The s390 DIAGNOSE call on KVM

KVM on s390 supports the DIAGNOSE call for making hypercalls, both for native hypercalls and for selected hypercalls found on other s390 hypervisors.

Note that bits are numbered as by the usual s390 convention (most significant bit on the left).

General remarks

DIAGNOSE calls by the guest cause a mandatory intercept. This implies all supported DIAGNOSE calls need to be handled by either KVM or its userspace.

All DIAGNOSE calls supported by KVM use the RS-a format:

```
-------------------------------------
|  '83'  | R1 | R3 | B2 |     D2     |
-------------------------------------
0        8    12   16   20           31
```

The second-operand address (obtained by the base/displacement calculation) is not used to address data. Instead, bits 48-63 of this address specify the function code, and bits 0-47 are ignored.

The supported DIAGNOSE function codes vary by the userspace used. For DIAGNOSE function codes not specific to KVM, please refer to the documentation for the s390 hypervisors defining them.

DIAGNOSE function code 'X'500' - KVM virtio functions

If the function code specifies 0x500, various virtio-related functions are performed.

General register 1 contains the virtio subfunction code. Supported virtio subfunctions depend on KVM's userspace. Generally, userspace provides either s390-virtio (subcodes 0-2) or virtio-ccw (subcode 3).

Upon completion of the DIAGNOSE instruction, general register 2 contains the function's return code, which is either a return code or a subcode specific value.

Subcode 0 - s390-virtio notification and early console printk

> Handled by userspace.

Subcode 1 - s390-virtio reset

> Handled by userspace.

Subcode 2 - s390-virtio set status

> Handled by userspace.

Subcode 3 - virtio-ccw notification

> Handled by either userspace or KVM (ioeventfd case).
>
> General register 2 contains a subchannel-identification word denoting the subchannel of the virtio-ccw proxy device to be notified.
>
> General register 3 contains the number of the virtqueue to be notified.
>
> General register 4 contains a 64bit identifier for KVM usage (the kvm_io_bus cookie). If general register 4 does not contain a valid identifier, it is ignored.
>
> After completion of the DIAGNOSE call, general register 2 may contain a 64bit identifier (in the kvm_io_bus cookie case), or a negative error value, if an internal error occurred.
>
> See also the virtio standard for a discussion of this hypercall.

DIAGNOSE function code 'X'501 - KVM breakpoint

If the function code specifies 0x501, breakpoint functions may be performed. This function code is handled by userspace.

This diagnose function code has no subfunctions and uses no parameters.

DIAGNOSE function code 'X'9C - Voluntary Time Slice Yield

General register 1 contains the target CPU address.

In a guest of a hypervisor like LPAR, KVM or z/VM using shared host CPUs, DIAGNOSE with function code 0x9c may improve system performance by yielding the host CPU on which the guest CPU is running to be assigned to another guest CPU, preferably

the logical CPU containing the specified target CPU.

DIAG 'X'9C forwarding

The guest may send a DIAGNOSE 0x9c in order to yield to a certain other vcpu. An example is a Linux guest that tries to yield to the vcpu that is currently holding a spinlock, but not running.

However, on the host the real cpu backing the vcpu may itself not be running. Forwarding the DIAGNOSE 0x9c initially sent by the guest to yield to the backing cpu will hopefully cause that cpu, and thus subsequently the guest's vcpu, to be scheduled.

diag9c_forwarding_hz
: KVM kernel parameter allowing to specify the maximum number of DIAGNOSE 0x9c forwarding per second in the purpose of avoiding a DIAGNOSE 0x9c forwarding storm. A value of 0 turns the forwarding off.