Module 2
Basic Linux Analysis and Observability Tools

Pseudo file system

- Also known as a virtual filesystem.
- It provides an interface for accessing kernel data structure and system information.
- Some common example of pseudo file systems in Linux includes:
  - /proc
  - /sys
  - /dev
  - /tmp
  - debugfs

/proc Filesystem

- It provides an virtual interface to access process and system related information.
- Each process is represented as a directory under /proc, with a unique process ID (PID).
- Within each process directory, there are various files containing information about the process, such as command-line arguments, environment variables, and status.
- /proc exposes system-wide information, including CPU and memory usage, loaded modules, interrupts, and file system statistics.
- /proc is used by system utilities, diagnostic tools, and monitoring applications to gather information about the system's current state.

/proc Filesystem

Important files and directories in /proc

- `/proc/cpuinfo` : CPU information
- `/proc/meminfo` : Memory information
- `/proc/loadavg` : Average system load
- `/proc/version` : Linux kernel version.
- `/proc/filesystems` : Filesystems supported by the kernel.
- `/proc/cmdline` : Kernel Command-line arguments.
- `/proc/<PID>/` : Process information.
  - `/proc/<pid>/status` process information
  - `/proc/<pid>/maps` process memory mappings
- Please refer to man proc(5) for the list of files and description.

/sys Filesystem

- /sys filesystem provides a view of the system's hardware, devices, drivers, and kernel modules.
- It is organized hierarchically, with each device and driver represented as a directory.
- It exposes information about devices, such as their attributes, status, and configuration parameters.
- Refer to man sysfs(5) for the list of files and description.

/dev Filesystem

- The dev filesystem also known as the device file system, is a pseudo file system in Linux that provides a way to access devices as if they were regular files.
- It acts as an interface between user-space applications and kernel device drivers.
- It allows processes to interact with devices using standard file operations such as reading, writing, and seeking.
- The /dev file system is typically managed by a combination of the udev daemon and the devtmpfs file system.
- Please refer to man udev(7) for information on udev.

Debug Filesystem

- DebugFS provides a mechanism for kernel developers to expose debugging and tracing information to user-space.
- Typically mounted on the /sys/kernel/debug mount point.
  - Pre-requisite: CONFIG_DEBUG_FS=y
  - `mount -t debugfs none /sys/kernel/debug`
- Commonly used by perf, ftrace (tracefs), dynamic debugging, kernel debuggers
  - Dynamic debug: `/sys/kernel/debug/dynamic_debug`
  - Ftrace: `/sys/kernel/tracing`

Linux Monitoring Tools

Linux Provides several monitoring tools available that can help monitor system performance, resource utilization, network activity, and various other aspects.

Commonly used Linux monitoring tools:

- Process Monitoring: ps, top, htop, pstree.
- Memory Monitoring: free, vmstat, pmap.
- Disk i/o Monitoring: iostat, iotop
- Scheduler: mpstat
- Networking: netstat, tcpdump, ethtool

