Direct-Assignment Networking Fault Isolation in a Data Center Environment

Application Note

Ethernet Products Group (EPG)

January 2022

Revision 1.0
655276-001

Revision History

Revision	Date	Comments
1.0	January 20, 2022	Initial release.

Contents

NOTE: ***This page intentionally left blank.***

1.0 Introduction

Customers with high performance or low-latency data center applications may use a technique called *direct device assignment* to install Ethernet networking interfaces directly into virtual machines allowing direct hardware access. Either an entire Ethernet device or an Ethernet port of a multi-function Ethernet device may be allocated to virtual machines through hypervisor-specific direct assignment methods.

Alternatively, with Ethernet devices supporting Single-Root I/O Virtualization[1] (SR-IOV) technology, SR-IOV Ethernet virtual functions may be installed into virtual machines. Using SR-IOV, multiple virtual machines can share a common Ethernet uplink, with each virtual machine capable of similar low-latency, high performance Ethernet networking services.

Virtual machine direct access to Ethernet hardware resources may pose both reliability or availability concerns to the network interface or the platform itself. In SR-IOV Ethernet configurations, malfunctioning software or malware running in a virtual machine could temporarily disable or disrupt the direct-assigned or virtualized Ethernet functions on the shared networking interface.

If an entire device interface is assigned to a virtual machine, any device-reported faults can be directly attributed to the virtual machine. However, in the case of SR-IOV, some network data processing faults triggered within the Ethernet interface by the virtual machine through the virtual function may be difficult or impossible to attribute to the original virtual machine.

This document discusses some methods that can be used to detect, isolate, and remediate such malfunctioning or malicious virtual machines from the data center operating environment.

The following configurations are addressed:

- Ethernet devices directly assigned exclusively to a virtual machine
- Ethernet functions of a multi-function Ethernet device assigned exclusively to a virtual machine
- Ethernet virtual functions (VFs) of an SR-IOV Ethernet device assigned exclusively to a virtual machine.

Virtual machines can generate faults in these configurations. As a side-effect, the Ethernet device will appear to fail temporarily or generate system faults that disrupt physical platform operation.

1.1 Direct Assignment and SR-IOV Background

During physical device enumeration by the guest operating system, a hypervisor creates the existence of a “PCI” device at a specific PCI segment/bus/device/function offset. The combination of the PCI segment, bus, device, and function form a unique Requester ID for the device in the system. PCI configuration cycles from the virtual machine are trapped (by hardware assist) by the hypervisor. The hypervisor emulates the PCIe transaction responses to advertise whatever type of device is desired to be installed in the virtual machine.

The hypervisor exposes a network controller as identified by the PCI vendor and device ID fields in PCI configuration space. This exposure triggers a built-in guest OS-resident device driver to load. The loaded driver is aware it is hardware-virtualized and accesses memory-mapped or port I/O ‘registers’ on the device—the memory mappings established by the hypervisor prior to guest access—to directly access hardware including devices exclusively assigned to virtual machines.

Furthermore, SR-IOV enables hardware resource partitioning to share a single hardware device among multiple virtual machines. Rather than transiting the hypervisor to transmit a frame, for example, the

1. *PCI-SIG SR-IOV Primer - An Introduction to SR-IOV Technology*

virtual machine can post the frame directly to the physical network interface, reducing latency and CPU utilization in the process. A network example of this model is illustrated in Figure 1.

