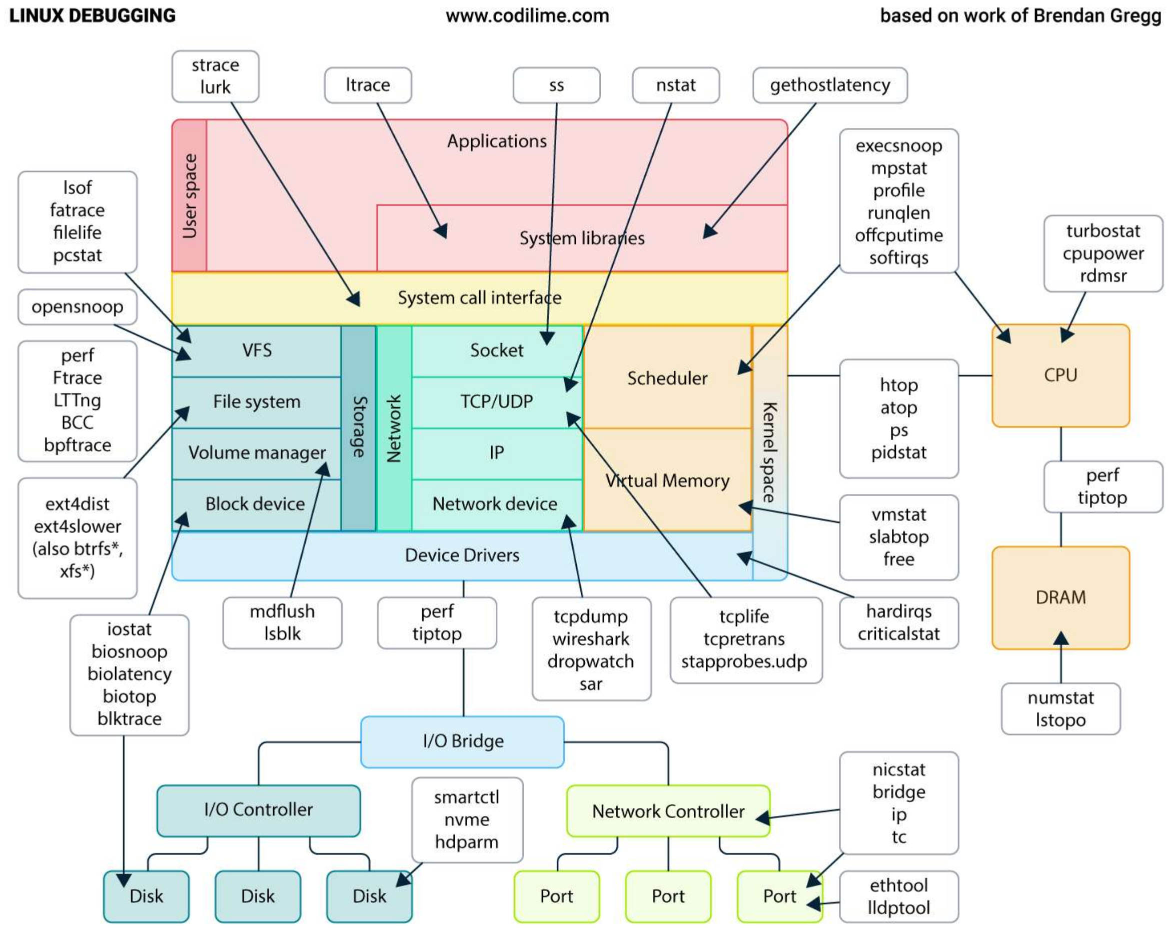
The diagram overview

So without further ado let's start with the diagram itself:

Fig.1: Linux observability and debugging tools



Note: The following blogpost is also available as a single-page cheat sheet for your convenience. You can find the file for download at the end of the article.

Linux observability and debugging tools list

As for the programs here is the list of them in alphabetical order:

atop - is a top-like utility but concentrating more on system resources such as I/O, network, cpu.

Documentation ↗

BCC - is an eBPF compiler for programs that can be attached to kprobes (Kernel Probes enable dynamically breaking into any kernel routine to collect debugging and performance information non-disruptively). BCC allows programs to be written in C, Python or Lua. It includes several utilities as examples that will be mentioned later.

Documentation ↗

biolatency - summarizes block device I/O latency as a histogram. This tool is included with BCC.

Documentation ↗

biosnoop - traces block device I/O and print details, including issuing PID. This tool is included with BCC.

Documentation ↗

biotop - is a top-like utility for I/O usage. This tool is included with BCC.

Documentation ↗

blktrace - generates traces of the I/O traffic on block devices.

Documentation ↗

bpftrace - is a high-level programming language dedicated to enhancing tracing for Linux systems. Similar to BCC.