Linux Performance Observability Tools

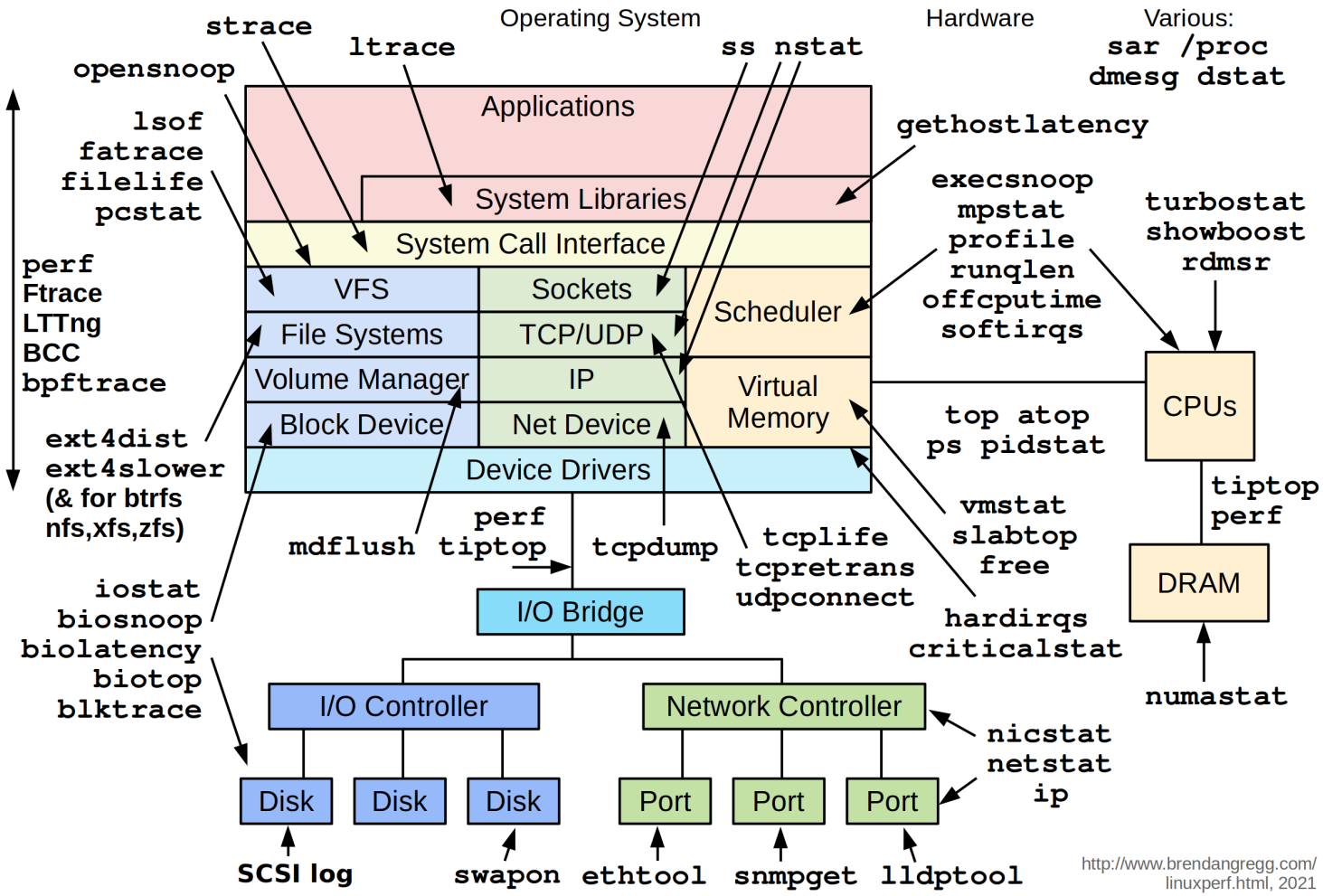

Image credits: https://www.brendangregg.com/linuxperf.html

Memory representation

- VSS (Virtual Set Size): Total virtual memory usage of a process, including shared and private memory.
- RSS (Resident Set Size): Total physical memory held in physical RAM including shared library.
- USS (Unique Set Size): Physical memory held in physical RAM excluding shared library.
- PSS (Proportional Set Size): Estimate of physical memory of process including proportionate shared memory. PSS divides the shared memory equally among the processes sharing it.
- VSS >= RSS >= PSS >= USS

Process Tools

Process status (ps) command

- `ps` is a command-line utility used to display active processes and their attributes (man ps(1)).
- It is one of the most commonly used commands and is essential for process management and troubleshooting.
- ps displays process IDs (PIDs), parent process IDs (PPIDs), CPU, memory usage, process status etc.

```
manas@manas-sandbox:~$ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.1 225508  9252 ?        Ss   08:02   0:01 /sbin/init splash
root           2  0.0  0.0      0     0 ?        S    08:02   0:00 [kthreadd]
root           8  0.0  0.0      0     0 ?        I<   08:02   0:00 [mm_percpu_wq]
root           9  0.0  0.0      0     0 ?        S    08:02   0:00 [ksoftirqd/0]
root         937  0.0  0.1 360600  9568 ?        Ssl  08:02   0:00 /usr/sbin/ModemManager --filter-policy=strict
manas       2213  0.0  0.0 281244  7704 ?        Sl   08:52   0:00 /usr/bin/gnome-keyring-daemon --daemonize --login
manas       2246  0.0  0.1 551960 14172 tty2     Sl+  08:52   0:00 /usr/lib/gnome-session/gnome-session-binary --session=ubuntu
manas       2895  1.7  2.3 1277124 185528 tty2   SLl+ 08:53   0:02 /usr/bin/gnome-software --gapplication-service
root        2923  0.2  0.9 681036 73732 ?        Ssl  08:53   0:00 /usr/lib/fwupd/fwupd
manas       2989  2.3  2.1 998600 175284 tty2    SNl+ 08:53   0:04 /usr/bin/python3 /usr/bin/update-manager --no-update --no-focus-on-map
manas       3078  0.0  0.4 797220 34280 tty2     Sl+  08:54   0:00 /usr/lib/deja-dup/deja-dup-monitor
```