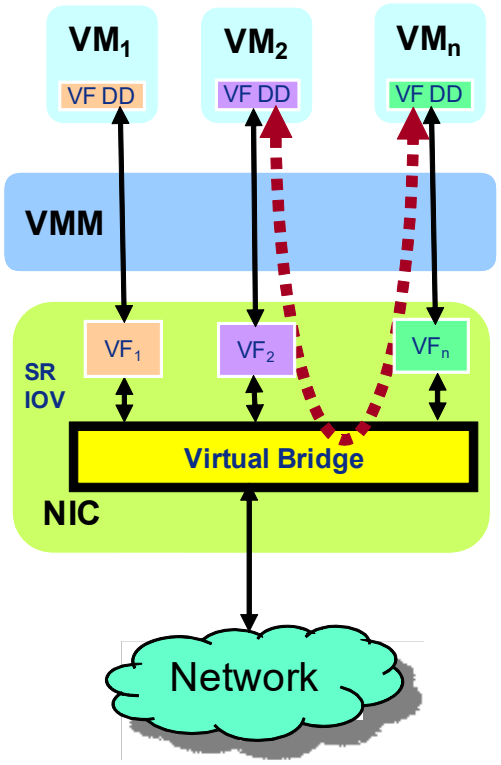

Figure 1. SR-IOV Networking

1.2 Malformed I/O Requests and Device Response

Malformed I/O requests come in several forms:

- Unexpected accesses from the device to the host
- Invalid requests from host software to the device
- Unexpected accesses from the host CPU to the device

Malformed I/O requests may halt data processing by the Ethernet device, which often appears to the system operator as an intermittent transmit or receive hang. The hang may be limited to the specific virtual function in the case of SR-IOV.

Malformed I/O requests result when either one or more invalid parameters are supplied to the device. As one example, a VF driver in a virtual machine may specify an out-of-range guest physical address (GPA) for a transmit data buffer. When the device attempts to access this data buffer, the device encounters an error returned by the system.

1.3 Device-Generated Advanced Error Reporting Events

Malformed I/O requests can also impact the overall system operation either triggering time consuming error handling flows or possibly causing system resets. Software running on the host CPU may inadvertently or purposefully violate the expected device access model. Some examples of these include host software attempting to read or write to non-existent memory offsets or generating misaligned accesses to device registers. In these cases, Intel® Ethernet devices generate **Advanced Error Reporting** (AER) indications in-band to the system chipset as either Unsupported Requests (URs) or Completer Aborts (CAs).

Some system configurations have been observed to stall at the system hypervisor-level, or possibly reset the entire platform in response to device-generated AER events. As AER events report general system data integrity issues such as uncorrectable data errors, some platforms perform extensive Reliability-Integrity-Serviceability (RAS) integrity checks in response to any AER event[1]. While the RAS routines execute, the system may appear to stall. An AER event by itself is not necessarily an indication of malicious or faulty system software.

Other observed system configurations do not configure AER event handling altogether. In this case, the default chipset behavior promotes an AER event to a non-maskable interrupt (NMI), leading to a system reset. Customers are advised to evaluate the AER functionality of any platform configuration—a combination of platform firmware as well as system software—prior to deploying a system into production.

In the case of SR-IOV, the hypervisor physical function driver detects the device hang condition and automatically restarts the device. For directly assigned devices where the virtual machine manages the entire operational state of the device, the virtual machine would be responsible for restarting the device.

2.0 Root Cause Determination Methods

Virtual machine generated device faults would manifest themselves in the following ways:

- Device hangs and driver attempts reset
- AER events reported by the system

In some cases, the hypervisor physical function driver can detect and log the specific function or virtual function attributed to a fault. Many devices implement AER through PCIe-extended configuration space and the Advanced Error Reporting Capability structure. SR-IOV devices (VFs) have per-VF PCIe configuration status registers that report VF-specific AER indications. For reference, see the *Intel® 82599 10 GbE Controller Datasheet* describing the VF Device Status register located at VF Configuration Space offset AAh, Bit 3 (“Unsupported Request Detected”).

The system chipset may have methods to capture diagnostic information of the failing transaction. Platforms report AER events in a variety of ways: system firmware logs[2], firmware errors, interrupts to notify the hypervisor, or system software-based AER handlers[3].

In the case of direct assignment of entire ports or devices to a virtual machine for AER events affecting overall system performance or operation, collecting the device PCIe requester ID (RID) associated with an AER event uniquely identifies the affected virtual machine. The requester ID explicitly identifies the device, and the device is explicitly assigned to a virtual machine. A device hang limited to the exclusive virtual machine where the virtual machine is responsible for resetting the device in response, is out of scope for this document.

