Machine checks on i386/x86-64

Andi Kleen, SuSE Labs
ak@suse.de
What is a machine check?

- **Hardware error**
  - Internal errors, Memory, Cache, IO, Busses
  - But users have a hard time to recognize this

- **Hardware is (mostly) error correcting**
  - 64bit is worse: more DIMMs, more bit flips
  - Smaller/more transistors
  - NMI, Thermal

- Can be panic or a logged event

- **Talking only about x86-64 here, a bit i386**
  - IA64, S390, PPC64 handle machine checks differently
The OS is just the messenger

- Caused by hardware
  - But always blamed on the software of course

- Sometimes drivers/BIOS can misprogram hardware

- Challenge is to explain to customers that their hardware is broken
General classification

- **Uncorrected**
  - Console log or a jpg
  - Don’t make it to disk
  - Sometimes available in BIOS event log afterwards
  - Sometimes can be recovered by kernel after reboot
  - Run through mcelog --ascii first

- **Corrected error but logged**
  - Logged in a binary log (/dev/mcelog)
  - mcelog cronjob decodes them into /var/log/mcelog
  - On i386 they just go to /var/log/messages
  - Nothing really bad happened, but if it happens often hardware will likely fail
A live example

2.4 AMD

kernel: Northbridge Machine Check exception = b60ea00100000813 0
ecc error
    link number 0
err cpu1
    uncorrected ecc error
processor context corrupt
error address valid
error enable
error uncorrected
previous error lost
    error address 00000001826ac018
Address: 00000001826ac018
CPU 1: Machine Check Exception: 0000000000000000
Kernel panic: Unable to continue
Another example

2.6 AMD fatal

HARDWARE ERROR
CPU 0: Machine Check Exception: 4 Bank 4: b605200100000813
TSC 24291bbb8104 ADDR fbf2c068
This is not a software problem!
Run through mcelog --ascii to decode and contact your hardware vendor
Kernel panic - not syncing: Machine check
Intel example

2.6 Fatal

CPU 1: Machine Check Exception: 0000000000000004
Bank 1: b200000000000115
Kernel panic: CPU context corrupt
Another Intel example

Non fatal mcelog entry

STATUS 9000020110800e0f MCGSTATUS 2
MCE 0
HARDWARE ERROR. This is *NOT* a software problem!
Please contact your hardware vendor
CPU 21 BANK 4 TSC 20e7a6ec72de3
RIP 00:ffffffff801098e7
MCG status:EIPV
MCi status:
Error enabled
MCA:BUS Generic Generic Generic Other-transaction Request-timeout Error
Model:Pad address glitch
Oops after machine check

(shouldn't happen)
User interface I

> find /sys | grep machine
/sys/devices/system/machinecheck
/sys/devices/system/machinecheck/machinecheck7
/sys/devices/system/machinecheck/machinecheck7/check_interval
/sys/devices/system/machinecheck/machinecheck7/tolerant
/sys/devices/system/machinecheck/machinecheck7/bank4ctl
/sys/devices/system/machinecheck/machinecheck7/bank3ctl
/sys/devices/system/machinecheck/machinecheck7/bank2ctl
/sys/devices/system/machinecheck/machinecheck7/bank1ctl
/sys/devices/system/machinecheck/machinecheck7/bank0ctl
/sys/devices/system/machinecheck/machinecheck6
/sys/devices/system/machinecheck/machinecheck6/check_interval
...
User Interface II

- Cronjob with mcelog
  - Decodes /dev/mcelog into /var/log/mcelog

- tolerance level
  - 0 always panic
  - 1 panic if deadlock
  - 2 try to avoid deadlock

- oops=panic, panic=timeout, mce=off
AMD Opteron specific MCEs

- When panic run through mcelog --ascii first

- Banks
  - 0-3 CPU internal (DC, IC, BU, LS)
  - 4 Northbridge: only really interesting one