Table of processes (Top) command

- `top` is a command-line utility that provides real-time monitoring of system processes, CPU usage, and memory usage. (man top(1))

```
manas@manas-sandbox:~$ top -d1
top - 09:26:33 up  1:24,  2 users,  load average: 0.07, 0.06, 0.02
Tasks: 324 total,   1 running, 259 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.1 us,  0.1 sy,  0.0 ni, 99.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  8063456 total,  4599568 free,  1324504 used,  2139384 buff/cache
KiB Swap:  2097148 total,  2097148 free,        0 used.  6362964 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 3252 manas     20   0   44372   4108   3324 R   6.2  0.1   0:00.01 top
    1 root      20   0  225508   9252   6596 S   0.0  0.1   0:01.73 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.00 kthreadd
...
   40 root     -51   0       0      0      0 S   0.0  0.0   0:00.00 idle_inject/5
   41 root      rt   0       0      0      0 S   0.0  0.0   0:00.09 migration/5
```

Process tree (pstree)

- `pstree` is a command-line utility that displays a tree-like representation of running processes, showing their parent-child relationships (man pstree(1)).
- The tree is rooted at PID (if mentioned) or it is rooted at `init` if PID is omitted.

```
manas@manas-sandbox:~$ pstree -p 3072
sshd(3072)───bash(3077)───sudo(3273)───sudo(3290)───su(3291)───bash(3292)───pstree(3460)
```

Memory Tools

free

- `free` is a command-line utility that provides information about system memory usage, including total, used, and free memory (man free(1)).
  - Uses `/proc/meminfo` file to get the memory information.

```
manas@manas-sandbox:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:            15Gi       684Mi        13Gi       614Mi       1.5Gi        13Gi
Swap:             0B          0B          0B
```

Virtual Memory Stats (vmstat)

- `vmstat` is a command-line utility used to display virtual memory statistics including information about system memory, processes, paging, block I/O, CPU usage, and more (man vmstat(8)).

```
manas@manas-sandbox:~$ vmstat 1 6
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0   2328 5286684 269072 1901760    0    0     0     5    4    5  1  0 99  0  0
 0  0   2328 5286740 269072 1901760    0    0     0     0   11   79  0  0 100  0  0
 0  0   2328 5286740 269072 1901760    0    0     0     0   13   91  0  0 100  0  0
 0  0   2328 5286740 269072 1901760    0    0     0     0    9   71  0  0 100  0  0
 0  0   2328 5286740 269072 1901760    0    0     0     0    9   71  0  0 100  0  0
 0  0   2328 5286740 269072 1901760    0    0     0     0   14   95  0  0 100  0  0
```

Process map (pmap)

- `pmap` is a command-line utility that provides detailed information about the memory mappings of a process (man pmap(1)).
- Uses `/proc/{PID}/maps` file.

```
ubuntu@sandbox:~/work/Examples/memory$ pmap -x `pidof memory`
8219:   ./memory
Address           Kbytes     RSS   Dirty Mode  Mapping
000055b7d159c000       4       4       0 r---- memory [Read Only, Private Segment (Contains Constants etc)]
000055b7d159d000       4       4       0 r-x-- memory [Executable Code segment]
000055b7d15a0000       4       4       4 rw--- memory
000055b7d1d6a000     132       4       4 rw---   [ anon ] [Heap section of process]
00007f5e59400000     160     160       0 r---- libc.so.6 [libc (Shared Lib) read only section]
00007f5e59428000    1620    1036       0 r-x-- libc.so.6 [Libc executable section]
00007f5e59619000       8       8       8 rw--- libc.so.6
00007f5e5961b000      52      20      20 rw---   [ anon ]
00007f5e596ba000       8       8       0 r---- ld-linux-x86-64.so.2
00007f5e596bc000     168     168       0 r-x-- ld-linux-x86-64.so.2
00007f5e596f4000       8       8       8 rw--- ld-linux-x86-64.so.2
00007ffe1782b000     132      16      16 rw---   [ stack ] [Stack section of the process]
00007ffe179a5000       8       4       0 r-x--   [ anon ]
ffffffffff600000       4       0       0 --x--   [ anon ]
---------------- ------- ------- -------
total kB            2776    1592     100
            2776K
```

CPU and I/O Related

I/O Statistics (iostat)