1. *Server RAS and UEFI CPER* – Spring 2017 UEFI Seminar and Plugfest.
2. *System Event Log (SEL) Troubleshooting Guide*
3. *The PCI Express Advanced Error Reporting Driver Guide HOWTO* (Linux)

3.0 Isolation Methods

Prior to removing the platform from service, the data center operator may attempt to migrate workloads[1] away from a suspected failing platform. Insufficient information may be available on the platform at the time of migration to uniquely identify a virtual machine that is generating the hardware faults.

Using data center operation methods, a data center operator should attempt to isolate the fault by tracking the migration of workloads to other platform resources. If platform faults follow a workload or virtual machine as it is migrated within the data center, the virtual machine should be flagged for remediation

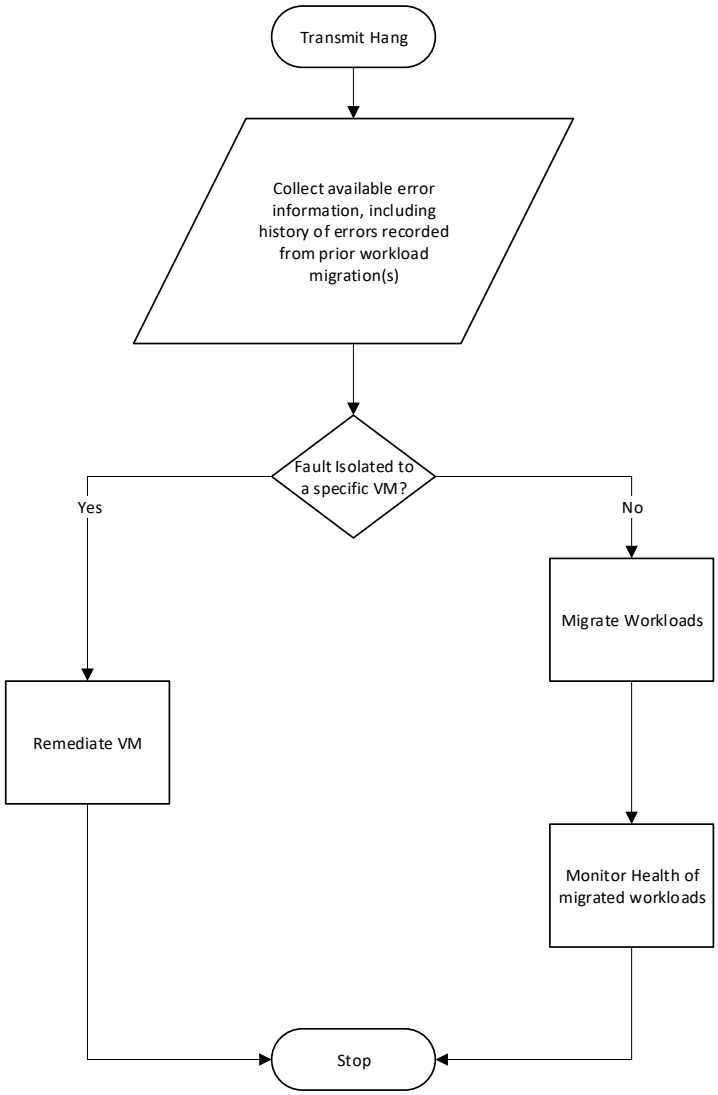

Figure 2. Workload Isolation Flow

1. Refer to *Migrating Virtual Machines* (VMware), *Configure Live Migration and Migrating Virtual Machines without Failover Clustering* (Microsoft), or *Migrating Virtual Machines* (Red Hat).

4.0 Remediation Methods

The specifics of remediation may vary from data center to data center. At a high-level, a virtual machine should be suspended or paused from execution. Subsequently, the virtual machine state and integrity should be examined and verified.

Intel recommends using the latest available Intel® device drivers. They may be downloaded from the Intel Download Center - Ethernet Products website or supplied as part of an operating system distribution or driver pack from other vendors. Contact your platform system vendor for system-specific driver updates as needed.

NOTICES & DISCLAIMERS

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors which may cause the products and services to deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

*Other names and brands may be claimed as the property of others.