

Preserving state for fast hypervisor update

Thursday, 23 September 2021 09:00 (45 minutes)

It is currently possible to do fast hypervisor update by preserving virtual machine state in memory during reboot. This approach relies on using emulated PMEM, DAX, and local live migration technologies.

As of today, there are a number of limitations with this approach:

1. The interface to preserve VM memory is not very flexible. The size and location of PMEM must be determined prior to hypervisor boot and cannot be changed later. Setting PMEM size and location requires intimate knowledge of the memory layout of the physical machine and thus, the settings are not portable.
2. Upstream kernel cannot preserve state of the devices. While there was work done by Intel in this direction, the work has not been upstreamed or discussed on public mailing lists. It also has some major limitations: 1) Intel IOMMU specific 2) reboot through firmware is not supported, only can work with kexec reboot 3) device state is preserved in a different memory from VM.
3. There is no way to preserve states of virtual functions.

In this presentation, we will show a demo of fast hypervisor update. We will have a discussion about how the three stated problems can be resolved.

The goal is to be able to preserve virtual machine state and any devices that are attached to it through kexec reboot and if firmware supports through the firmware. Also, the approach should be expandable to work on any platforms with KVM and IOMMU support.

I agree to abide by the anti-harassment policy

I agree

Primary author: TATASHIN, Pasha

Presenter: TATASHIN, Pasha

Session Classification: Performance and Scalability MC

Track Classification: Performance and Scalability MC