

Linux服务器构建与运维管理

第12章：系统监控

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

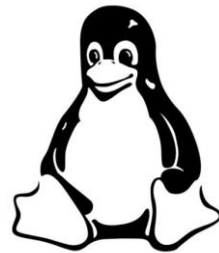
<https://internet.hactcm.edu.cn>
<http://www.51xueweb.cn>

河南中医药大学信息管理与信息系统教研室
河南中医药大学信息技术学院互联网技术教学团队
河南中医药大学医疗健康信息工程技术研究所

2022.10

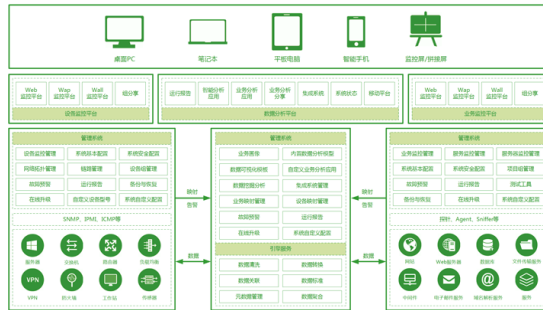
提纲

- 系统管理与系统监控
- 查看系统的性能状态
 - 内存与缓存监控
 - CPU监控
 - 磁盘与IO监控
 - 进程监控与管理
- 实时监控系统的运行状态
 - top htop atop
 - dstat
 - PROC虚拟文件系统
- 使用Linux-dash实现可视化监控
- 实现网络与系统运维监控系统



1.系统管理与系统监控

- 系统管理是对系统当前运行状态进行控制，使之与预期目标一致，同时要结合外界环境，综合操作系统以往运行特征进行分析，实现对操作系统未来发展趋势的预测。



1.系统管理与系统监控

- 系统管理员日常操作内容
 - 权限管理
 - 负责为新用户增设账号、将不再活动的用户删除，处理账号相关事务。
 - 当某个用户不应该再访问系统时，必须禁用该用户的账号，该账号拥有的所有文件必须备份后给予删除，以使系统不会随着时间的增长而积累无用信息。
 - 磁盘管理
 - 配置系统能够识别新磁盘或磁盘阵列，使用新存储资源。
 - 文件管理
 - 维护文件系统内容，保证系统文件内容清晰化，方便其他账号访问文件。
 - 内存管理
 - 需要时刻监视系统内关键业务的内存使用情况，合理调配资源为业务提供保障。
 - 进程管理
 - 监控并处理系统中的无用进程，降低系统负载压力。
 - 日志管理
 - 合理记录系统日志，便于操作追溯和日志审查分析。

1.系统管理与系统监控

- 系统管理的方式可分为命令化管理和自动化管理。
 - 命令化管理是通过操作系统的相关命令实现系统配置
 - vi 对文件进行编辑管理
 - fdisk 对磁盘进行管理
 - nmcli 对网络进行管理
 - systemctl 对服务进行管理
 - 自动化管理是通过自动化运维工具实现对大量主机的配置管理，对系统的网络、存储、应用交付等进行自动化配置，降低运维管理人员的压力，消减重复性工作。



