NVMe Fault Injection

Linux's fault injection framework provides a systematic way to support error injection via debugfs in the /sys/kernel/debug directory. When enabled, the default NVME_SC_INVALID_OPCODE with no retry will be injected into the nvme_try_complete_req. Users can change the default status code and no retry flag via the debugfs. The list of Generic Command Status can be found in include/linux/nvme.h

Following examples show how to inject an error into the nvme.

First, enable CONFIG_FAULT_INJECTION_DEBUG_FS kernel config, recompile the kernel. After booting up the kernel, do the following.

Example 1: Inject default status code with no retry

```
mount /dev/nvme0n1 /mnt
echo 1 > /sys/kernel/debug/nvme0n1/fault_inject/times
echo 100 > /sys/kernel/debug/nvme0n1/fault_inject/probability
cp a.file /mnt
```

Expected Result:

```
cp: cannot stat â€˜/mnt/a.fileâ€™: Input/output error
```

Message from dmesg:

```
FAULT_INJECTION: forcing a failure.
name fault_inject, interval 1, probability 100, space 0, times 1
CPU: 0 PID: 0 Comm: swapper/0 Not tainted 4.15.0-rc8+ #2
Hardware name: innotek GmbH VirtualBox/VirtualBox,
BIOS VirtualBox 12/01/2006
Call Trace:
  <IRQ>
  dump_stack+0x5c/0x7d
  should_fail+0x148/0x170
  nvme_should_fail+0x2f/0x50 [nvme_core]
  nvme_process_cq+0xe7/0x1d0 [nvme]
  nvme_irq+0x1e/0x40 [nvme]
  __handle_irq_event_percpu+0x3a/0x190
  handle_irq_event_percpu+0x30/0x70
  handle_irq_event+0x36/0x60
  handle_fasteoi_irq+0x78/0x120
  handle_irq+0xa7/0x130
  ? tick_irq_enter+0xa8/0xc0
  do_IRQ+0x43/0xc0
  common_interrupt+0xa2/0xa2
  </IRQ>
RIP: 0010:native_safe_halt+0x2/0x10
RSP: 0018:ffffffff82003e90 EFLAGS: 00000246 ORIG_RAX: ffffffffffffffdd
RAX: ffffffff817a10c0 RBX: ffffffff82012480 RCX: 0000000000000000
RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000
RBP: 0000000000000000 R08: 000000008e38ce64 R09: 0000000000000000
R10: 0000000000000000 R11: 0000000000000000 R12: ffffffff82012480
R13: ffffffff82012480 R14: 0000000000000000 R15: 0000000000000000
  ? __sched_text_end+0x4/0x4
  default_idle+0x18/0xf0
  do_idle+0x150/0x1d0
  cpu_startup_entry+0x6f/0x80
  start_kernel+0x4c4/0x4e4
  ? set_init_arg+0x55/0x55
  secondary_startup_64+0xa5/0xb0
  print_req_error: I/O error, dev nvme0n1, sector 9240
EXT4-fs error (device nvme0n1): ext4_find_entry:1436:
inode #2: comm cp: reading directory lblock 0
```

Example 2: Inject default status code with retry

```
mount /dev/nvme0n1 /mnt
echo 1 > /sys/kernel/debug/nvme0n1/fault_inject/times
echo 100 > /sys/kernel/debug/nvme0n1/fault_inject/probability
echo 1 > /sys/kernel/debug/nvme0n1/fault_inject/status
echo 0 > /sys/kernel/debug/nvme0n1/fault_inject/dont_retry

cp a.file /mnt
```

