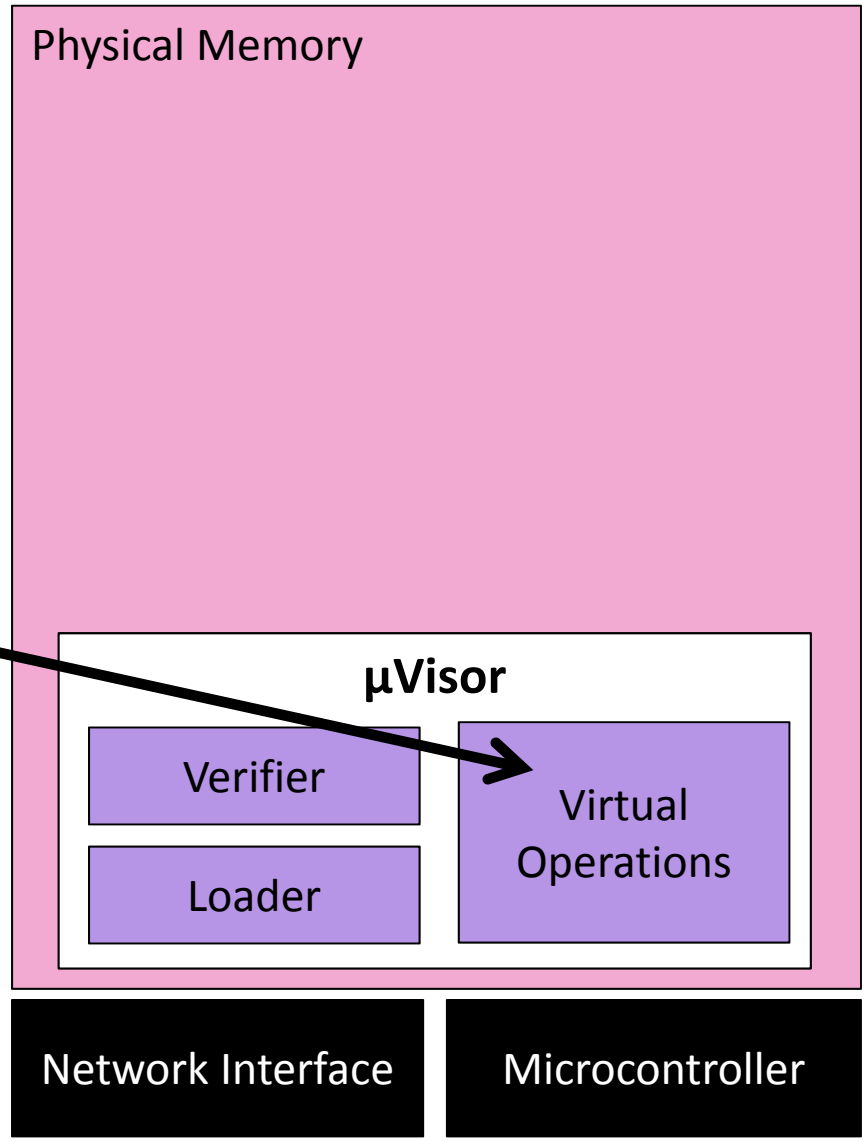
# The Security MicroVisor

- Partially virtualizes the microcontroller:
- Replace insecure ASM operations.
- Keep other ops to maximize speed.
- Provide remote security operations.



