

# Linux 性能调优工具的 9 张图

Linux爱好者 2023-08-14 11:50 发表于浙江



Linux爱好者

点击获取《每天一个Linux命令》系列和精选Linux技术资源。「Linux爱好者」日常分享 Linu... >

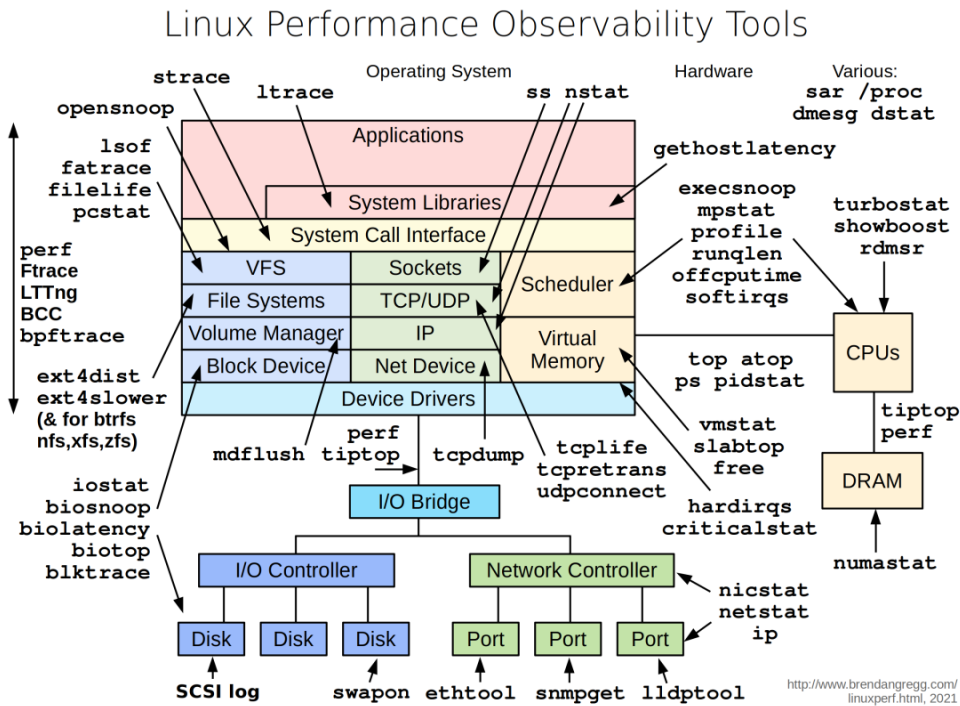
75篇原创内容

公众号

转自：网络

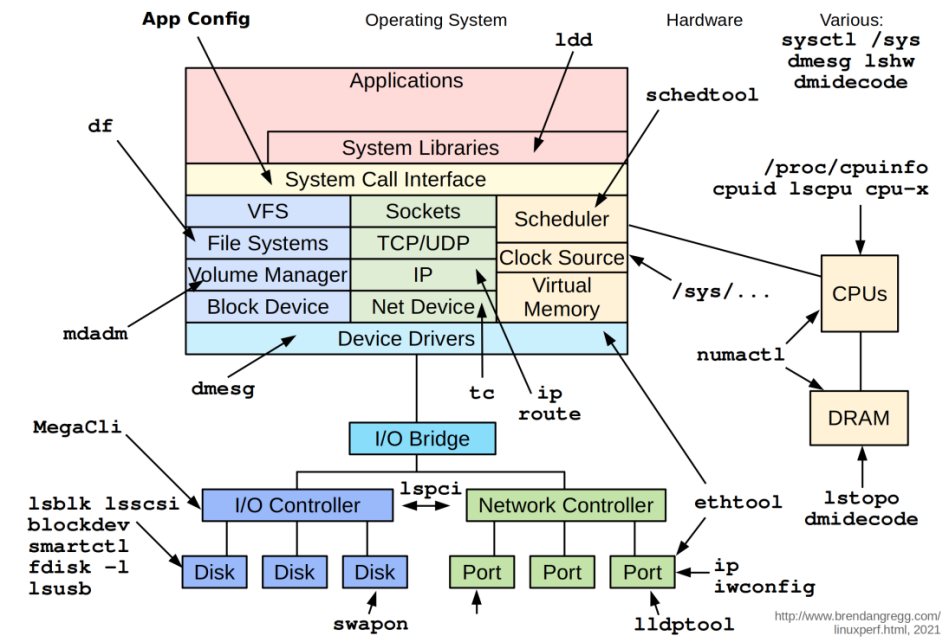
这里包含Linux 性能资料的工具图。它们展示了：Linux 可观察性工具、Linux 静态性能分析工具、Linux 基准测试工具、Linux 调优工具和Linux sar。

