

# Linux问题分析与性能优化

极客重生 极客重生 2021-08-20 08:05

收录于话题

#解决问题高手系列 1 #深入理解Linux系统 29



## 极客重生

大厂资深工程师，专注硬核知识分享，包括云计算，后台开发，网络，高性能服务器，L... >

83篇原创内容

公众号

## 目录

- 排查顺序
- 方法论
- 性能分析工具
- CPU分析思路
- 内存分析思路
- IO分析思路
- 网络分析思路
- 基准测试工具
- 参考

## 排查顺序

整体情况：

1. top/htop/atop 命令查看进程/线程、CPU、内存使用情况，CPU使用情况；
2. dstat 2 查看CPU、磁盘IO、网络IO、换页、中断、切换，系统I/O状态；
3. vmstat 2 查看内存使用情况，内存状态；
4. iostat -d -x 2 查看所有磁盘的IO情况，系统I/O状态；
5. iotop 查看IO靠前的进程，系统的I/O状态；
6. perf top 查看占用CPU最多的函数，CPU使用情况；
7. perf record -ag -- sleep 15;perf report 查看CPU事件占比，调用栈，CPU使用情况；
8. sar -n DEV 2 查看网卡的吞吐，网卡状态；

9. `/usr/share/bcc/tools/filetop -C` 查看每个文件的读写情况，系统的I/O状态；
10. `/usr/share/bcc/tools/opensnoop` 显示正在被打开的文件，系统的I/O状态；
11. `mpstat -P ALL 1` 单核CPU是否被打爆；
12. `ps aux --sort=-%cpu` 按CPU使用率排序，找出CPU消耗最多进程；
13. `ps -eo pid,comm,rss | awk '{m=$3/1e6;s["*"]+=m;s[$2]+=m} END{for (n in s) printf "%10.3f GB %s\n",s[n],n}' | sort -nr | head -20` 统计前20内存占用；
14. `awk 'NF>3{s["*"]+=s[$1]==$3*$4/1e6} END{for (n in s) printf "%10.1f MB %s\n",s[n],n}' /proc/slabinfo | sort -nr | head -20` 统计内核前20slab的占用；

进程分析，进程占用的资源：

1. `pidstat 2 -p` 进程号 查看可疑进程CPU使用率变化情况；
2. `pidstat -w -p` 进程号 2 查看可疑进程的上下文切换情况；
3. `pidstat -d -p` 进程号 2 查看可疑进程的IO情况；
4. `lsof -p` 进程号 查看进程打开的文件；
5. `strace -f -T -tt -p` 进程号 显示进程发起的系统调用；

协议栈分析，连接/协议栈状态：

1. `ethtool -S` 查看网卡硬件情况；
2. `cat /proc/net/softnet_stat/ifconfig eth1` 查看网卡驱动情况；
3. `netstat -nat|awk '{print awk $NF}'|sort|uniq -c|sort -n` 查看连接状态分布；
4. `ss -ntp` 或者 `netstat -ntp` 查看连接队列；
5. `netstat -s` 查看协议栈情况；

## 方法论

RED方法：监控服务的请求数（Rate）、错误数（Errors）、响应时间（Duration）。Weave Cloud在监控微服务性能时提出的思路。

USE方法：监控系统资源的使用率（Utilization）、饱和度（Saturation）、错误数（Errors）。

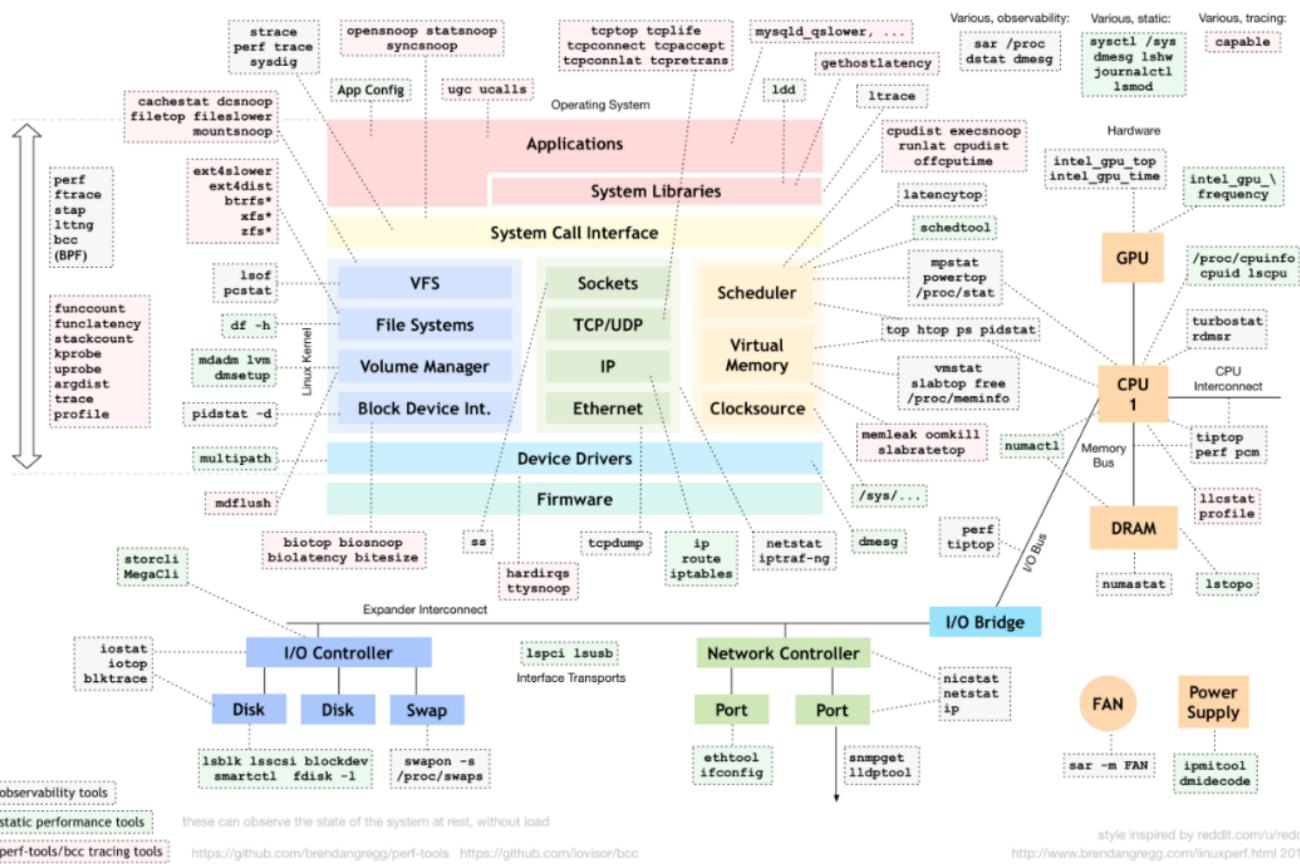
## 常见指标分类 (USE 法)