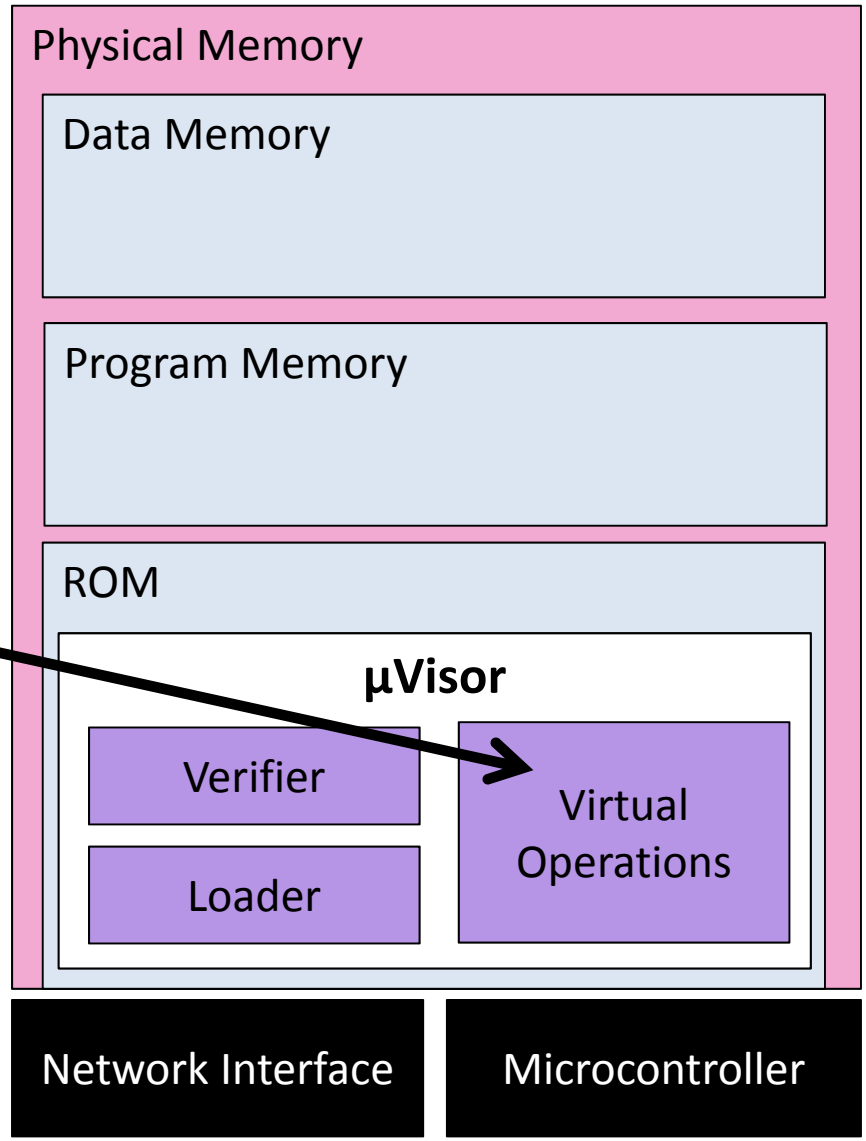
# The Security MicroVisor

- Secure memory operations modify architecture.



