Enhancements to IOMMU and VFIO User APIs for guest SVA

Linux Plumbers 2020 VFIO/IOMMU/PCI Track
Yi Liu <yi.l.liu@intel.com>, Jacob Pan <jacob.jun.pan@linux.intel.com>
Recap Guest SVA

- **Bind/unbind PASID**
  - Intel platform, PASID alloc/free will be required

- **Forward guest vIOMMU cache invalidation to host**

- **Page fault reporting and servicing**

- **Forward page response to host**
IOMMU UAPI Enhancements

• IOMMU UAPI was first merged in 5.3
  - Includes data structures used by guest SVA

• Enhancements:
  - Define the roles between IOMMU and VFIO UAPI
    ▪ VFIO: handles its own data and passes user pointer to IOMMU core
    ▪ IOMMU core: copies user data and then call into vendor driver
  - Add sanity checking of UAPI data based on argsz, flags
  - Added a new UAPI for reporting domain nesting info*
  - Document UAPI design and provide examples of interactions with VFIO
vSVA Upstream Plan

PCI device no page fault*

- Initial IOMMU UAPI
- Intel IOMMU driver extension to support guest SVA (bind/unbind gpasid, cache invalid)
- IOMMU user API enhancement
- IOASID extensions for guest SVA
- VFIO guest SVA extension

Page request service

- Intel IOMMU driver PRQ reporting support
- VFIO guest PRQ support

Mediated device

- IOMMU driver to support multiple fault handler data (per PASID) per handler
- VFIO extension to support reporting PRQ for mdev

*PCI device no page fault

- merged during 5.3 - 5.9
- under review
- not out
## Patches under review

<table>
<thead>
<tr>
<th>Patch series</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOMMU user API enhancement</td>
<td>v7</td>
</tr>
<tr>
<td>IOASID extensions for guest SVA</td>
<td>v2</td>
</tr>
<tr>
<td>vfio: expose virtual Shared Virtual Addressing to VMs</td>
<td>v6</td>
</tr>
<tr>
<td>intel_iommu: expose Shared Virtual Addressing to VMs</td>
<td>Qemu/rfcv9</td>
</tr>
</tbody>
</table>
Call for help for review 😊
Q&A
Backup
VFIO UAPI Extension

• VFIO cap nesting

/*
 * The nesting capability allows to report the related capability
 * and info for nesting iommu type.
 * The structures below define version 1 of this capability.
 * Nested capabilities should be checked by the userspace after
 * setting VFIO_TYPE1_NESTING_IOMMU.
 * @info: the nesting info provided by IOMMU driver.
 */
#define VFIO_IOMMU_TYPE1_INFO_CAP_NESTING  3

struct vfio_iommu_type1_info_cap_nesting {
    struct vfio_info_cap_header header;
    struct iommu_nesting_info info;
};
VFIO UAPI Extension (Cont.)

- **VFIO_IOMMU_PASID_REQUEST**
  - PASID allocate
  - PASID free

- **VFIO_IOMMU_NESTING_OP**
  - BIND_PGTBL
  - UNBIND_PGTBL
  - CACHE_INVLD
  - PAGE_RESP (not out yet)