- Monitor and report I/O statistics of disk, disk controller, and filesystem performance (man iostat(1)).
- Useful to understand system wide I/O load using metrics like disk utilization, I/O rates, throughput, and response times.

```
manas@sandbox:~$ iostat
Linux 5.19.0-42-generic (manas-sandbox)    20/06/23        _x86_64_        (8 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.16    0.03    0.22    0.04    0.00   99.55

Device             tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
loop0             0.01         0.01         0.00         0.00         17          0          0
loop1             0.02         0.13         0.00         0.00        346          0          0
loop2             0.02         0.13         0.00         0.00        364          0          0
loop3             0.02         0.39         0.00         0.00       1095          0          0
nvme0n1          10.56       336.19       152.87         0.00     926703     421397          0
```

iotop

- `iotop` is used to monitor real time I/O statistics on a per-process basis (man iotop(8)).
- Helps identify processes generating high I/O load and causing performance issues.
- Metrics includes total I/O, read and write rates, and I/O priorities.

```
Total DISK READ:         0.00 B/s | Total DISK WRITE:         0.00 B/s
Current DISK READ:       0.00 B/s | Current DISK WRITE:       0.00 B/s
    TID  PRIO  USER     DISK READ  DISK WRITE  SWAPIN     IO>    COMMAND
      1 be/4 root        0.00 B/s    0.00 B/s  ?unavailable?  init splash
      2 be/4 root        0.00 B/s    0.00 B/s  ?unavailable?  [kthreadd]
      3 be/0 root        0.00 B/s    0.00 B/s  ?unavailable?  [rcu_gp]
      4 be/0 root        0.00 B/s    0.00 B/s  ?unavailable?  [rcu_par_gp]
      5 be/0 root        0.00 B/s    0.00 B/s  ?unavailable?  [slub_flushwq]
      ...
```

Multi Processor Statistic (mpstat)

- Helps in monitoring individual CPU core usage using metrics like user, system, and idle time, as well as other statistics like interrupts and context switches (man mpstat(1)).
- Useful in identifying CPU bottlenecks, load imbalances, and overall CPU performance.

```
manas@sandbox:~$ mpstat -P ALL
Linux 5.19.0-42-generic (manas-sandbox)    20/06/23        _x86_64_        (8 CPU)

08:14:27     CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
08:14:27     all    0.23    0.01    0.23    0.03    0.00    0.00    0.00    0.00    0.00   99.50
08:14:27       0    0.31    0.01    0.41    0.02    0.00    0.01    0.00    0.00    0.00   99.24
08:14:27       1    0.36    0.01    0.21    0.02    0.00    0.01    0.00    0.00    0.00   99.39
08:14:27       2    0.20    0.04    0.27    0.04    0.00    0.01    0.00    0.00    0.00   99.44
08:14:27       3    0.21    0.01    0.21    0.03    0.00    0.01    0.00    0.00    0.00   99.54
08:14:27       4    0.18    0.02    0.15    0.02    0.00    0.00    0.00    0.00    0.00   99.63
08:14:27       5    0.26    0.01    0.27    0.02    0.00    0.00    0.00    0.00    0.00   99.44
```

Network Related

Network statistics (netstat)

- Netstat is used to display network connections and routing tables. (man netstat(8)).
    - Active network connections, listening ports, and established connections.
    - Statistics related to network protocols, such as TCP, UDP, and ICMP.
    - Multicast group information.
- Utilize `/proc/net` interfaces to provide the network information.
- Helpful in troubleshooting network connectivity issues, monitoring network activity, and analyzing network performance.

ethtool

- Focuses on querying and controlling network interface settings and statistics (man ethtool(8)).
- Provides detailed information about Ethernet devices, such as link status, speed, duplex mode, and driver information.
- Gathers statistics on network interface performance, such as packet counts and error statistics.
- Allows configuration of features like Wake-on-LAN, offloading capabilities, and flow control settings.

tcpdump

- It is a packet capture tool used to capture and analyze network traffic (man tcpdump(1)).
- It can filter the packets based on various criteria such as:
  - hostname filter: `tcpdump host 192.168.1.100`
  - Port filter: `tcpdump port 80`
  - Protocol filter: `tcpdump icmp`
  - Source filter: `tcpdump src 192.168.1.100`
  - Destination filer: `tcpdump dst 192.168.1.100`
  - Protocol flag filter: `tcpdump 'tcp[13] & 1 != 0` (captures TCP packets with the SYN flag set)
  - Logical operators: `tcpdump host 192.168.1.100 and port 80`

References

- Proc filesystem
  - https://docs.kernel.org/filesystems/proc.html
- Sys filesystem
  - https://docs.kernel.org/filesystems/sysfs.html
- Dev filesystem
  - https://tldp.org/LDP/Linux-Filesystem-Hierarchy/html/dev.html
- Debug filesystem
  - https://docs.kernel.org/filesystems/debugfs.html
- Brendan Gregg's post about Linux performance and observability tools.
  - https://www.brendangregg.com/linuxperf.html
- TCPDUMP tutorial
  - https://danielmiessler.com/p/tcpdump/