## 性能观察工具：



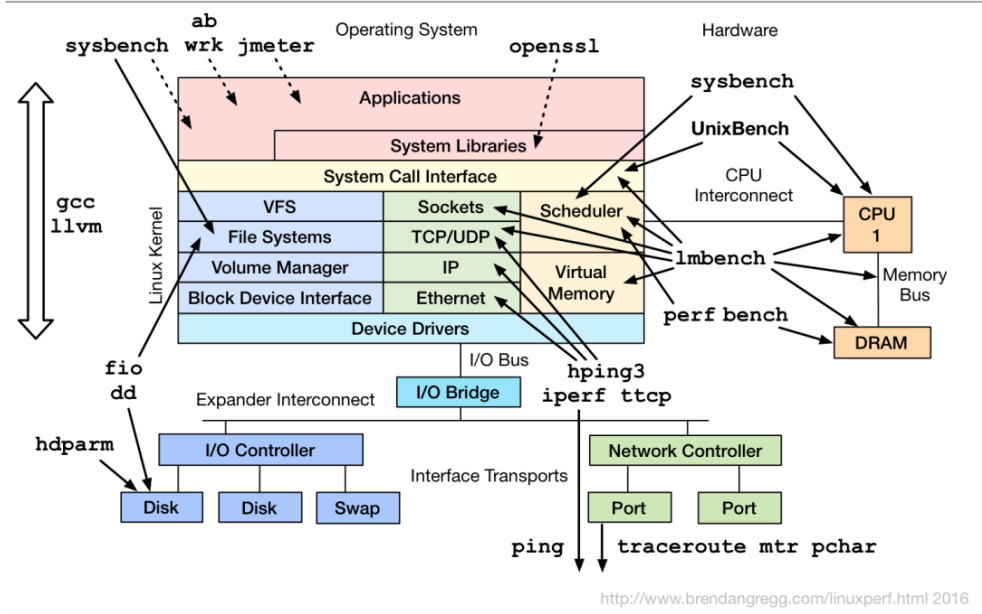
## 静态性能工具

Linux Static Performance Tools



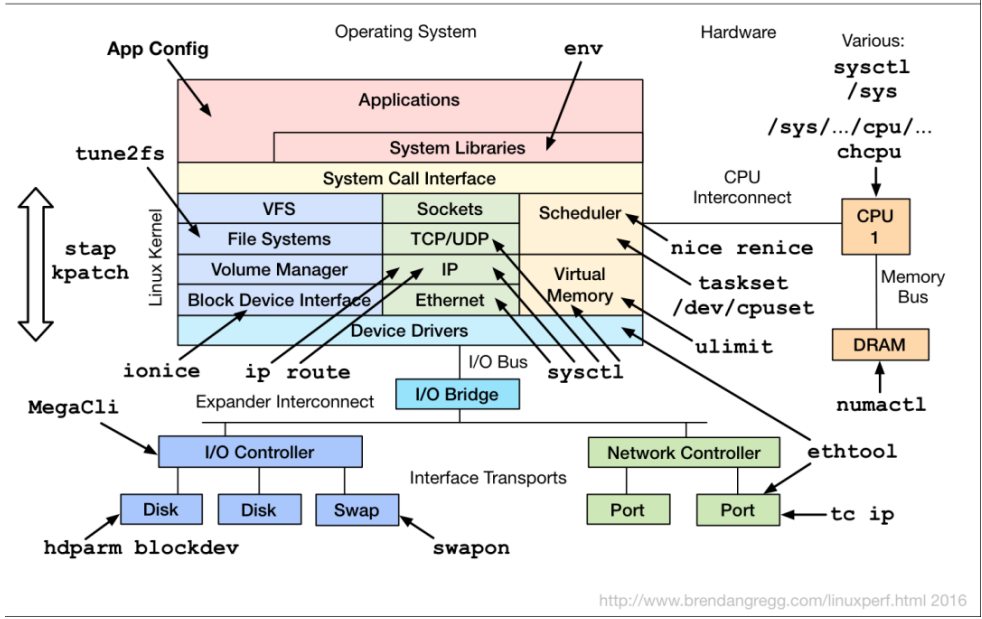
性能压测工具：

Linux Performance Benchmark Tools



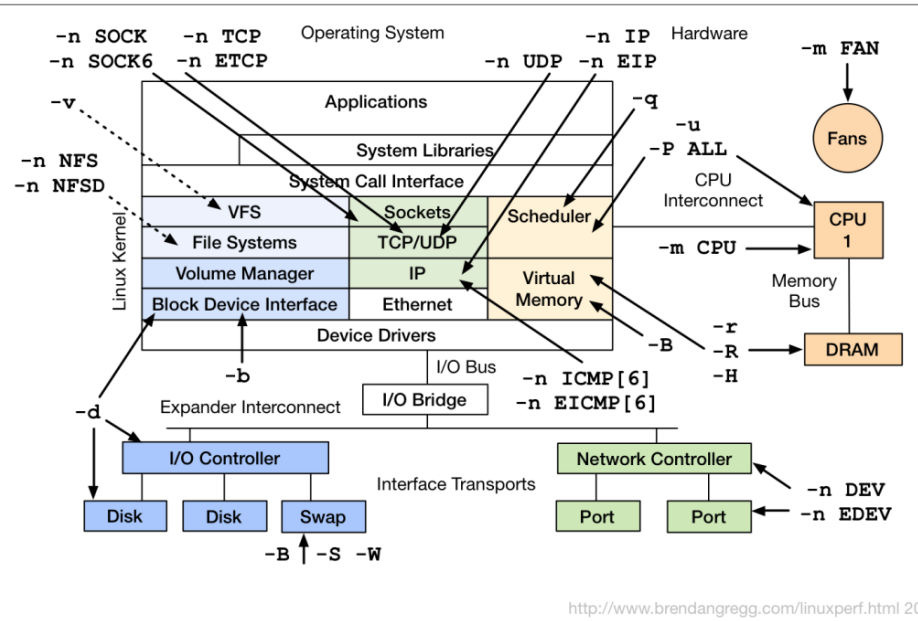
性能调优工具

Linux Performance Tuning Tools



sar

Linux Performance Observability: sar



perf-tools

The diagram illustrates the Linux Performance Observability Tools (perf-tools) stack, showing the flow of data and the components involved in performance monitoring.

**Operating System Layer:**

- Applications:** The top layer of the stack.
- System Libraries:** Below Applications.
- System Call Interface:** Below System Libraries.
- VFS (Virtual File System):** Below System Call Interface.
- File Systems:** Below VFS.
- Volume Manager:** Below File Systems.
- Block Device Interface:** Below Volume Manager.
- Sockets:** Below System Call Interface.
- TCP/UDP:** Below Sockets.
- IP:** Below TCP/UDP.
- Ethernet:** Below IP.
- Scheduler:** Below System Call Interface.
- Virtual Memory:** Below Scheduler.

**Hardware Layer:**

- CPU 1:** The hardware component that interacts with the Operating System.
- Memory Bus:** Connects the CPU to the DRAM.
- DRAM (Dynamic Random Access Memory):** The hardware component that stores data.

**Various: tpoint:**

- CPU 1:** The hardware component that interacts with the Operating System.
- Memory Bus:** Connects the CPU to the DRAM.
- DRAM (Dynamic Random Access Memory):** The hardware component that stores data.

**Tools and Data Flow:**

- opensnoop, syscount, execsnoop:** Tools that monitor system calls and execution.
- funccount, functrace, funcslower, funcgraph, kprobe:** Tools that monitor function calls and execution.
- iosnoop, iolatility, bitesize:** Tools that monitor I/O operations.
- tcpretrans:** Tool that monitors TCP retransmissions.

**Interface Transports:**

- I/O Controller:** Connects the Operating System to the Disk and Swap.
- Network Controller:** Connects the Operating System to the Port.