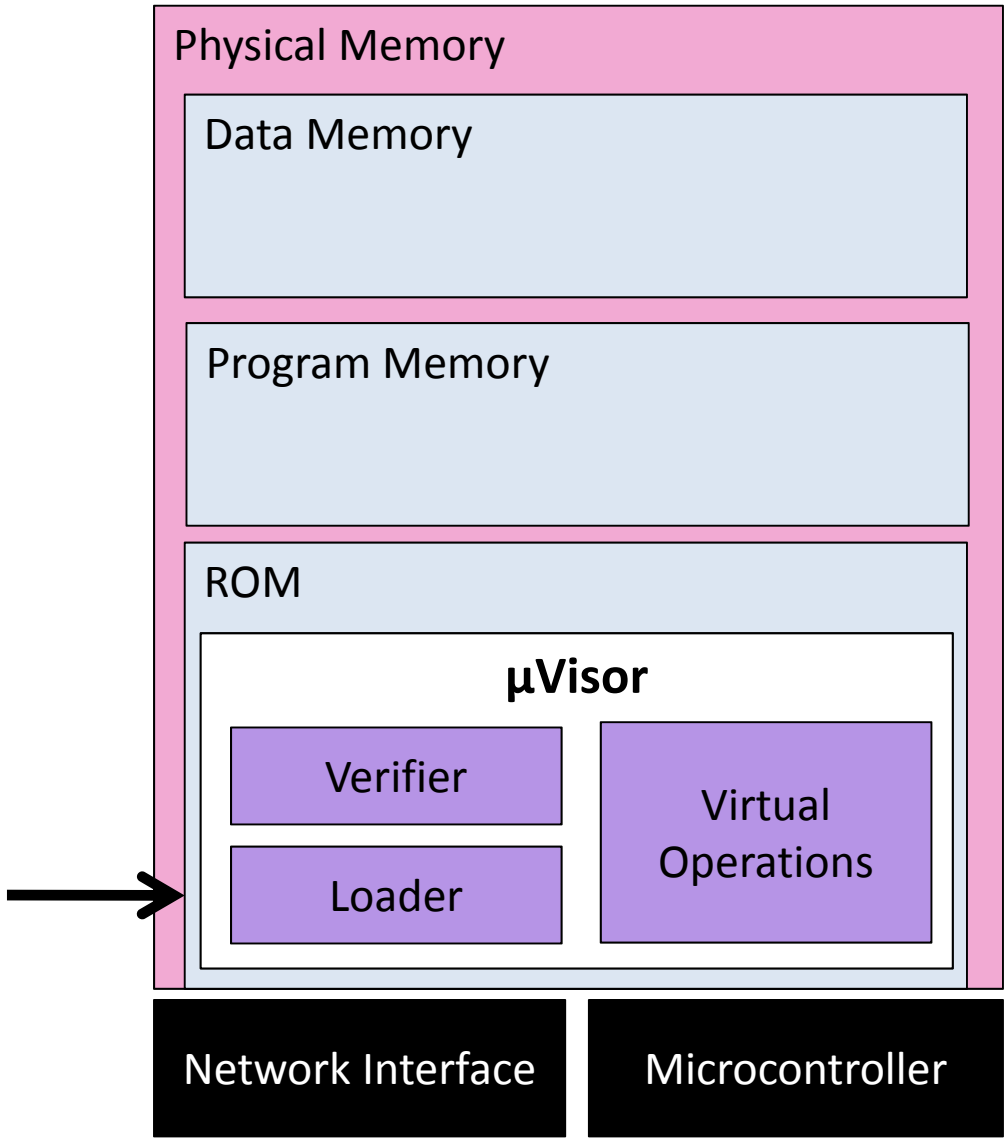
# The Security MicroVisor

- Secure memory operations modify architecture.