资源	类型	性能指标
CPU	使用率	CPU 使用率
CPU	饱和度	运行队列长度或平均负载
CPU	错误数	硬件CPU错误数
内存	使用率	已用内存百分比或SWAP用量百分比
内存	饱和度	内存换页量
内存	错误数	内存分配失败或OOM
存储设备I/O	使用率	设备I/O时间百分比
存储设备I/O	饱和度	等待队列长度或延迟
存储设备I/O	错误数	I/O错误数
文件系统	使用率	已用容量百分比
文件系统	饱和度	已用容量百分比
文件系统	错误数	文件读写错误数
网络	使用率	带宽使用率
网络	饱和度	重传报文数
网络	错误数	网卡收发错误数、丢包数
文件描述符	使用率	已用文件描述符数百分比
连接跟踪	使用率	已用连接跟踪数百分比
连接数	饱和度	TIMEWAIT 状态连接数



## 性能分析工具

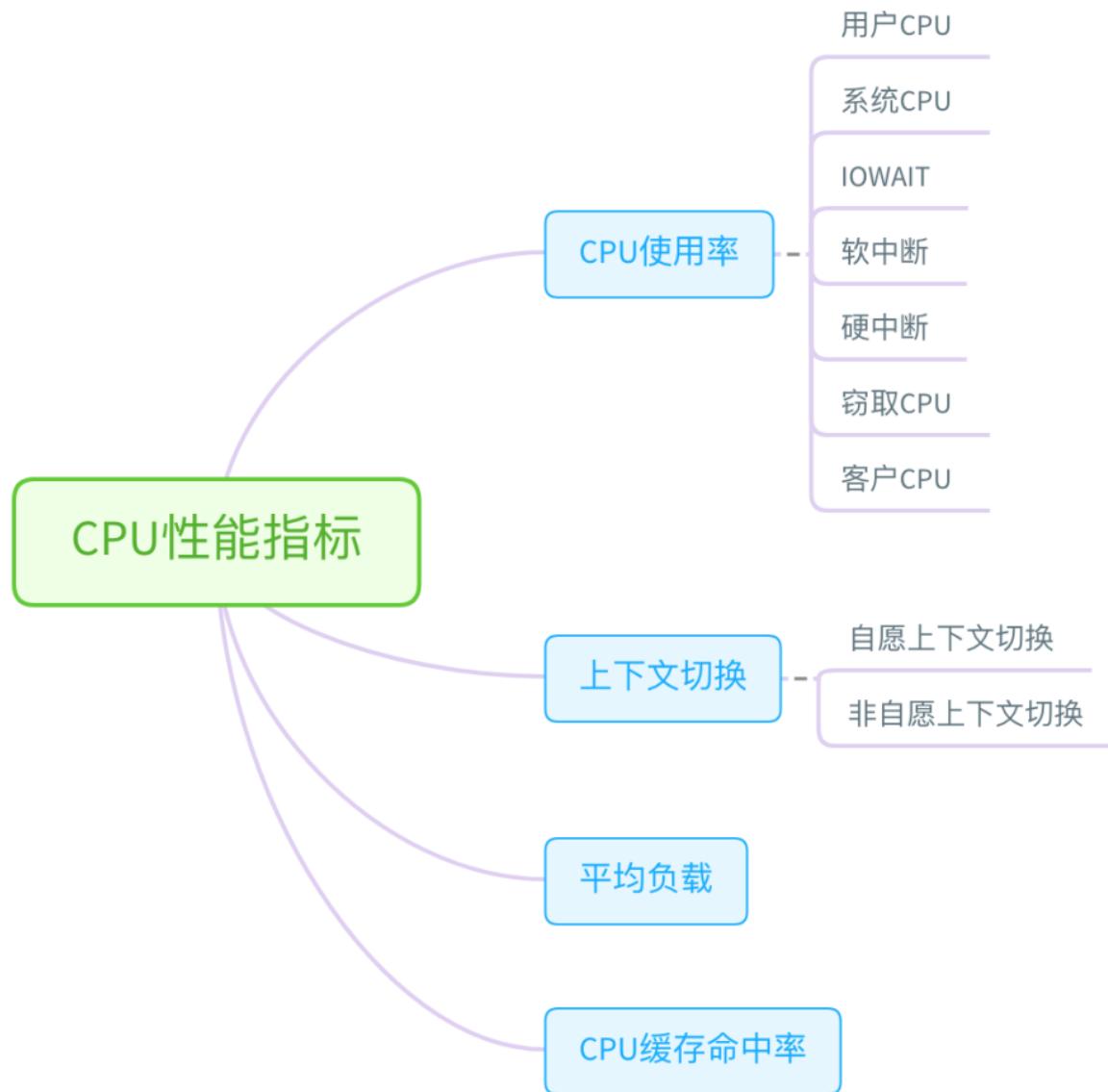
## Linux Performance Tools



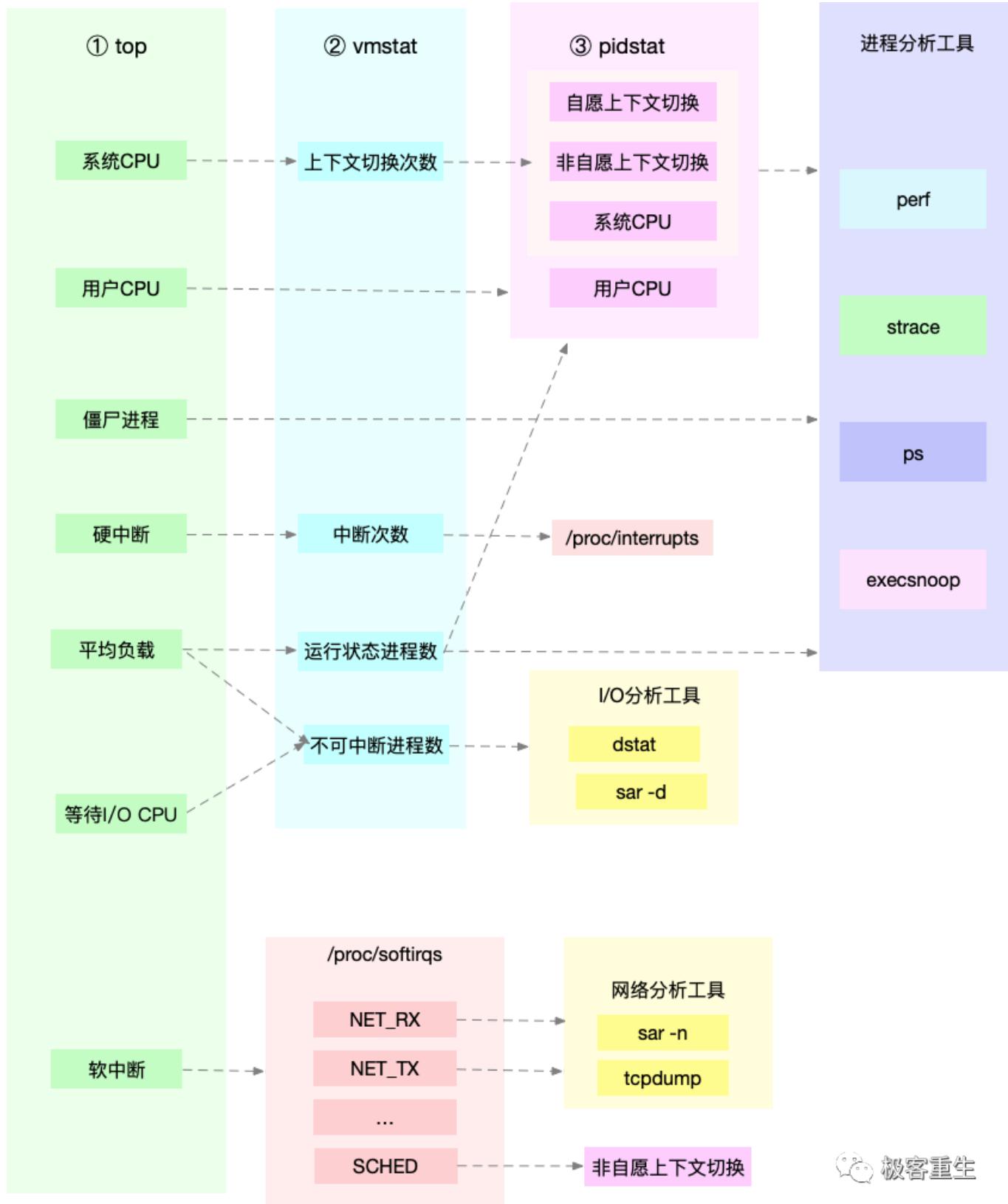
Linux 内核的各个子系统出发，汇总了对各个子系统进行性能分析时，你可以选择的工具。不过，虽然这个图是性能分析最好的参考资料之一，它其实还不够具体。比如，当你需要查看某个性能指标时，这张图里对应的子系统部分，可能有多个性能工具可供选择。但实际上，并非所有这些工具都适用，具体要用哪个，还需要你去查找每个工具的手册，对比分析做出选择。

