

PCI Data Object Exchange - Mediating access and related issues

DOE (PCI ECN) provides a standard mailbox definition, so far used for query / response type protocols. There can be multiple instances of a DOE on each PCI function, and each instance can support multiple protocols. Currently we have published definitions of the Discovery, CMA, IDE (available from the PCI SIG) and CDAT protocols (available from UEFI forum). Some of these protocols are intended for Linux kernel access (e.g. CDAT), others are less clear but there are possible use cases (CMA, IDE).

Patches to support DOE mailboxes in PCI extended config space have raised questions about how to ensure that these mailboxes, which may be of interest to various software entities (Userspace / kernel / firmware / TEE etc) can be safely used.

The DOE design does not easily allow for concurrent use by different software entities (even if possible, we cannot rely on other software elements doing this safely), so it seems some level of mediation is required. The topics for discussion include:

- 1) Do we want to enable any direct userspace access to these mailboxes or should we address on a per protocol basis (if at all)?
- 2) Do we need to 'prevent' userspace being able to access these registers whilst the DOE is in use?
- 3) How do we know the kernel should not touch a given mailbox (in use by other system software)? Perhaps a code first submission to ACPI to define a mediation mechanism? Is this sufficient for expected use cases? (What other suggestions do people have?)

A very brief overview of DOE and proposed kernel support will be presented to make sure everyone is aware of the background - then straight into the discussion of the above questions.

References:

https://lore.kernel.org/linux-pci/CAPcyv4i2ukD4ZQ_KfTaKXLYMakpSk=Y3_QJGV2P_PLHHVkpWfw@mail.gmail.com/
<https://lore.kernel.org/linux-pci/20210520092205.000044ee@Huawei.com/>

I agree to abide by the anti-harassment policy

I agree

Primary authors: CAMERON, Jonathan (Huawei Technologies R&D (UK)); WILLIAMS, Dan (Intel Open Source Technology Center)

Session Classification: VFIO/IOMMU/PCI MC

Track Classification: VFIO/IOMMU/PCI MC