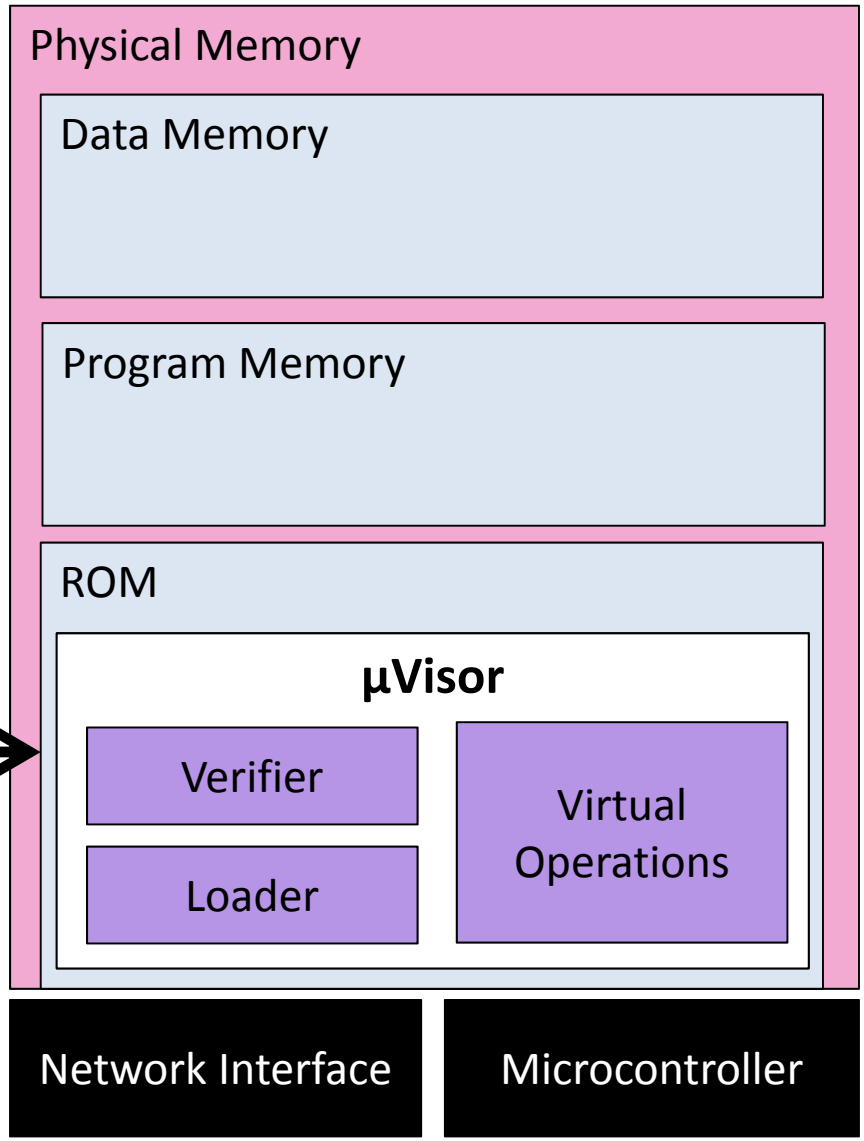
# The Security MicroVisor

- Loader accepts native code from network.