## CPU分析思路

首先，从 CPU 的角度来说，主要的性能指标就是 CPU 的使用率、上下文切换以及 CPU Cache 的命中率等。下面这张图就列出了常见的 CPU 性能指标。



极客重生

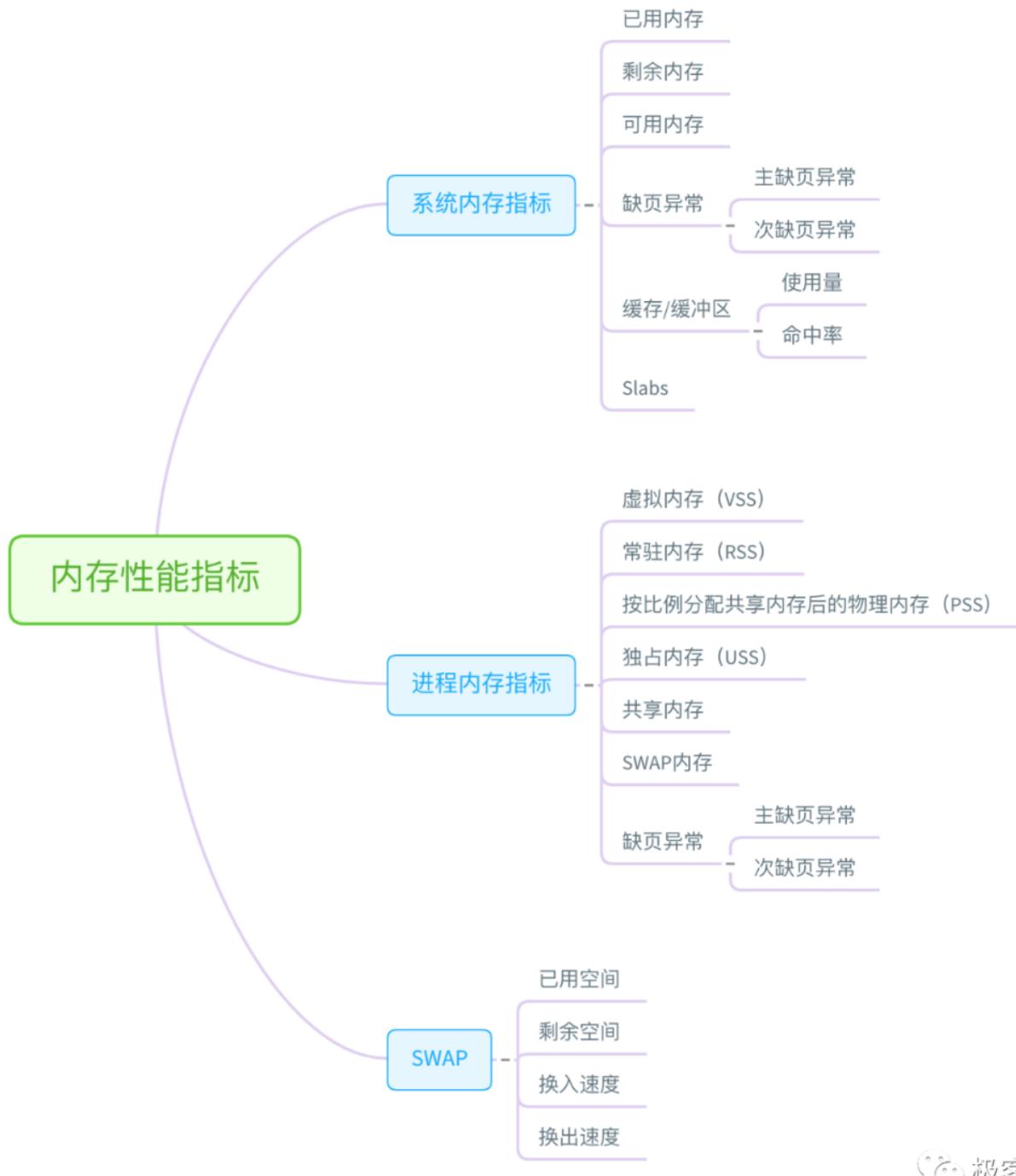


## CPU性能工具

性能指标	性能工具	说明
平均负载	uptime top /proc/loadavg	uptime最简单；top提供了更全的指标；/proc/loadavg常用于监控系统
系统CPU使用率	vmstat mpstat top sar /proc/stat	top、vmstat、mpstat 只可以动态查看，而sar 还可以记录历史数据；/proc/stat 是其他性能工具的数据来源，也常用于监控
进程CPU使用率	top ps pidstat htop atop	top和ps可以按CPU使用率给进程排序，而pidstat只显示实际用了CPU的进程；htop 和atop以不同颜色显示更直观
系统上下文切换	vmstat	除了上下文切换次数，还提供运行状态和不可中断状态进程的数量
进程上下文切换	pidstat	注意加上 -w 选项
软中断	top mpstat /proc/softirqs	top提供软中断CPU使用率，而/proc/softirqs和mpstat提供了各种软中断在每个CPU上的运行次数
硬中断	vmstat /proc/interrupts	vmstat 提供总的中断次数，而/proc/interrupts提供各种中断在每个CPU上运行的累积次数
网络	dstat sar tcpdump	dstat和sar提供总的网络接收和发送情况，而tcpdump则是动态抓取正在进行的网络通讯
I/O	dstat sar	dstat和sar都提供了I/O的整体情况
CPU缓存	perf	使用 perf stat 子命令
CPU数	lscpu /proc/cpuinfo	lscpu更直观
事件剖析	perf、火焰图 execsnoop	perf和火焰图用来分析热点函数以及调用栈，execsnoop用来监测短时进程