Documentation ↗

bridge - is part of iproute2 concentrating on L2 traffic in Linux that is passing through a bridge. Aside from listing bridges, it also can show FDB, MDB (unicast and multicast forwarding database) or VLAN/VXLAN data.

Documentation ↗

cpupower - is a collection of tools to examine and tune power saving settings (such as frequency or Intel Turbo Boost status).

Documentation ↗

criticalstat - reports long atomic critical sections in kernel with useful stacktraces showing their origins. This tool is included with BCC.

Documentation ↗

dropwatch - shows the reason why a packet was dropped allowing pinpoint debugging to NIC, firewall, routing or others.

Documentation ↗

ethtool - shows hardware counter configuration and other data (such as inserted SFP module) that can be extracted from NIC. Can also be used to configure NIC on a hardware level, such as hashing, queues and other knobs that can enhance performance.

Documentation ↗

execsnoop - traces the usage of exec() system calls. This utility is ideal for the monitoring of short-lived processes that would be easily missed in top/ps. This tool is included with BCC.

Documentation ↗

ext4dist - summarizes ext4 operation latency. This toolis included with BCC. Similar tools for XFS and BTRFS exist as well.

Documentation ↗

ext4slower - traces slow ext4 file operations, with per-event details. This tool is included with BCC. Similar tools for XFS and BTRFS exist as well.

Documentation ↗

fatrace - reports file access events (from all processes). Its main purpose is to help find the purpose of HDD not going to sleep.

Documentation ↗

filelife - with the help of eBPF, this utility helps trace short-lived files for performance purposes.

Documentation ↗

free - displays the amount of free and used memory in the system.

Documentation ↗

ftrace - is an internal tracer designed to help out developers and designers of systems to find out what is going on inside the kernel. It's especially useful for analyzing latencies and performance issues affecting user space.

Documentation ↗

gethostlatency - is a utility designed to solve issues with host name resolution. It shows latency for getaddrinfo/gethostbyname calls. This tool is included with BCC.

Documentation ↗

hardirqs - summarizes the time spent servicing hard interrupts (IRQ created by physical hardware) and shows this time as either totals or histogram distributions. This tool is included with BCC.

Documentation ↗

hdparm - reads and writes hardware disk registers such as performance/power mode, encryption, and spinning status.

Documentation ↗

htop - provides similar information to top command but with much more detail, such as per CPU usage. Allows the user to quickly glance at what is happening on the system.

Documentation ↗

iostat - is used for monitoring system input/output device loading by observing the time the devices are active in relation to their average transfer rates.

Documentation ↗

ip - is part of iproute2, concentrating on several layers of OSI level, from interface/link statistics, through L3 routing up to L4 traffic policy.

Documentation ↗

lldptool - reads and interprets LLDP packets sent by the attached switch(es). Allows to easily identify which port Linux is connected to and advertises Linux’s presence to neighboring switch(es).

Documentation ↗

lsblk - list all block devices (physical and virtual) and their topology in the underlying system.

Documentation ↗

lsof - outputs a list of currently open files (hence the name). File can be a regular file, directory, socket, or the network connection/port on which the process is listening.

Documentation ↗

lstopo - shows system hardware configuration along with device NUMA assignment in a nice graphical output.

Documentation ↗

ltrace - is a similar program to strace but more concentrated on dynamic library calls which are called by the executed process and the signals which are received by that process. It can also intercept and print the system calls executed by the program.

Documentation ↗

LTTng - The Linux Trace Toolkit (LTTng in short) is an open-source software toolkit that one can use to trace the Linux kernel, user applications, and user libraries concurrently.

Documentation ↗

lurk - is similar to strace with some optimizations made for readability.

Documentation ↗

mdflush - traces flush events by md, the Linux multiple device driver (used for the software RAID). This tool is included with BCC.

Documentation ↗

mpstat - prints CPU usage statistics divided per CPU in SMP (symmetric multiprocessing) systems.

Documentation ↗

nicstat - prints out network statistics for all network cards including PPS, throughput, packet size, etc.

Documentation ↗

nstat - is a simple tool designed to monitor kernel SNMP counters and network interface stats.

Documentation ↗

numastat - show per-NUMA-node memory statistics for processes and the operating system.

Documentation ↗

nvme - is an NVM storage command line utility. Among other uses, it can read S.M.A.R.T events, NAND, PCIE statistics and send custom commands to underlying devices.

Documentation ↗

offcputime - summarizes off-CPU (where time is spent waiting while blocked on I/O, locks, timers, paging/swapping, etc.) time by kernel stack trace. This tool is included with BCC.

Documentation ↗

opensnoop - traces open() syscalls, showing the file name (pathname) and returned file descriptor number (or -1, for error).

Documentation ↗

pcstat - gets page cache statistics for files in order to provide an answer as to whether Linux is caching data or not.