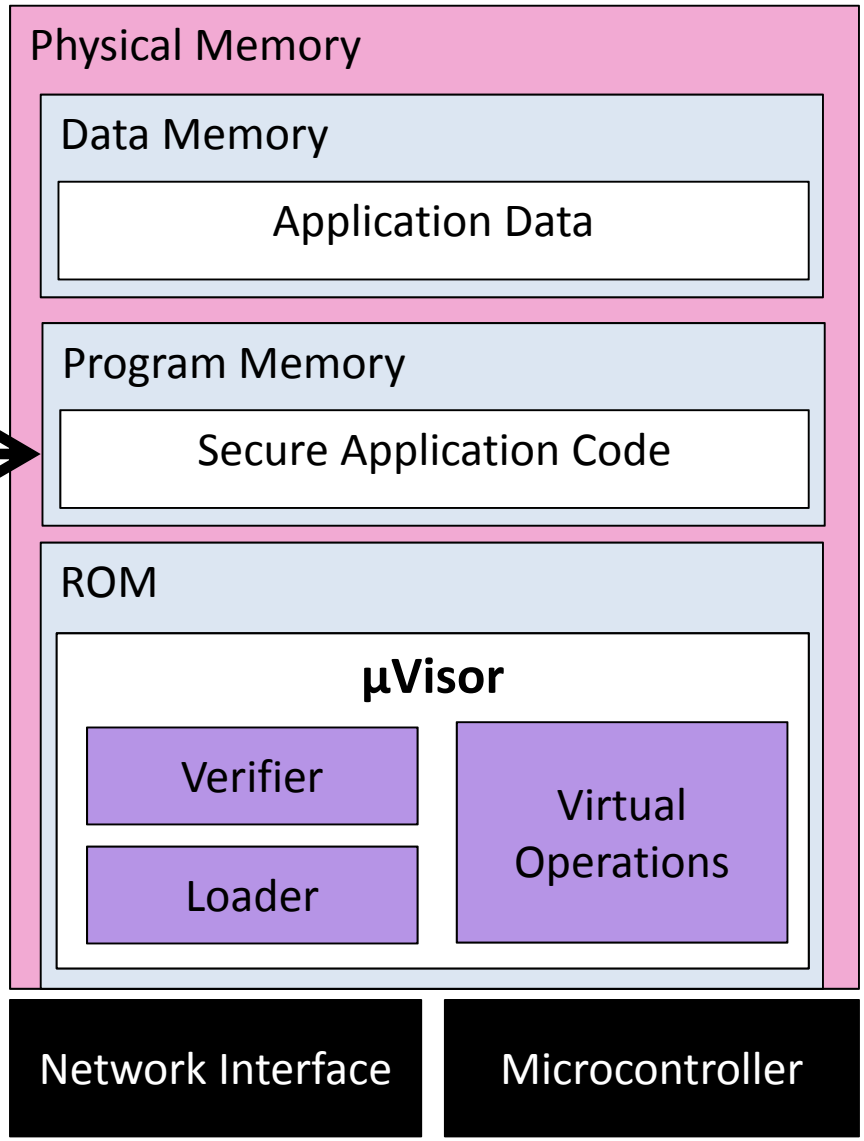
# The Security MicroVisor

- Verifier rejects code if insecure ops are present.



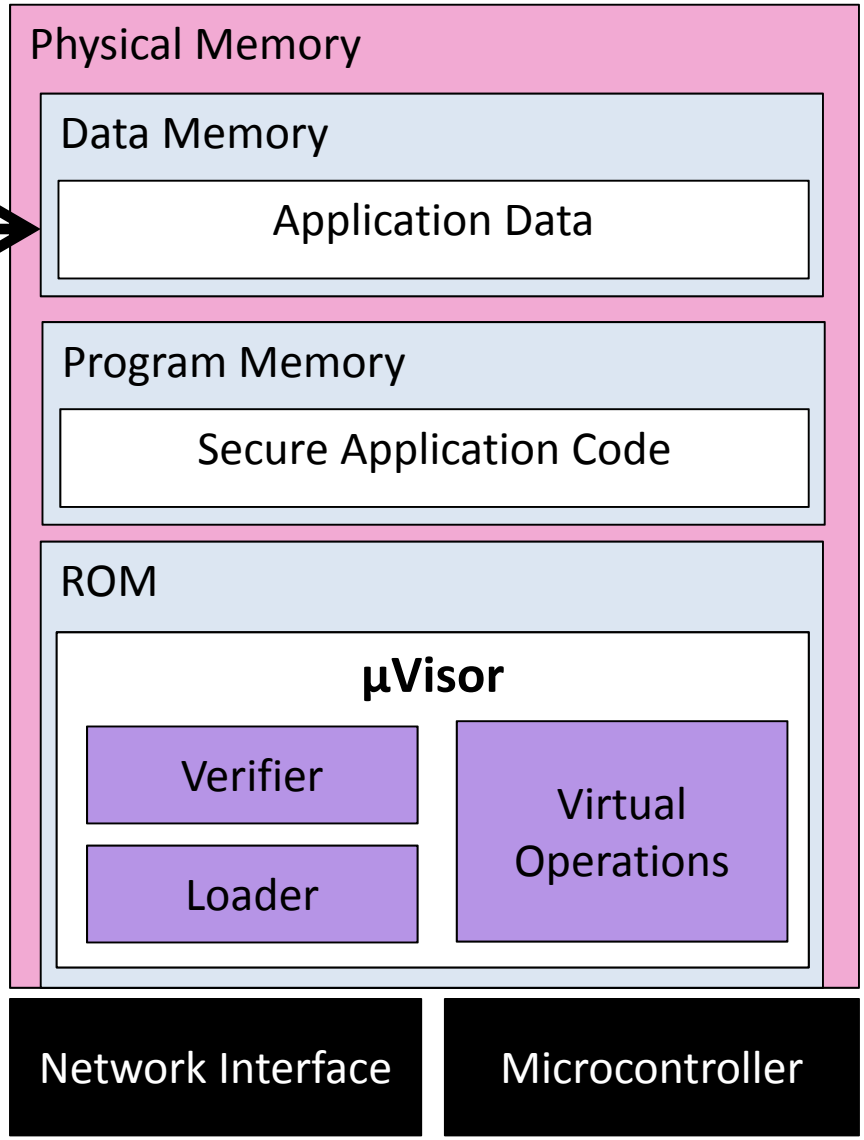
# The Security MicroVisor

- All code is verified free of insecure ops.