动态追踪	ftrace bcc、systemtap	ftrace用于跟踪内核函数调用栈，而bcc和systemtap则用于跟踪内核或应用程序的执行过程（注意bcc要求内核 版本为4.14及以上版本）

# 内存分析思路

接着我们来看内存方面。从内存的角度来说，主要的性能指标，就是系统内存的分配和使用、进程内存的分配和使用以及 SWAP 的用量。下面这张图列出了常见的内存性能指标。



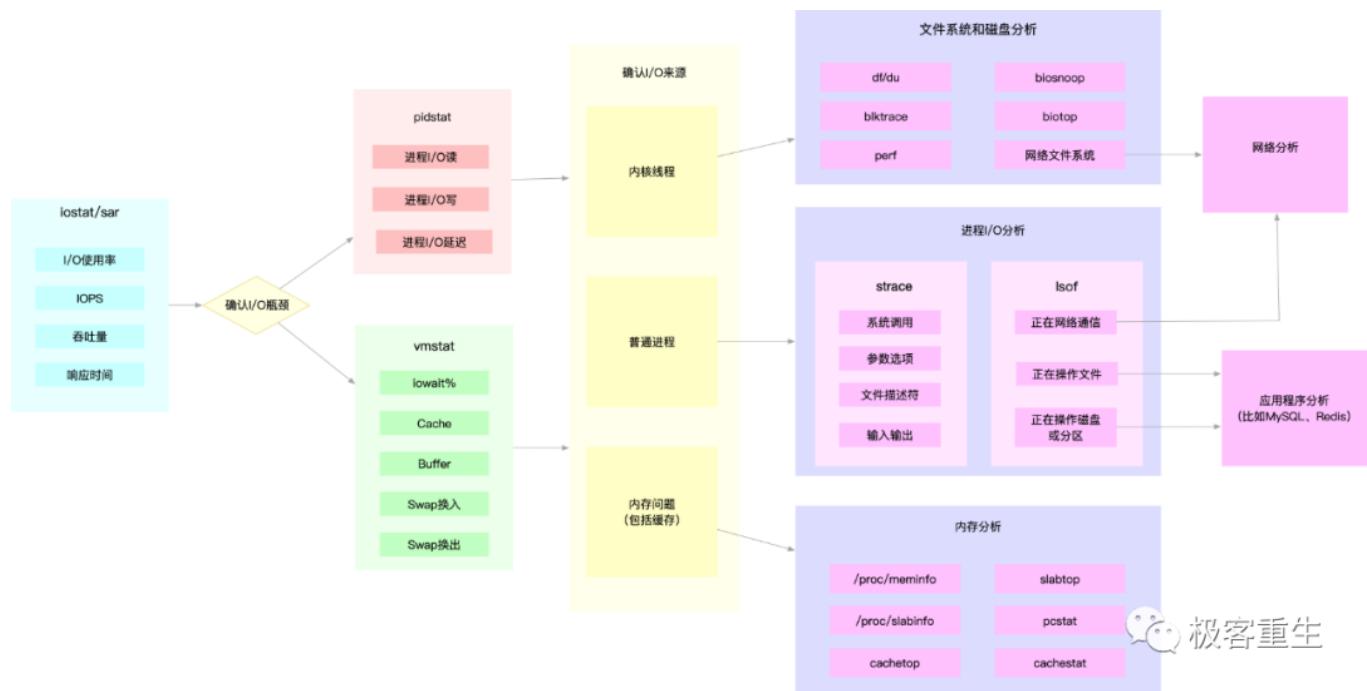
极客重生

## 内存性能工具

性能指标	性能工具	说明
系统已用、可用、剩余内存	free、vmstat、sar /proc/meminfo	free最为简单，而vmstat、sar更为全面； /proc/meminfo是其他工具的数据来源，也常用于监控系统中
进程虚拟内存、常驻内存、共享内存	ps、top、pidstat /proc/pid/stat /proc/pid/status	ps和top最简单，而pidstat则需要加上-r选项；/proc/pid/stat和/proc/pid/status是其他工具的数据来源，也常用于监控系统中
进程内存分布	pmap /proc/pid/maps	/proc/pid/maps是pmap的数据来源
进程Swap换出内存	top、/proc/pid/status	/proc/pid/status是top的数据来源
进程缺页异常	ps、top、pidstat	注意给pidstat加上-r选项
系统换页情况	sar	注意加上-B选项
缓存/缓冲区用量	free、vmstat、sar cachestat	vmstat最常用，而cachestat需要安装bcc
缓存/缓冲区命中率	cachetop	需要安装bcc
SWAP已用空间和剩余空间	free、sar	free最为简单，而sar还可以记录历史
Swap换入换出	vmstat、sar	vmstat最为简单，而sar还可以记录历史
内存泄漏检测	memleak、valgrind	memleak需要安装bcc，valgrind还可以在旧版本（如3.x）内核中使用
指定文件的缓存大小	pcstat	需要从 <a href="#">源码</a> 下载安装  极客重生

