**MENG PROJECT, FA20** **Linux Kernel Modifications**
Name: Jonathan Gao NetID: jg992

## Progress this week

Not as much time this week as I'm finishing up my final projects and such. I did take a look into writing a syscall, but it seems that is much easier than I thought. There are macros set for defining syscalls within the table, in which I can then call it from user space using the `syscall()` function as seen in this tutorial [1].

There seems to be a lot of layers above the niceness system of the CFS scheduler. The grouping system of the scheduler "augments" the functionality of niceness slightly? I'm still a little unsure of this, but the autogrouping feature for CFS uses niceness values from -20 to +19 [2]. However, the cgroup utility it is based off of uses relative timeslice numbers in nanoseconds [3, 4, 5]. I think this means that the autogrouping feature automatically adjusts the values to achieve the desired niceness values. Cgroup has had two iterations, both of them using a mounted filesystem to manage process runtime. This might be useful as it gives information on the process runtime, process ids, etc. that can help with data for a machine learning network.

## Plan for coming week

I think the cgroup v2 functionality can prove useful. It's a big mess, but I think it can be worked out so that I can adjust the niceness values of groups. I think this is the way to go, as most programs now rely on groups of processes/threads due to increasing reliance on multiprocessor architectures. I'll take some time looking into how to work this functionality, and then on how to write a syscall that implements this functionality.

There are some concerns to this however,

1. Mounting on the file system, how to interact with these files through C
2. From my findings so far, it seems like every process has its own group. This might be from autogrouping, so it might be worthwhile to just use the autogrouping functionality instead. It is pretty similar, but doesn't offer any runtime information.

## Problems

Perhaps some, as mentioned above.

## Long term plan: still OK?

A little behind, but still within reason.

## References

[1] S. Brennan, “Tutorial - Write a System Call - Stephen Brennan.” https://brennan.io/2016/11/14/kernel-dev-ep3/.

[2] “Linux-Kernel Archive: RFC: Documentation of the autogroup feature [v2].” http://lkml.iu.edu/hypermail/linux/kernel/1611.3/00766.html.

[3] “Cgroups(7) - Linux manual page.” https://man7.org/linux/man-pages/man7/cgroups.7.html.

[4] “Control Groups — The Linux Kernel documentation.” https://www.kernel.org/doc/html/latest/admin-guide/cgroup-v1/cgroups.html.

[5] T. Heo, “Control Group v2.” https://www.kernel.org/doc/Documentation/cgroup-v2.txt, Oct. 2015.