Expected Result:

```
command success without error
```

Message from dmesg:

```
FAULT_INJECTION: forcing a failure.
name fault_inject, interval 1, probability 100, space 0, times 1
CPU: 1 PID: 0 Comm: swapper/1 Not tainted 4.15.0-rc8+ #4
Hardware name: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
Call Trace:
  <IRQ>
  dump_stack+0x5c/0x7d
  should_fail+0x148/0x170
  nvme_should_fail+0x30/0x60 [nvme_core]
  nvme_loop_queue_response+0x84/0x110 [nvme_loop]
  nvmet_req_complete+0x11/0x40 [nvmet]
  nvmet_bio_done+0x28/0x40 [nvmet]
  blk_update_request+0xb0/0x310
  blk_mq_end_request+0x18/0x60
  flush_smp_call_function_queue+0x3d/0xf0
  smp_call_function_single_interrupt+0x2c/0xc0
  call_function_single_interrupt+0xa2/0xb0
  </IRQ>
RIP: 0010:native_safe_halt+0x2/0x10
RSP: 0018:ffffc9000068bec0 EFLAGS: 00000246 ORIG_RAX: ffffffffffffff04
RAX: ffffffff817a10c0 RBX: ffff88011a3c9680 RCX: 0000000000000000
RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000
RBP: 0000000000000001 R08: 000000008e38c131 R09: 0000000000000000
R10: 0000000000000000 R11: 0000000000000000 R12: ffff88011a3c9680
R13: ffff88011a3c9680 R14: 0000000000000000 R15: 0000000000000000
  ? __sched_text_end+0x4/0x4
  default_idle+0x18/0xf0
  do_idle+0x150/0x1d0
  cpu_startup_entry+0x6f/0x80
  start_secondary+0x187/0x1e0
  secondary_startup_64+0xa5/0xb0
```

Example 3: Inject an error into the 10th admin command

```
echo 100 > /sys/kernel/debug/nvme0/fault_inject/probability
echo 10 > /sys/kernel/debug/nvme0/fault_inject/space
echo 1 > /sys/kernel/debug/nvme0/fault_inject/times
nvme reset /dev/nvme0
```

Expected Result:

```
After NVMe controller reset, the reinitialization may or may not succeed.
It depends on which admin command is actually forced to fail.
```

Message from dmesg:

```
nvme nvme0: resetting controller
FAULT_INJECTION: forcing a failure.
name fault_inject, interval 1, probability 100, space 1, times 1
CPU: 0 PID: 0 Comm: swapper/0 Not tainted 5.2.0-rc2+ #2
Hardware name: MSI MS-7A45/B150M MORTAR ARCTIC (MS-7A45), BIOS 1.50 04/25/2017
Call Trace:
 <IRQ>
 dump_stack+0x63/0x85
 should_fail+0x14a/0x170
 nvme_should_fail+0x38/0x80 [nvme_core]
 nvme_irq+0x129/0x280 [nvme]
 ? blk_mq_end_request+0xb3/0x120
 __handle_irq_event_percpu+0x84/0x1a0
 handle_irq_event_percpu+0x32/0x80
 handle_irq_event+0x3b/0x60
 handle_edge_irq+0x7f/0x1a0
 handle_irq+0x20/0x30
 do_IRQ+0x4e/0xe0
 common_interrupt+0xf/0xf
 </IRQ>
RIP: 0010:cpuidle_enter_state+0xc5/0x460
Code: ff e8 8f 5f 86 ff 80 7d c7 00 74 17 9c 58 0f 1f 44 00 00 f6 c4 02 0f 85 69 03 00 00 31 ff e8 62 aa 8c ff fb 66 0f 1f 44 00 00 <45>
RSP: 0018:ffffffff88c03dd0 EFLAGS: 00000246 ORIG_RAX: ffffffffffffffdc
RAX: ffff9dac25a2ac80 RBX: ffffffff88d53760 RCX: 000000000000001f
RDX: 0000000000000000 RSI: 000000002d958403 RDI: 0000000000000000
RBP: ffffffff88c03e18 R08: fffffff75e35ffb7 R09: 00000a49a56c0b48
R10: ffffffff88c03da0 R11: 0000000000001b0c R12: ffff9dac25a34d00
R13: 0000000000000006 R14: 0000000000000006 R15: ffffffff88d53760
 cpuidle_enter+0x2e/0x40
 call_cpuidle+0x23/0x40
 do_idle+0x201/0x280
 cpu_startup_entry+0x1d/0x20
 rest_init+0xaa/0xb0
 arch_call_rest_init+0xe/0x1b
 start_kernel+0x51c/0x53b
 x86_64_start_reservations+0x24/0x26
 x86_64_start_kernel+0x74/0x77
 secondary_startup_64+0xa4/0xb0
nvme nvme0: Could not set queue count (16385)
nvme nvme0: IO queues not created
```