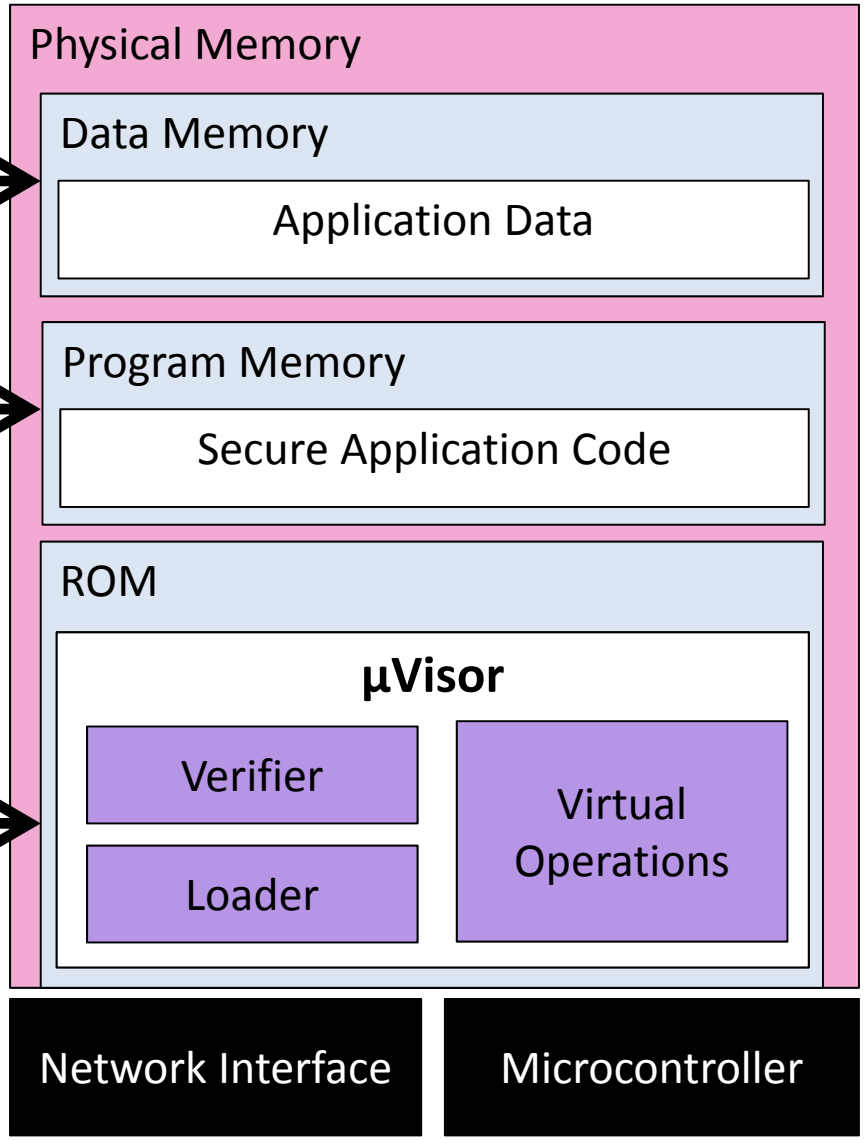
# The Security MicroVisor

- No requirements on app. data.



