Case Study: Resolving frozen processes in kubernetes

See an up to date, HTML version of this case study on andrewhowden.com

Problem

In previous work it was determined that there were outstanding issues with the kernel implementation in NFSv4 and a change was made to shift from the NFSv4 protocol to NFSv3. At the same a few PHP workloads started becoming "stuck" during their application bootstrap.

Colleagues were able to refine the issue down to a single workload that would reliably break during the first requests to a newly created version of that workload.

Investigation

The processes were easily identifiable from the host `pid` namespace[1] as they were a series of php-fpm worker processes under a single fpm manager process, all of which were in D[2] state. Given this, it was easiest to use `nsenter --all --target ${PID}`[3].

From there, the goal is to determine exactly what the process was doing when it got stuck. To that end I executed `sudo strace -f -s 8096 -p ${PID}`[4] to listen to what syscalls that process was making to the kernel, hopefully including the last syscall with arguments that was executed. Unfortunately this did not work; rather, `strace` got stuck and could not be terminated with `SIGINT` (`ctrl + c`) nor `SIGKILL`. That reduced the problem down to something in "kernel land".

Resolution

Evaluation

References

bibilography::[]