Documentation ↗

perf - is a performance analysis tool in Linux. It's a userspace controlling utility, accessed from the command line which provides a number of subcommands such as: stat, top, record, report, etc. It supports hardware performance counters, tracepoints, software performance counters, and dynamic probes.

Documentation ↗

pidstat - is used to monitor every individual task currently being managed by the Linux kernel on the Linux system. IT can monitor every task on the system, including the child's task of any task, along with details such as CPU usage or disk I/O.

Documentation ↗

ps - shows information about current processes. Although not the most sophisticated tool, it's always available on Linux.

Documentation ↗

rdmsr - reads CPU model-specific registers (MSR). MSRs are control registers provided by the processor implementation so that system software can interact with a variety of features, including performance monitoring, checking processor status, etc.

Documentation ↗

runqlen - summarizes scheduler queue length as a histogram. It can be used to identify imbalances such as processes occupying a CPU causing queuing. This tool is included with BCC.

Documentation ↗

sar - collects, reports or saves system statistics. Aside from network statistics it can be used to monitor other devices, such as disks as well.

Documentation ↗

slabtop - displays kernel slab cache information in real time. A slab is a set of one or more contiguous pages of memory while a slab cache is a “container” of multiple slabs of the same type.

Documentation ↗

smartctl - reads S.M.A.R.T data from the underlying disk device. Data includes sector remapping, errors, rereads, logs and other statistics reflecting the health and performance of storage devices. Older versions of smartctl were unable to access NVM devices. In such cases, the nvme utility should be used.

Documentation ↗

softirqs - summarizes the time spent servicing soft IRQs (soft interrupts), and can show this time as either totals or histogram distributions. This tool is included with BCC.

Documentation ↗

ss - is the tool that replaced the depreciated netstat utility. Allows viewing of port binding on the running system along with process names, connection statutes. The CLI is almost identical to its predecessor.

Documentation ↗

stapprobes.udp - is a part of the SystemTap (stap for short) utility used for gathering information about the running Linux system. Among other things, it provides probe points for UDP activity.

Documentation ↗

strace - runs the specified command until it exits. It intercepts and records the system calls which are called by a process and the signals which are received by a process.

Documentation ↗

tc - is part of iproute2 concentrating traffic control settings. Aside from displaying QoS counters it can also deal with traffic offloading (tc flower).

Documentation ↗

tcpdump - is a network sniffer that can display traffic activity happening on selected (or all) interfaces. Due to CLI's nature, it provides invaluable information on traffic that is happening on systems where a graphical environment is not an option..

Documentation ↗

tcplife - traces TCP sessions in systems and summarizes their lifespan. This tool is included with BCC.

Documentation ↗

tcpretrans - shows possible issues with TCP connections by displaying retransmits and other details.

Documentation ↗

tiptop - displays hardware performance counters for Linux tasks. It's similar to the top utility but enriched by hardware counters.

Documentation ↗

turbostat - shows CPU topology, temperature, frequency and idle statistics.

Documentation ↗

vmstat - reports information about processes, memory, paging, block I/O, traps, disks and cpu activity in defined intervals.

Documentation ↗

wireshark - is a graphical tool complementing tcpdump in many ways. Due to the several subtools included, as well as a vast number of supported protocols, it makes network debugging easier than tcpdump.

