linaro™ Arm Solutions at Lightspeed

PCI Device Authentication & Encryption

Mathieu Poirier

Linux Plumbers 2025

The Problem

- How to Authenticate SPDM-TDISP devices without building a monster
- Current work is on PCIe
- 2 contexts:
  - CMA/SPDM (component measurement and authentication)
  - PCI/TSM (Trusted VMs)
- 2 heuristics:
  - Local attestation
  - Remote attestation

Focus of the Presentation

- Device authentication is a complex topic that means different things to different people
- I want to talk about the integration of device authentication in the frameworks we have
- How device authentication is done is a conversation for another day

Two Patchsets

- [PATCH v2 00/18] PCI device authentication (Lukas Wunner)
  - A new version with signature artefacts sent via netlink socket is available
  - Proposes a framework for device authentication using the kernel's keyring mechanic
  - Exposes SPDM specifics via SYSFS
  - Centers on CMA/SPDM authentication (no TDISP state machine)
  - Does a lot of good things
  - Referred to as CMA/SPDM in this presentation

- [PATCH v8 0/9] PCI/TSM: Core infrastructure for PCI device security (TDISP) (Dan Williams)
  - Merged on Saturday, will be part of the 6.19 cycle
  - A framework to add devices in a TVM (TDISP state machine)
  - Centers on the support of trusted VMs
  - No support for device authentication (yet)
  - Does a lot of good things
  - Referred to as PCI/TSM in this presentation

Aiming for a Holistic Solution

- CMA/SPDM and PCI/TSM will need to coexist - right now they can’t (see next slide for reasons)
- PCI/TSM will need device authentication mechanic - CMA/SPDM has a solution for that
- I propose we integrated CMA/SPDM into PCI/TSM - this would give us:
  - Same representation of SPDM artefacts in sysfs (certificates and measurements)
  - Same user space notification process (netlink) for artefacts needed for signature verification
  - Same device probing policy
  - Same kernel authentication process

Aiming for a Holistic solution: Challenges

1. Sysfs SPDM artifacts need to be supplemented with measurements (same as slot[0-7])
2. We needed a flexible backend to populate SPDM artefacts in sysfs (DOE availability not guaranteed)
3. CMA/SPDM and PCI/TSM expose different “authenticated” entries in sysfs - this needs to be sorted out
4. Devices are currently probed when the TVM boots - “device_cc_probe()” in PCI core?
5. CMA/SPDM restarts device authentication when a device is reset → breaks PCI/TSM SPDM workflow

What else am I missing?

Device Authentication for PCI/TSM

1. Where should authentication take place in TVM?
   a. I'm proposing after ->lock()
   b. And before →accept()

2. Is a "uevent" acceptable to tell a user space agent that →lock() operation is finished?

3. How does the user space agent informs the kernel that a device has been authenticated?
   a. I'm proposing to use the same CMA/SPDM ".cma" keyring mechanic
      i. After 3rd party authentication, user space agent receives endorsement certificate of attested devices
      ii. User space agent adds the certificates to the ".cma" keyring
      iii. User space agent triggers a re-authentication of the device via sysfs
      iv. Successful authentication calls "device_cc_accept()"
      v. Device can be reprobed.