**Other Components:**

- I/O Bridge:** Connects the I/O Controller to the Network Controller.
- Expander Interconnect:** Connects the I/O Controller to the I/O Bridge.
- Interface Transports:** Connects the I/O Controller to the Network Controller.

## 追踪工具

The diagram illustrates the Linux bcc/BPF Tracing Tools architecture, showing the flow of data and the tools used to trace various system components.

**System Components and Tracing Tools:**

- Applications:** Traced by `filetop`, `filelife`, `fileslower`, `vfscount`, `vfsstat`, `opensnoop`, `statsnoop`, `syncsnoop`, `c*`, `java*`, `node*`, `php*`, `python*`, `ruby*`, `mysqld_qlower`, `bashreadline`, `gethostlatency`, `memleak`, `sslsniff`, and `Other: capable`.
- System Libraries:** Traced by `ucalls`, `uflow`, `ugc`, `uobjnew`, `ustat`, and `uthreads`.
- System Call Interface:** Traced by `cachestat`, `cachetop`, `dcstat`, `dcsnnoop`, `mountsnnoop`, `execsnnoop`, `pidpersec`, `cpudist`, `runqlat`, `runqlen`, `deadlock`, `detector`, and `cpuunclaimed`.
- File Systems:** Traced by `trace`, `argdist`, `funccount`, `funclslower`, `funclatency`, `stackcount`, and `profile`.
- Volume Manager:** Traced by `mdflush`.
- Block Device Interface:** Traced by `btrfsdist`, `btrfsslower`, `ext4dist`, `ext4slower`, `xfsdist`, `xfsslower`, `zfsdist`, and `zfslower`.
- Device Drivers:** Traced by `biotop`, `biosnoop`, `biolatency`, `bitesize`, `hardirqs`, `ttysnoop`, `tcpdump`, `tcpcli`, `tcptracer`, `tcpconnect`, `tcpaccept`, `tcpconnlat`, and `tcpretrans`.
- Scheduler:** Traced by `offcputime`, `wakeuptime`, and `offwaketime`.
- Virtual Memory:** Traced by `softirqs`, `oomkill`, `memleak`, and `slabratetop`.

**Hardware Components:**

- DRAM:** Connected to the CPU.
- CPU:** The central processing unit, receiving input from the DRAM and the system.

**Tracing Tools:**

- `trace`, `argdist`, `funccount`, `funclslower`, `funclatency`, `stackcount`, and `profile` are used to trace the File Systems.
- `mdflush` is used to trace the Volume Manager.
- `btrfsdist`, `btrfsslower`, `ext4dist`, `ext4slower`, `xfsdist`, `xfsslower`, `zfsdist`, and `zfslower` are used to trace the Block Device Interface.
- `biotop`, `biosnoop`, `biolatency`, `bitesize`, `hardirqs`, `ttysnoop`, `tcpdump`, `tcpcli`, `tcptracer`, `tcpconnect`, `tcpaccept`, `tcpconnlat`, and `tcpretrans` are used to trace the Device Drivers.
- `offcputime`, `wakeuptime`, and `offwaketime` are used to trace the Scheduler.
- `softirqs`, `oomkill`, `memleak`, and `slabratetop` are used to trace the Virtual Memory.

**bpftrace/eBPF Tools**

Tools monitoring the stack layers:

- Applications / System Libraries:** opensnoop, statsnoop, bashreadline, gethostlatency
- System Call Interface:** vfstcount, vfststat, writeback, bpftrace, xfsdist, mdflush, biosnoop, biolatility, bitesize
- VFS / File Systems / Volume Manager / Block Device:** vfstcount, vfststat, writeback, bpftrace, xfsdist, mdflush, biosnoop, biolatility, bitesize
- Sockets / TCP/UDP / IP / Net Device:** tcpconnect, tcpaccept, tcpretrans, tcpdrop
- Scheduler / Virtual Memory:** execsnoop, pidpersec, cpuwalk, runqlat, runqlen, offcputime, oomkill

Other: capable

Diagram by Brendan Gregg, early 2019, <https://github.com/iovisor/bpftools>

## BPF性能工具:

**New tools** developed for the book **BPF Performance Tools: Linux System and Application Observability** by Brendan Gregg (Addison Wesley, 2019), which also covers **prior BPF tools**

