DAWR issues on POWER9

On POWER9 the Data Address Watchpoint Register (DAWR) can cause a checkstop if it points to cache inhibited (CI) memory. Currently Linux has no way to distinguish CI memory when configuring the DAWR, so (for now) the DAWR is disabled by this commit:

```
commit 9654153158d3e0684a1bdb76dbababdb7111d5a0
Author: Michael Neuling <mikey@neuling.org>
Date:   Tue Mar 27 15:37:24 2018 +1100
powerpc: Disable DAWR in the base POWER9 CPU features
```

Technical Details:

DAWR has 6 different ways of being set. 1) ptrace 2) h_set_mode(DAWR) 3) h_set_dabr() 4) kvmppc_set_one_reg() 5) xmon

For ptrace, we now advertise zero breakpoints on POWER9 via the PPC_PTRACE_GETHWDBGINFO call. This results in GDB falling back to software emulation of the watchpoint (which is slow).

h_set_mode(DAWR) and h_set_dabr() will now return an error to the guest on a POWER9 host. Current Linux guests ignore this error, so they will silently not get the DAWR.

kvmppc_set_one_reg() will store the value in the vcpu but won't actually set it on POWER9 hardware. This is done so we don't break migration from POWER8 to POWER9, at the cost of silently losing the DAWR on the migration.

For xmon, the 'bd' command will return an error on P9.

Consequences for users

For GDB watchpoints (ie 'watch' command) on POWER9 bare metal , GDB will accept the command. Unfortunately since there is no hardware support for the watchpoint, GDB will software emulate the watchpoint making it run very slowly.

The same will also be true for any guests started on a POWER9 host. The watchpoint will fail and GDB will fall back to software emulation.

If a guest is started on a POWER8 host, GDB will accept the watchpoint and configure the hardware to use the DAWR. This will run at full speed since it can use the hardware emulation. Unfortunately if this guest is migrated to a POWER9 host, the watchpoint will be lost on the POWER9. Loads and stores to the watchpoint locations will not be trapped in GDB. The watchpoint is remembered, so if the guest is migrated back to the POWER8 host, it will start working again.

Force enabling the DAWR

Kernels (since ~v5.2) have an option to force enable the DAWR via:

```
echo Y > /sys/kernel/debug/powerpc/dawr_enable_dangerous
```

This enables the DAWR even on POWER9.

This is a dangerous setting, USE AT YOUR OWN RISK.

Some users may not care about a bad user crashing their box (ie. single user/desktop systems) and really want the DAWR. This allows them to force enable DAWR.

This flag can also be used to disable DAWR access. Once this is cleared, all DAWR access should be cleared immediately and your machine once again safe from crashing.

Userspace may get confused by toggling this. If DAWR is force enabled/disabled between getting the number of breakpoints (via PTRACE_GETHWDBGINFO) and setting the breakpoint, userspace will get an inconsistent view of what's available. Similarly for guests.

For the DAWR to be enabled in a KVM guest, the DAWR needs to be force enabled in the host AND the guest. For this reason, this won't work on POWERVM as it doesn't allow the HCALL to work. Writes of 'Y' to the dawr_enable_dangerous file will fail if the hypervisor doesn't support writing the DAWR.

To double check the DAWR is working, run this kernel selftest:

tools/testing/selftests/powerpc/ptrace/ptrace-hwbreak.c

Any errors/failures/skips mean something is wrong.