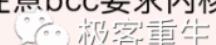
## IO分析思路

从文件系统和磁盘 I/O 的角度来说，主要性能指标，就是文件系统的使用、缓存和缓冲区的使用，以及磁盘 I/O 的使用率、吞吐量和延迟等。下面这张图列出了常见的 I/O 性能指标。



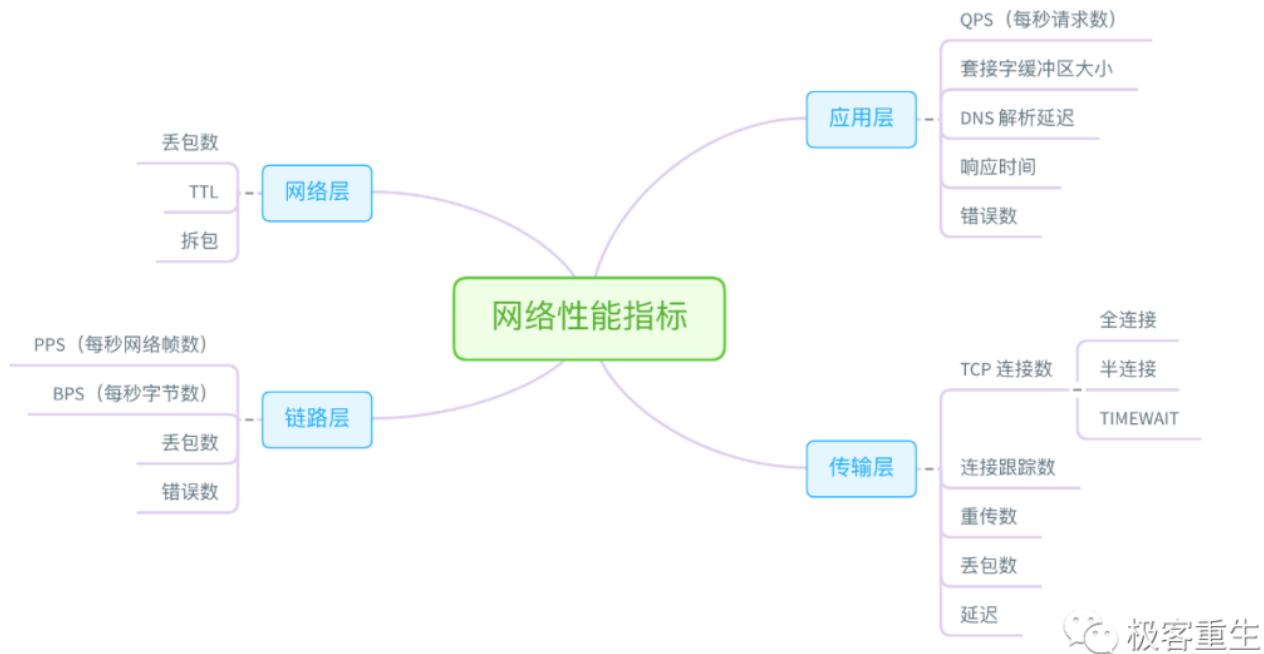
## 文件系统和磁盘I/O性能工具

性能指标	性能工具	说明
文件系统空间容量、使用量以及剩余空间	df	详细文档可以执行 info coreutils 'df invocation' 命令查询
索引节点容量、使用量以及剩余量	df	注意加上 -i 选项
页缓存和可回收Slab缓存	/proc/meminfo sar、vmstat	注意sar需要加上-r选项，而/proc/meminfo是其他工具的数据来源，也常用于监控
缓冲区	/proc/meminfo sar、vmstat	注意sar需要加上-r选项，而/proc/meminfo是其他工具的数据来源，也常用于监控
目录项、索引节点以及文件系统的缓存	/proc/slabinfo slabtop	slabtop更直观，而/proc/slabinfo常用于监控
磁盘 I/O 使用率、IOPS、吞吐量、响应时间、I/O平均大小以及等待队列长度	iostat、sar、dstat /proc/diskstats	iostat最为常用，注意使用 iostat -d -x 或 sar -d 选项；/proc/diskstats则是其他工具数据来源，也常用于监控
进程I/O大小以及I/O延迟	pidstat、iostop	注意使用 pidstat -d 选项
块设备 I/O 事件跟踪	blktrace	需要跟blkparse配合使用，比如 blktrace -d /dev/sda -o-   blkparse -i-
进程 I/O 系统调用跟踪	strace、perf trace	strace只可以跟踪单个进程，而perf trace还可以跟踪所有进程的系统调用
进程块设备I/O大小跟踪	biosnoop、biotop	需要安装bcc
动态追踪	ftrace bcc、systemtap	ftrace用于跟踪内核函数调用栈，而bcc和systemtap则用于跟踪内核或应用程序的执行过程（注意bcc要求内核版本>=4.1）

