Kernel lab report

2013-11415 이산하, 2013-11406 송원욱

1. Process tree

We implemented a kernel module (chardev) that prints all of the parent processes to the root from the process calling the device. The communication has been done using the ioctl() function inside a process, and we have used our own structure containing the process name and the process id. Inside the module, it searches the processes using task_struct and task_struct→parent, and puts the information in our structure, and we simply read the information back from our structure.

Result for 1.

```
Opening device file: 3
get_process_tree:
\process name: swapper/0, process id: 0
        \process name: init, process id: 1
                \process name: mdm, process id: 1480
                        \process name: mdm, process id: 1560
                                \process name: cinnamon-sessio, process id: 1904
                                        \process name: cinnamon-launch, process id: 2606
                                                \process name: cinnamon, process id: 2622
                                                        \process name: gnome-terminal, process id: 2904
                                                                \process name: bash, process id: 2911
                                                                        \process name: make, process id: 4677
                                                                                \process name: sudo, process id: 4678
                                                                                        \process name: tree, process id: 4679
```

2. Program Management Unit

We also implemented a kernel module starting, stopping, resetting, selecting, and reading the PMU counter and also reading the TSC register. For the job in between resetting, starting, and stopping the counter, we thought about doing a simple bubble sort for an array of size 8. You'll have to enter one of the provided options to allocate some events to general-purpose counter. The fixed-purpose counter and TSC's value will be always given. Here are the options:

(1) TLB misses when load & store

You can see the misses in TLB that cause a page walk or page walk completed during load and store.

* Our bubble sort is a very short operation, so these values could be 0. If such action occurs, please run it several times.

```
Stopping PMU counter...Done! Reading PMU counter...Done!
Printing general purpose function couter
Miss TLB load walk:          0
Miss TLB load compl:         0
Miss TLB store walk:         0
Miss TLB store compl:        0
Printing fixed purpose function counter
```

(2) Retired load micro-operations in cache

You can see the retired load uops with L1 cache hit | miss, L2 cache hit | miss.

L1 cache is closer to CPU then L2 cache, so it made more hits and misses than L2 cache.

Result for 2. (1)

```
Please select PMU event to monitor!
Type your event: 2
Stopping PMU counter...Done! Reseting PMU counter...Done! Reading PMU counter...Done!
Printing general purpose function couter
Retired uops L1 hit:        0
Retired uops L1 miss:       0
Retired uops L2 hit:        0
Retired uops L2 miss:       0
Printing fixed purpose function counter
instr retired:              0
core cycles:                0
ref cycles:                 0
TSC: 5077260204027
Starting PMU counter...Done!
array[8]: [7, 9, 3, 1, 7, 9, 3, 1]
array[8]: [7, 3, 9, 1, 7, 9, 3, 1]
array[8]: [7, 3, 1, 9, 7, 9, 3, 1]
array[8]: [7, 3, 1, 7, 9, 9, 3, 1]
array[8]: [7, 3, 1, 7, 9, 3, 9, 1]
array[8]: [7, 3, 1, 7, 9, 3, 1, 9]
array[8]: [3, 7, 1, 7, 9, 3, 1, 9]
array[8]: [3, 1, 7, 7, 9, 3, 1, 9]
array[8]: [3, 1, 7, 7, 3, 9, 1, 9]
array[8]: [3, 1, 7, 7, 3, 1, 9, 9]
array[8]: [1, 3, 7, 7, 3, 1, 9, 9]
array[8]: [1, 3, 7, 3, 7, 1, 9, 9]
array[8]: [1, 3, 7, 3, 1, 7, 9, 9]
array[8]: [1, 3, 3, 7, 1, 7, 9, 9]
array[8]: [1, 3, 3, 1, 7, 7, 9, 9]
array[8]: [1, 3, 1, 3, 7, 7, 9, 9]
array[8]: [1, 1, 3, 3, 7, 7, 9, 9]
Stopping PMU counter...Done! Reading PMU counter...Done!
Printing general purpose function couter
Retired uops L1 hit:    25433
Retired uops L1 miss:     135
Retired uops L2 hit:      102
Retired uops L2 miss:      33
Printing fixed purpose function counter
instr retired:          98013
core cycles:            47252
ref cycles:            106326
TSC: 5077261034613
```

Result for 2. (2)

```
Please select PMU event to monitor!
Type your event: 2
Stopping PMU counter...Done! Reseting PMU counter...Done! Reading PMU counter...Done!
Printing general purpose function couter
Retired uops L1 hit:        0
Retired uops L1 miss:       0
Retired uops L2 hit:        0
Retired uops L2 miss:       0
Printing fixed purpose function counter
instr retired:              0
core cycles:                0
ref cycles:                 0
TSC: 5077260204027
Starting PMU counter...Done!
array[8]: [7, 9, 3, 1, 7, 9, 3, 1]
array[8]: [7, 3, 9, 1, 7, 9, 3, 1]
array[8]: [7, 3, 1, 9, 7, 9, 3, 1]
array[8]: [7, 3, 1, 7, 9, 9, 3, 1]
array[8]: [7, 3, 1, 7, 9, 3, 9, 1]
array[8]: [7, 3, 1, 7, 9, 3, 1, 9]
array[8]: [3, 7, 1, 7, 9, 3, 1, 9]
array[8]: [3, 1, 7, 7, 9, 3, 1, 9]
array[8]: [3, 1, 7, 7, 3, 9, 1, 9]
array[8]: [3, 1, 7, 7, 3, 1, 9, 9]
array[8]: [1, 3, 7, 7, 3, 1, 9, 9]
array[8]: [1, 3, 7, 3, 7, 1, 9, 9]
array[8]: [1, 3, 7, 3, 1, 7, 9, 9]
array[8]: [1, 3, 3, 7, 1, 7, 9, 9]
array[8]: [1, 3, 3, 1, 7, 7, 9, 9]
array[8]: [1, 3, 1, 3, 7, 7, 9, 9]
array[8]: [1, 1, 3, 3, 7, 7, 9, 9]
Stopping PMU counter...Done! Reading PMU counter...Done!
Printing general purpose function couter
Retired uops L1 hit:    25433
Retired uops L1 miss:     135
Retired uops L2 hit:      102
Retired uops L2 miss:      33
Printing fixed purpose function counter
instr retired:          98013
core cycles:            47252
ref cycles:            106326
TSC: 5077261034613
```