Call details

Wed, May 2nd, UTC 16:00-17:00
Wed, May 2nd, BST 17:00-18:00
Other times

On-line:
https://www.gotomeet.me/LarsKurth

Phone:
Access Code: 906-886-965
United Kingdom: +44 330 221 0088
United States: +1 (571) 317-3129
Australia: +61 2 9087 3604
China (Toll Free): 4008 811084
Other numbers: see
https://lists.xenproject.org/archives/html/xen-devel/2018-04/msg01809.html

Attendees

Alexey G
Stefano Stabellini
Paul Durrant
Lars Kurth
Roger Pau Monne
Chao Gao
Christopher Clark
Julien Grall
Daniel Smith
Tamas K Lengyel
Marek Marczykowski-Górecki
Rich Persaud

Agenda

Roger: Q35 (chipset) HVM emulation, adding MCFG support to guests:

- Alexey: role of QEMU in PCI and MCFG emulation.
- Alexey: emulation of specific chipset registers: *DRAM Controller Registers (D0:F0)* PCIEXBAR, which controls the position of the MCFG.
- Alexey: size of the MMIO hole. Related to passthrough and how to fit BARs of PCI devices below the 4GB boundary.

Alexey: Providing support for Q35 - Alexey posted a proposal
Roger, Paul: looked at the proposal. More or less OK with it but would like to discuss some of the details to break the work into smaller portions. Some stuff could be deferred to later.

Roger: already replied.
Alexey: agrees
Alexey: have different features we need to support (besides bridges)
Alexey: can focus on support for reading the extended capabilities, which is my main task. The task is to use the extended capabilities, while being able to use multiple emulators. Need to choose a long-term solution alongside the short-term.

Described some possible ways forward in my email

Paul: went through the options - favouring option 2 (implement in Xen). Should not require that much code

Alexey: depends on how deep the emulation goes. Could just implement north bridge device. Or can implement ICA? and ICH line

Paul: Do we need to emulate south bridge at all?

Alexey: This may break the QEMU configuration. Would be hard to extend it further

Roger: looked at MCH - which is the only thing we would need to implement.

There was a discussion about size of some registers.

Roger: not clear why we need to emulate MCH registers initially.

Next steps:

- Agreed to go for a very simple implementation initially as a proof of concept
- Try to get together some patches with Q35 emulation in Xen and see how well it works with QEMU
- Paul to send a patch to QEMU to cleanup/improve forwarding of PCI config space accesses.
- **Roger to Alexey:** would you be OK to send a more detailed design proposal to the list?
- Alexey: agreed to continue discussion on mailing list

Roger: PVH/HVM guest pci-passthrough: using the internal vPCI infrastructure

One month ago sent initial patches to do pci-emulation withion Xen. Those at the moment are only used by PVH Dom0. Cover the missing pieces for Dom0 and DomU

- Dom0: Full access for the config space - fine for Dom0, but not acceptable for DomU
- Dom0: support for SR-IOV express capability.
- DomU: Blacklist all the PCI capabilities we don't know about (not very hard to). At the moment Xen only knows about MSI, MSIX and MSI headers.
- DomU: change config space logic to reject accesses to regions not handled by Xen.
- DomU: prevent DomU from relocating BARs.

Next steps for Roger:

- Dom0: support for SROIV express capability.
- Then looking into using this infrastructure for DomU.
- Want to synchronise with ARM

Julien: how do you hide a device from Dom0 when we do passthrough? How do you specify what capability are accessible to Xen at boot time

Roger: there’s no way to currently hide a device from Dom0 (neither for PV or PVH). Dom0 has access to all the capabilities. On PVH Xen could hide devices from Dom0 by preventing Dom0 to access the configuration space of certain devices.

Julien: where would the reset code live then?

Christopher: would want to avoid Dom0 having access to the config space. The VM hosting the toolstack will need to exercise control over access to the config space.

Roger: Another option would be to do this inside of Xen via a hypercall

Julien: moving reset from Linux into Xen would be quote complex.

Paul: Handling the reset and quirks within Xen seems perfectly reasonable

Christopher: handling the sequence to reset the device is quite complex

Stefano: Aside from who does what are there any specific requirements we need to pay attention to for complex devices such as GPUs (such as IOMMU mapping)

Alexey: saw devices which do not like secondary bus reset (e.g. some NVIDIA GPUs) - When we use the device and restart the domain, it will hang during boot.