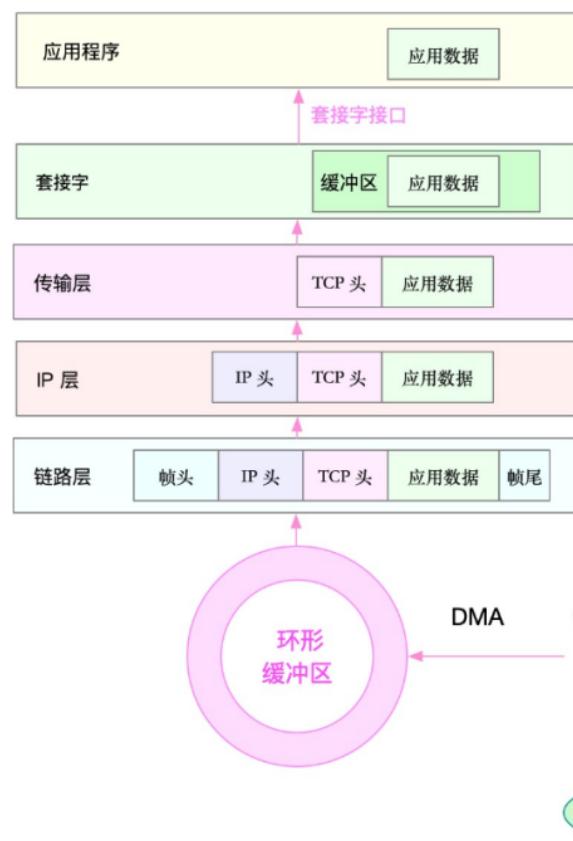


## 网络分析思路

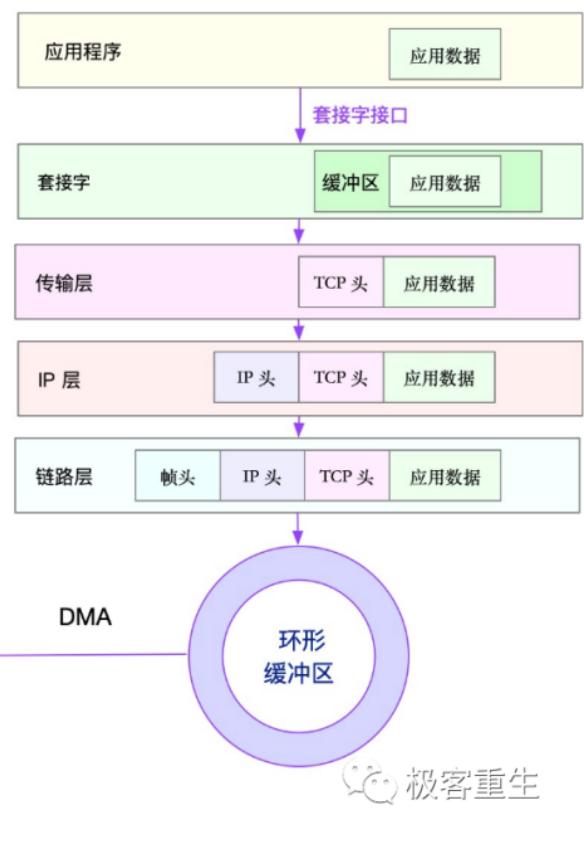
从网络的角度来说，主要性能指标就是吞吐量、响应时间、连接数、丢包数等。根据 TCP/IP 网络协议栈的原理，我们可以把这些性能指标，进一步细化为每层协议的具体指标。这里我同样用一张图，分别从链路层、网络层、传输层和应用层，列出了各层的主要指标。



网络接收流程



网络发送流程



## 网络性能工具

性能指标	性能工具	说明
吞吐量 (BPS)	sar、nethogs、iftop /proc/net/dev	分别可以查看网络接口、进程以及IP地址的网络吞吐量；/proc/net/dev常用于监控
吞吐量 (PPS)	sar、/proc/net/dev	注意使用sar -n DEV选项
网络连接数	netstat、ss	ss速度更快
网络错误数	netstat、sar	注意使用netstat -s或者sar -n EDEV/EIP选项
网络延迟	ping、hping3	ping基于ICMP，而hping3则基于TCP协议
连接跟踪数	conntrack /proc/sys/net/netfilter/nf_conntrack_count /proc/sys/net/netfilter/nf_conntrack_max	conntrack可用来查看所有连接跟踪的相信信息，nf_conntrack_count只是连接跟踪的数量，而nf_conntrack_max则限制了总的连接跟踪数量
路由	mtr、traceroute、route	route用于查询路由表，而mtr和traceroute则用来排查和定位网络链路中的路由问题
DNS	dig、nslookup	用于排查DNS解析的问题
防火墙和NAT	iptables	用于排查防火墙及NAT的问题
网卡选项	ethtool	用于查看和配置网络接口的功能选项
网络抓包	tcpdump、Wireshark	通常在服务器中使用tcpdump抓包后再复制出来用Wireshark的图形界面分析
动态追踪	ftrace bcc、systemtap	ftrace用于跟踪内核函数调用栈，而bcc和systemtap则用于跟踪内核或应用程序的执行过程（注意bcc要求内核版本>=4.1）