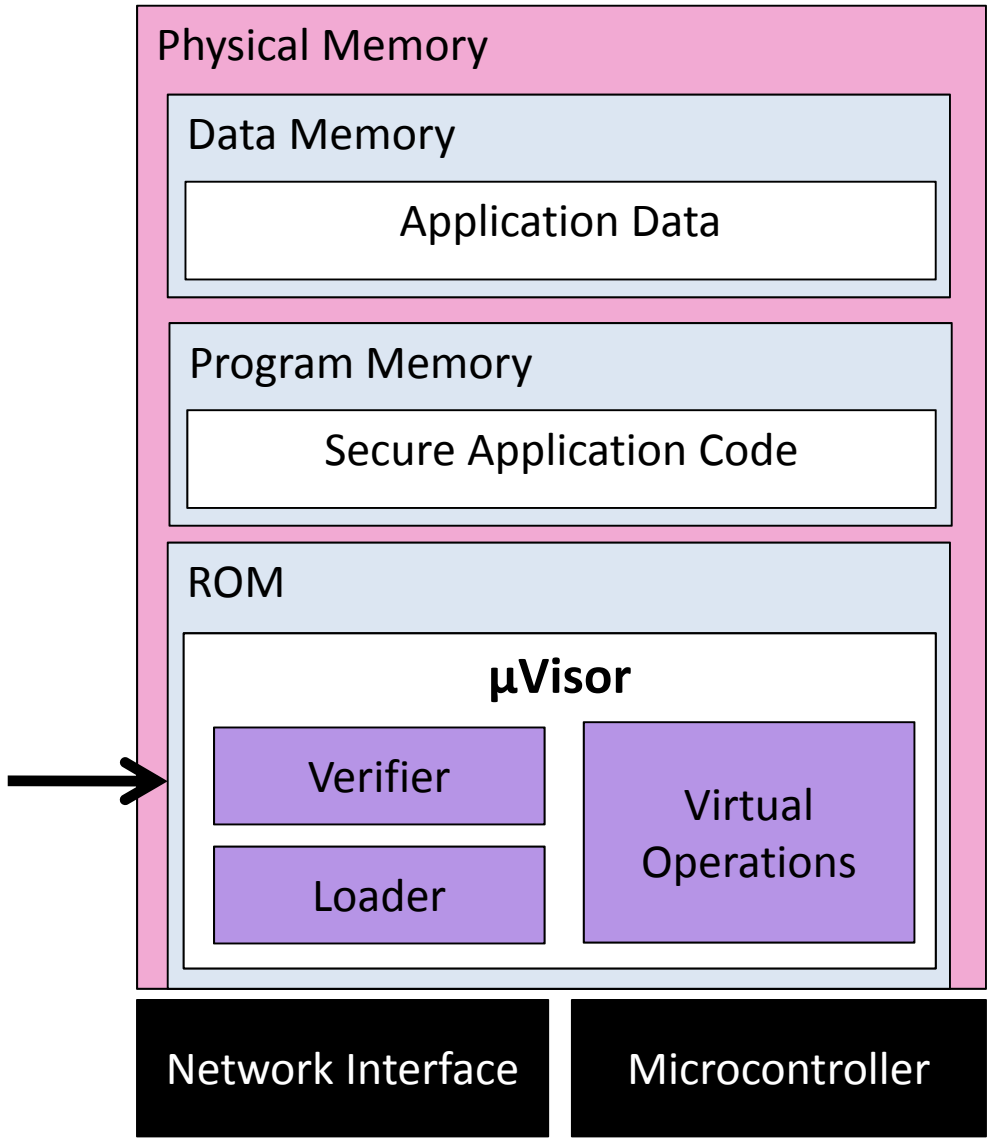
# The Security MicroVisor

- No execution
- No read or write
- No execution, reads or writes



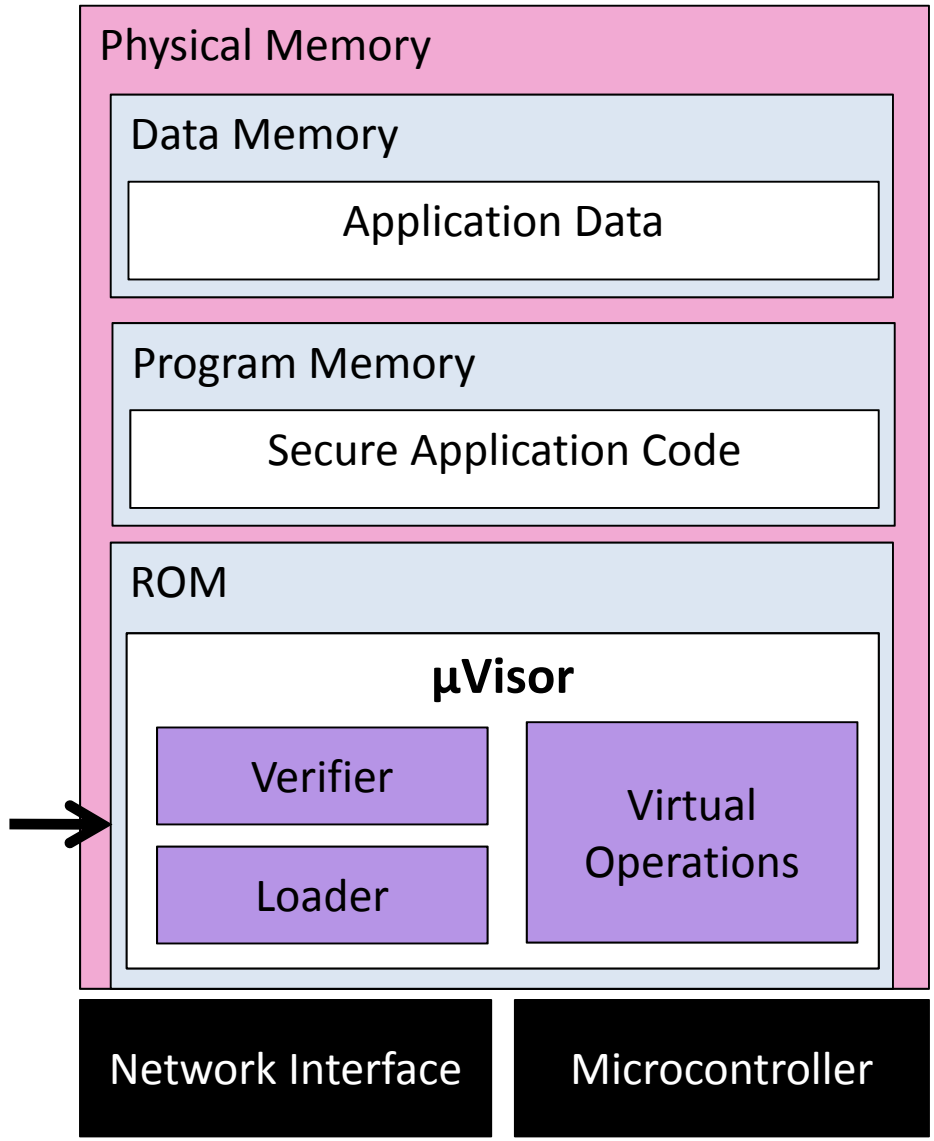
# The Security MicroVisor

- Interrupt management by µVisor, prevents hijacking.



# The Security MicroVisor

- Secure remote commands allow for control over apps.



# Tool-chain Support

- Minimal impact on development tool-chain:
  - Assembly post-processor replaces all insecure ops with calls to secure virtual functions in ROM.
  - Thin libraries are required to access virtualized interrupts.
- Zero hardware requirements on the MCU.
- The **compiler is not trusted**, all verification happens on the IoT device.

# What we Have Gained

- Protection against attacker will full network access who can write hand-crafted assembly:
  - No impersonation (secrets hidden in soft-ROM)
  - No hij
  - No ab into it (separ ).
- Evaluation shows **minimal impact on battery life or latency** in realistic scenarios.
- MicroVisor is implemented in a few KB of ROM.

Stop worrying and learn to love virtualization!

# Conclusion

- The Security MicroVisor can provide strong malware protection for all IoT devices.
- We think that there are many applications beyond malware…
- We will apply the technique in the next generation of VersaSense devices.
- Contact us:
  - Web: https://distrinet.cs.kuleuven.be/
  - Mail: danny.hughes@cs.kuleuven.be