Roger: know there are issues with some devices.

Stefano: Surprisingly high number of quirks. So the question is who maintains the quirks. If we moved it to Xen, we may not get contributions to fix quirks. We would have to monitor Linux and then move code, which increases the codsize

Roger: The code would be somewhere in any case, either Xen or Dom0 kernel: so why does the codesize matter?

Daniel: the code size does not go away, but the question is how it can be isolated

Stefano: depending on where it is, the stability of the system is directly impacted

Alexey: need to provide device specific quirks to reset the device

Alexey: Have not looked at Linux quirks for resetting devices. Reset is mandatory (must be performed in many cases such as domain restart, …). Can move from secondary reset to other reset methods and work around specific quirks.

Rich: Mentioned that Oracle posted some reset code recently for XenClient into Linux.

Next steps:

- Should we start a discussion on the mailing list on how to resolve the reset question. ACTION: Rich to start the thread (the people participating in the reset discussion to be CC'ed)

Stefano/Julien: ARM guest pci-passthrough

Julien: the idea was not really speaking about PCI passthrough, but to follow what is happening on ARM. Don't have any specific things to talk about.

Stefano: The challenge on ARM has been a few incompatible implementations in the config space. Initially we didn't know what to do. We then decided to start simple and implement the standard compliant functions in the HV. And then cross the bridge of incompatible config space registers when we come to it.

Julien: mostly looking on what is going on. Not currently working on PCI passthrough

Roger: asks whether suitable for ARM

Julien: in principle yes, but the different implementations (e.g. for timers). IOMMU may not translate all the hardware (some commands may bypass). Not sure whether the same challenge exists on x86.

Rich: discuss the level of security support that will be asserted in SUPPORT.md for driver domains which contain untrusted PCI devices.

- Will Xen security support be different for SR-IOV devices? GPUs vs. NICs?
- There have been past discussions on this topic and a proposed PCI-iommu-bugs.txt file to help Xen users and developers understand the risks [2][3][4] that may arise from a hostile device and potentially buggy firmware. If we can document specific risks, we can ask firmware developers to make specific improvements to improve the security of PCI emulation.
- There is an active effort [4] underway to improve firmware security in servers (and eventually desktops), including a reduction of attack surface due to SMM. There is also work underway [5][6] to perform secure boot between individual PCI devices and server motherboards. Some of these concepts may already be deployed in Azure.
- Several stakeholders will be attending or presenting at the PSEC [6] conference.

[1] Performance Isolation Exposure in Virtualized Platforms with PCI Passthrough I/O Sharing, https://mediatum.ub.tum.de/doc/1187609/972322.pdf

[2] Securing Self-Virtualizing Ethernet Devices, https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-smolyar.pdf
[3] Denial-of-Service Attacks on PCI Passthrough Devices, http://publications.andre-richter.com/richter2015denial.pdf
[4] Open Compute Open System Firmware, http://www.opencompute.org/wiki/Open_System_Firmware
[5] Open Compute Security, http://www.opencompute.org/wiki/Security
[6] Firmware attestation: https://www.platformsecuritysummit.com/prepare/#attestation
[0] Notes for upcoming PCI emulation call thread:
https://lists.xenproject.org/archives/html/xen-devel/2018-05/msg00091.html

Note: we have no stake-holders from the security team on the call, which makes this a difficult discussion.

Rich: Andrew, Roger mentioned some problems related to security support in a previous discussion <Lars: is there a link to it?>

Rich: Earlier in this meeting we mentioned blacklisting, but thought we were going to use whitelisting?

Alexey: we know nothing about vendor specific capabilities for some devices which we may to expose, so whitelisting is problematic

Roger: maybe add a list of extra capabilities.

Rich: roughly agrees. Maybe someone can write down what the plan is such that it can be reviewed?

Alexey: there are a series of patches in this area to expose capabilities after the Q35 patches (such as support for dynamic fields?).

Rich: once we can document precisely how this works we can revisit the security support question

Roger: part of the problem was that some devices expose a configuration space on a Base Address Register (e.g. for Windows drivers).

- Could whitelist some known devices
- Paul confirms that some devices did that - ACTION: Paul to write up a couple

AOB

- Continue on the mailing list
- If needed try and arrange a all with a more narrow topic