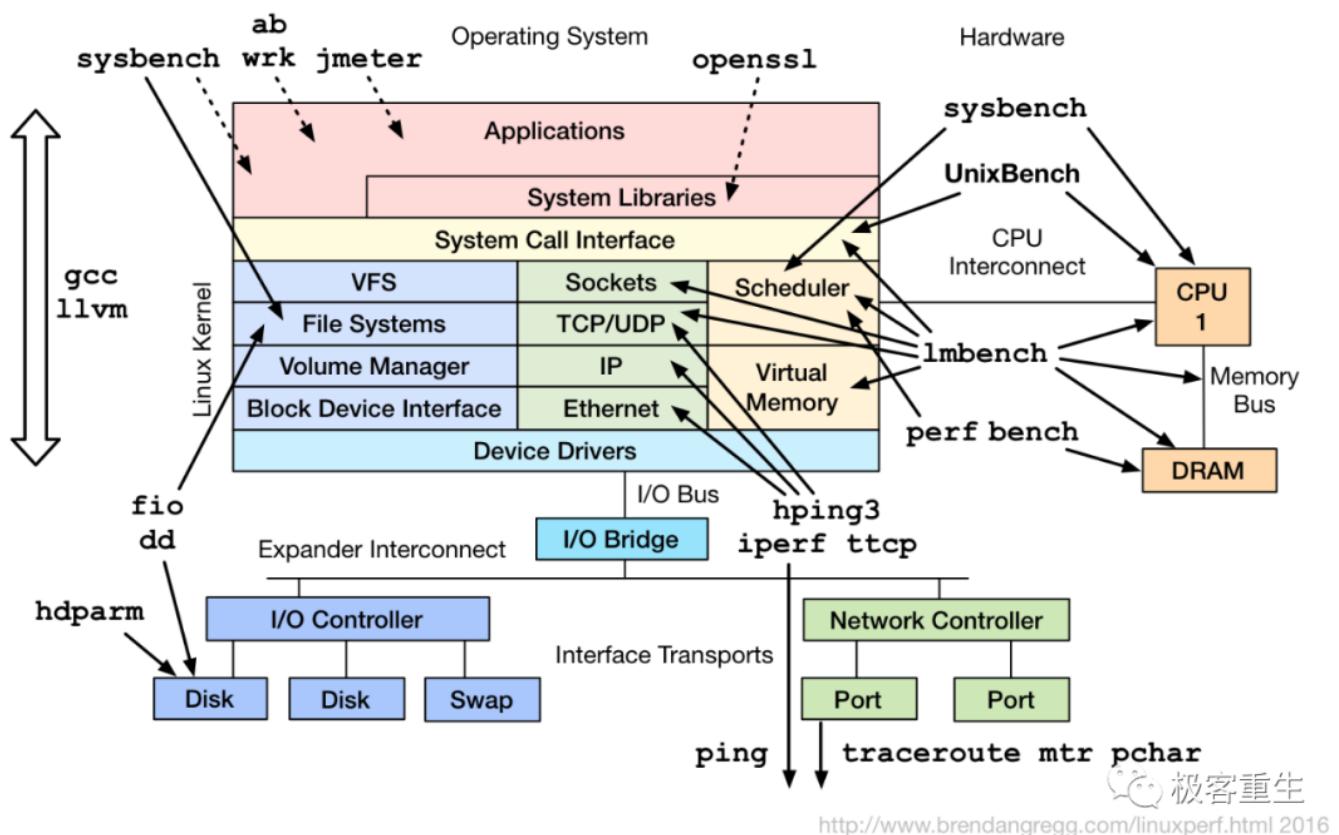
 极客重生

# 基准测试工具

除了性能分析外，很多时候，我们还需要对系统性能进行基准测试。比如，

- 在文件系统和磁盘 I/O 模块中，我们使用 fio 工具，测试了磁盘 I/O 的性能。
- 在网络模块中，我们使用 iperf、pktgen 等，测试了网络的性能。
- 而在很多基于 Nginx 的案例中，我们则使用 ab、wrk 等，测试 Nginx 应用的性能。

Linux Performance Benchmark Tools



## 参考

- 相当一部分内容来自极客时间出品的倪鹏飞专栏《Linux性能优化》，这是之前这个专栏的学习笔记。
- 另一份资料是IBM红宝书Linux性能调优指南。
- 此外，The Linux Documentation Project是一个非常好的资料库。
- 将硬件中断的处理任务分配给多个CPU：SMP affinity and proper interrupt handling in Linux
- Hidden Costs of Memory Allocation

- <https://www.lijiaocn.com/soft/linux/>

- END -

看完一键三连**在看，转发，点赞**

是对文章最大的赞赏，极客重生感谢你❤

#### 推荐阅读

如何分析常见的TCP问题？

网络排障全景指南手册v1.0精简版pdf

云网络丢包故障定位全景指南

硬核分析|腾讯云原生OS内存回收导致关键业务抖动问题

一个奇葩的网络问题，把技术砖家"搞蒙了"



收录于话题 #深入理解Linux系统 29

< 上一篇

突破各个子系统，你就能对Linux了如指掌

下一篇 >

Linux调度系统全景指南(中篇)

喜欢此内容的人还喜欢

算法面试 | 论如何4个月高效刷满 500 题并形成长期记忆

极客重生