- Banks 0-3
  - Ask customer to check cooling or their CPU might be just dying
  - Or power supply/VRM broken
0 / CorrEccEn single bit flip in memory
  ◦ If they get a lot DIMM might be dying or cooling/PS/VRM bad

1 / UnCorrEccEn
  ◦ Can be triple fault or panic
  ◦ memtest86 and exchange DIMMs/VRMs

12 Watchdog
  ◦ Could be plugin card/BIOS/driver
  ◦ Or another CPU is dead or mainboard flakey.
More AMD notes

- Bank 4 GART table walk is harmless
  - SLES10 filters it out

- On some old systems bootup leaves bogus events in the log
  - Newer kernels filter that out
  - Can be overwritten with mce=bootlog to catch old machine checks from before reboot

- RevF: DRAM thresh pseudo events
  - Only if configured using sysfs
Intel

- Banks depends on the family
  - P4/Xeon
  - P3/P-M
  - Conroe/Woodcrest will have new ones

- Thermal events cause pseudo entry in mcelog
  - On i386 it’s a log entry in the kernel log.
  - CPU is overheating.
Intel II

- Only BUS errors are interesting normally
  - Non bus is CPU internal

- Don’t report addresses normally

- We don’t get a lot of reports in general
  - Good or bad sign?

- For more details see Appendix E in IA32 manual vol 3
Tainted bits in Ooopses

* 'P' - Proprietary module has been loaded.
* 'F' - Module has been forcibly loaded.
* 'S' - SMP with CPUs not designed for SMP.
* 'R' - User forced a module unload.
* 'M' - Machine had a machine check experience.
* 'B' - System has hit bad_page.
* 'U' - Unsupported module loaded.
Differences on i386

- Logs to normal /var/log/messages
- Doesn’t decode by default
- Doesn’t have tolerance levels
- Generally less reliable
Differences on 2.4/x86-64

- mcelog doesn’t exist

- Decoded logs go into normal kernel log
  - /var/log/messages

- Doesn’t filter out some bogus one
Other tools

- `mcelog --cputype --ascii`
  - `cputype: --k8 or --p4`
  - `Just paste it in`

- `parsemce for i386`
Tips and tricks

When you see an oops after the machine check

- Sometimes it’s a real bug, sometimes it is unavoidable
- Put it in bugzilla (but it might be not fixable)
- But explain the customer that fixing the oops won’t fix the original problem
Tips and tricks II

- **mce=off**
  - Or disable the event in sysfs
  - Only recommended in emergencies
  - Cause silent data corruption
  - Hardware needs fixing.

- **Corrected with PCC**
  - Disable the subevent in sysfs
  - Will be still logged, but no panics
  - BIOS/kernel do that for known offenders
  - If there is a pattern report please
Tips and tricks III

- Low tolerance level / panic=XXX for clusters
- mce=bootlog

- NMI can have multiple causes
  - User presses button
  - Can be caused by some PCI errors
mcelog --dmi

- Run on the same machine and with same configuration
  - As root
  - You need an address (ADDR xxxx)

- Gives label(s) on the mainboard

- Problems
  - Unreliable due to BIOS bugs
  - Often gives multiple DIMMs because of interleaving or bad BIOS
# echo "ADDR 0x123456" | mcelog --dmi --ascii
WARNING: with --dmi mcelog --ascii must run on the same machine with the same BIOS/memory configuration as where the machine check occurred.
Resolving address 123456 using SMBIOS
WARNING: SMBIOS data is often unreliable. Take with a grain of salt!
DRAM DIMM Synchronous Width 128 Data Width 64 Size 1 GB
Device Locator: H0_DIMM3
Bank Locator: Bank 4
More information

- **Vendor documentation**
  - AMD BIOS and Kernel Developers Guide
  - IA32 Intel Architecture SW Developers manual: Volume 3

- parsemce for i386

- **LinuxKongress paper**
  - ~ak/pub/mce.pdf