Red Hat Virtualization 4.1 Hardware Considerations for Implementing SR-IOV

Hardware considerations for implementing SR-IOV with Red Hat Virtualization

Red Hat Virtualization Documentation TeamRed Hat

Red Hat Virtualization 4.1 Hardware Considerations for Implementing SR-IOV

Hardware considerations for implementing SR-IOV with Red Hat Virtualization

Red Hat Virtualization Documentation Team
Red Hat Customer Content Services
rhev-docs@redhat.com

Legal Notice

Abstract

This guide outlines hardware considerations for implementing SR-IOV with Red Hat Enterprise Linux, and for device assignment with Red Hat Virtualization.

Table of Contents

1. INTRODUCTION

Single Root I/O Virtualization (SR-IOV) is a hardware reference that enables a single PCI Express (PCIe) endpoint to be used as multiple separate devices. This is achieved through the introduction of two PCIe functions: Physical Functions (PFs) and Virtual Functions (VFs).

Physical functions are traditional PCIe functions that include SR-IOV capability, possessing full configuration and control of the PCIe device, including data movement. Each PCIe device can have between one and eight independent PFs.

Virtual functions are lightweight PCIe functions that contain the resources necessary for data movement and a minimized set of configuration resources. Multiple VFs can be created on each PF and each PF can support a different amount of VFs. The total number of VFs allowed is dependent on the PCIe device vendor, and is different between devices.

The PCIe specification supports greater numbers of VFs through the implementation of Alternative Routing ID Interpretation (ARI), which reinterprets the device number field in the PCIe header allowing for more than eight functions. This translation relies on both the PCIe device and the port immediately upstream of the device, whether root port or switch, supporting ARI.

The system firmware (BIOS or UEFI) allocates resources, including memory, I/O port apertures, and PCIe bus number ranges, for the PCIe topology. As such, SR-IOV must be supported and enabled by the firmware in order to allocate sufficient resources.

1.1. Summary of Hardware Considerations for SR-IOV

- Firmware (BIOS or UEFI) must support SR-IOV. Check if the extension is enabled by default. If not, enable it manually. This is similar to enabling the virtualization extension (VT-d or AMD-Vi). Refer to vendor manuals for specific details.
- Root ports, or ports immediately upstream of the PCIe device (such as a PCIe switch), must support ARI.
- PCIe device must support SR-IOV.

Refer to vendor specification and datasheets to confirm that hardware meets these requirements.

The **`lspci -v`** command can be used to print information for PCI devices already installed on a system.

2. ADDITIONAL HARDWARE CONSIDERATIONS FOR USING DEVICE ASSIGNMENT

Device assignment provides the capacity to assign a virtual guest directly to a PCIe device, giving the guest full access and offering near-native performance. Implemented in conjunction with SR-IOV, a virtual guest is directly assigned a VF. In this way, multiple virtual guests can be directly assigned to VFs of a single PCIe device.

SR-IOV does not need to be enabled to directly assign virtual machines to PCIe devices, nor is device assignment the only application for creating VFs, however the two features are complementary and there are additional hardware considerations if they are to be used together.

Device assignment requires I/O Memory Management Unit (IOMMU) support in the CPU and firmware. The IOMMU translates between I/O Virtual Addresses (IOVA) and physical memory addresses. This allows the virtual guest to program the device with guest physical addresses, which are then translated to host physical addresses by the IOMMU.

IOMMU groups are sets of devices that can be isolated from all other devices in the system. IOMMU groups represent the smallest sets of devices with both IOMMU granularity and isolation from all other IOMMU groups within the system. This allows the IOMMU to distinguish transactions to and from the IOMMU group while restricting direct memory access (DMA) between devices outside of the IOMMU group and the control of the IOMMU.

Isolation of transactions between the virtual guest and the virtual functions of the PCIe device is fundamental to device assignment. Access Control Service (ACS) capabilities defined in the PCIe and server specifications are the hardware standard for maintaining isolation within IOMMU groups. Without native ACS, or confirmation otherwise from the hardware vendor, any multifunction device within the IOMMU group risks exposing peer-to-peer DMA between functions occurring outside of the protection of IOMMU, extending the IOMMU group to include functions lacking appropriate isolation.

Native ACS support is also recommended for the root ports of the server, otherwise devices installed on these ports will be grouped together. There are two varieties of root ports, processor-based (northbridge) root ports and controller hub-based (southbridge) root ports. As above, if device assignment is used in conjunction with SR-IOV, and virtual guests are being assigned to VFs, then these ports must support both ACS and ARI.

Intel’s Xeon Processor E5 Family, Xeon Processor E7 Family, and High End Desktop Processors include native ACS support on the processor-based root ports.

Intel Platform Controller Hub-based (PCH) PCI Express Root Ports currently either do not support ACS or make use of non-standard ACS implementations, making fine-grained isolation of devices connected via these Root Ports difficult. Many of these Root Ports do support ACS-equivalent functionality. The Red Hat Enterprise Linux 7.3 kernel includes support for enabling this ACS-equivalent functionality on X79, X99, 5-series though 9-series, as well as 100-series PCI Express chipsets.

Refer to vendor specification for determining processor-based and controller hub-based root ports when installing PCIe devices to ensure the root port supports ACS.

In addition, any PCIe switch or bridge within the I/O topology also requires ACS support, otherwise it may extend the IOMMU group.

2.1. Summary of Hardware Considerations for Device Assignment

- CPU must support IOMMU (for example, VT-d or AMD-Vi). IBM POWER8 supports IOMMU by default.
- Firmware must support IOMMU.
- CPU root ports used must support ACS or ACS-equivalent capability.
- PCIe device must support ACS or ACS-equivalent capability.
- It is recommended that all PCIe switches and bridges between the PCIe device and the root port should support ACS. For example, if a switch does not support ACS, all devices behind that switch share the same IOMMU group, and can only be assigned to the same virtual machine.

Refer to vendor specification and datasheets to confirm that hardware meets these requirements.

The **`lspci -v`** command can be used to print information for PCI devices already installed on a system.