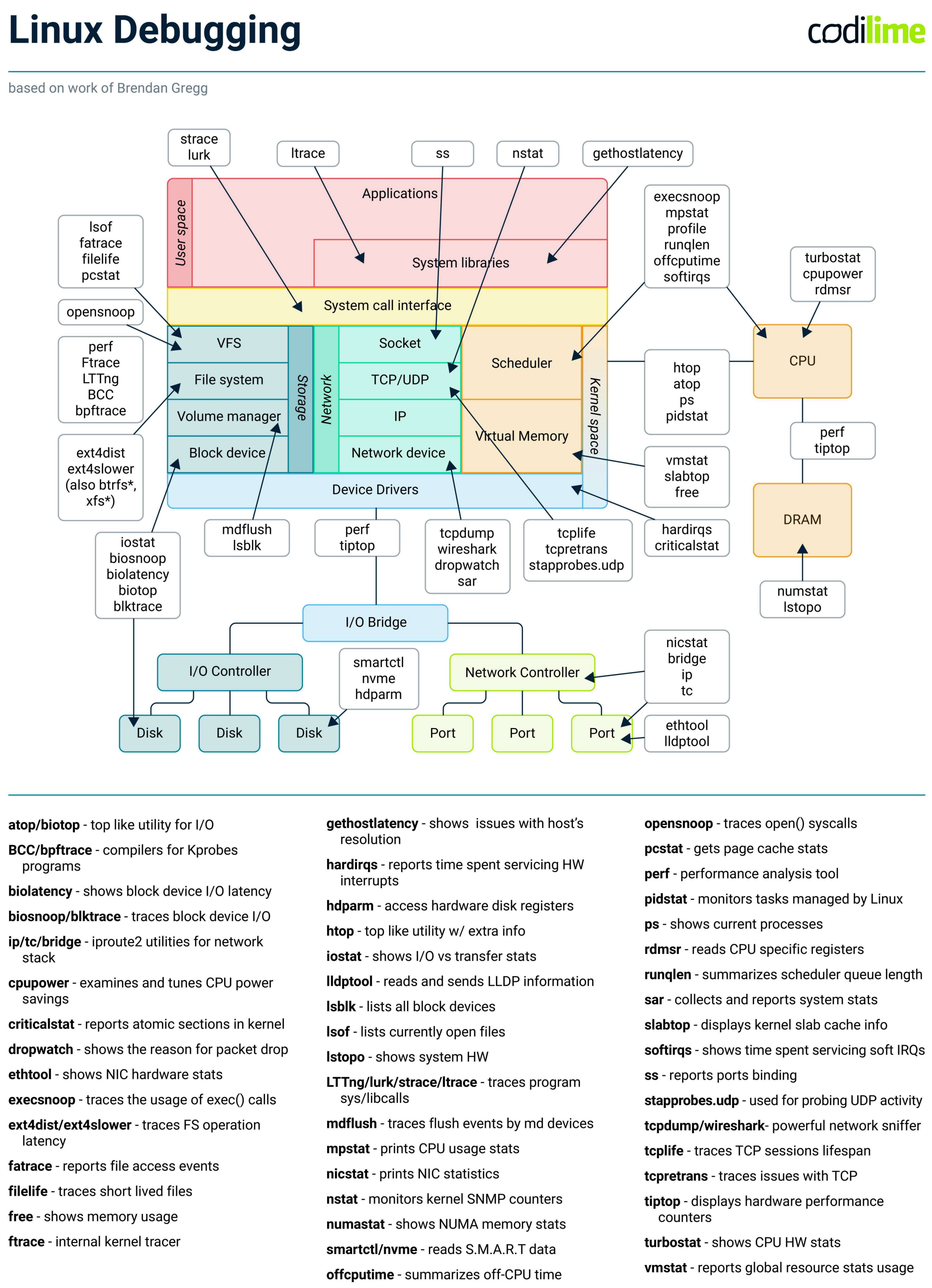
Documentation ↗

Summary

While the picture and list are far from being comprehensive, this is a good place to start. If you think that there are tools missing, please contact us and we will update this blog post.

Linux observability and debugging tools cheat sheet to download

For those who prefer a printable version, we have provided the cheat sheet in PDF format here.



Linux Debugging

codilime

based on work of Brendan Gregg

atop/biotop - top like utility for I/O

BCC/bpftrace - compilers for Kprobes programs

biolatency - shows block device I/O latency

biosnoop/blktrace - traces block device I/O

ip/tc/bridge - iproute2 utilities for network stack

cpupower - examines and tunes CPU power savings

criticalstat - reports atomic sections in kernel

dropwatch - shows the reason for packet drop

ethtool - shows NIC hardware stats

execsnoop - traces the usage of exec() calls

ext4dist/ext4slower - traces FS operation latency

fatrace - reports file access events

filelife - traces short lived files

free - shows memory usage

ftrace - internal kernel tracer

gethostlatency - shows issues with host's resolution

hardirqs - reports time spent servicing HW interrupts

hdparm - access hardware disk registers

htop - top like utility w/ extra info

iostat - shows I/O vs transfer stats

lldptool - reads and sends LLDP information

lsblk - lists all block devices

lsof - lists currently open files

lstopo - shows system HW

LTTng/lurk/strace/ltrace - traces program sys/libcalls

mdflush - traces flush events by md devices

mpstat - prints CPU usage stats

nicstat - prints NIC statistics

nstat - monitors kernel SNMP counters

numastat - shows NUMA memory stats

smartctl/nvme - reads S.M.A.R.T data

offcputime - summarizes off-CPU time

opensnoop - traces open() syscalls

pcstat - gets page cache stats

perf - performance analysis tool

pidstat - monitors tasks managed by Linux

ps - shows current processes

rdmsr - reads CPU specific registers

runqlen - summarizes scheduler queue length

sar - collects and reports system stats

slabtop - displays kernel slab cache info

softirqs - shows time spent servicing soft IRQs

ss - reports ports binding

stapprobes.udp - used for probing UDP activity

tcpdump/wireshark- powerful network sniffer

tcplife - traces TCP sessions lifespan

tcpretrans - traces issues with TCP

tiptop - displays hardware performance counters

turbostat - shows CPU HW stats

vmstat - reports global resource stats usage