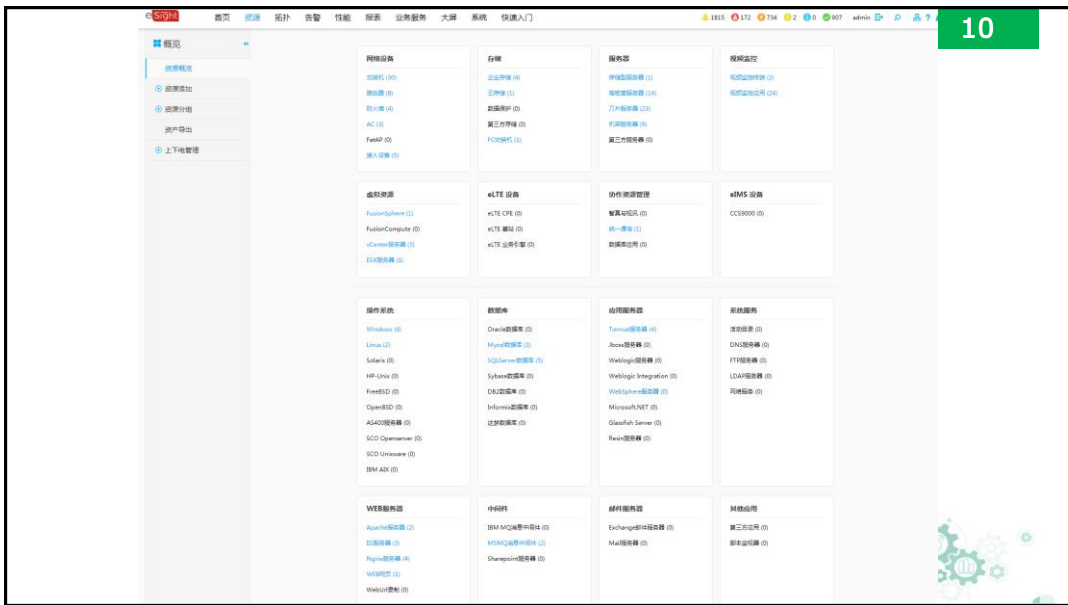
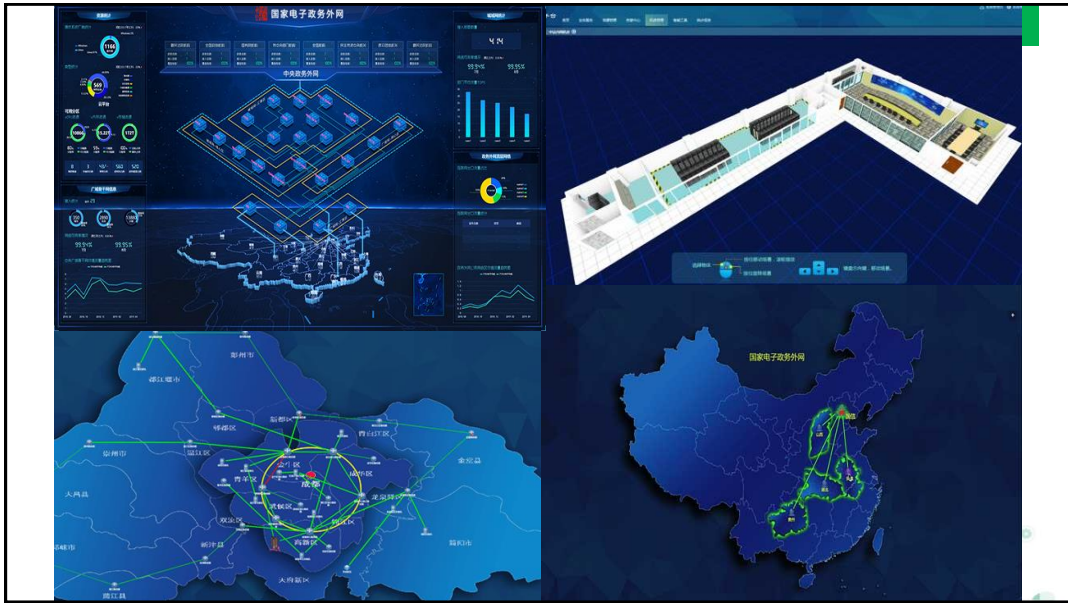
1.系统管理与系统监控

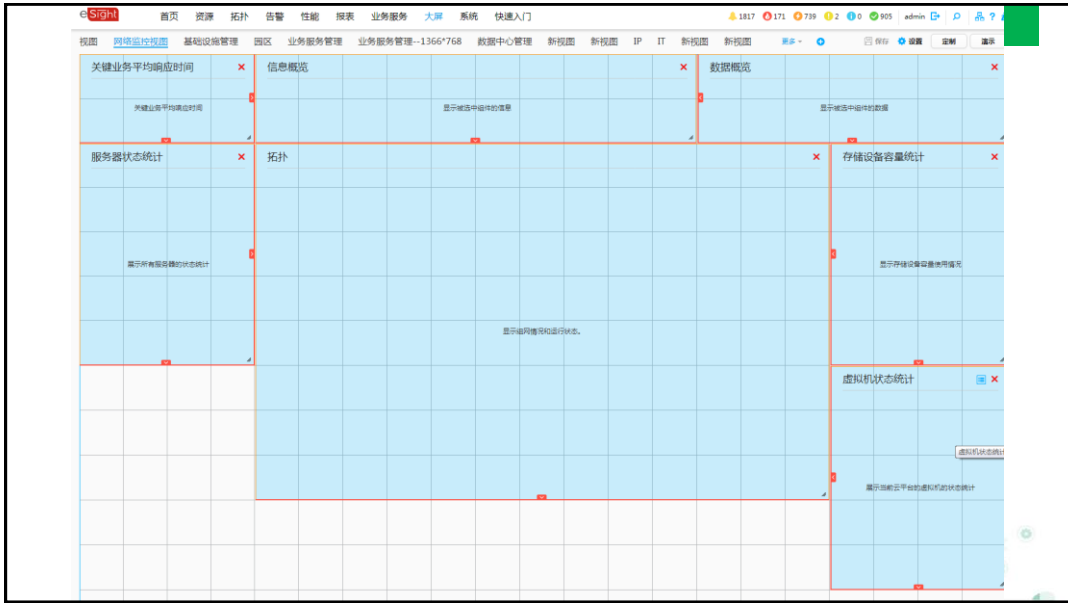
- 为什么要监控系统
 - 随着信息化建设的不断深入，应用系统不断增多，运维人员需要管理的设备、业务数量急剧增加，如何直观地查看多个设备、业务的运行情况，并保证出现异常时能及时发现，已成为运维人员最关心也最需解决的问题。
 - 通过系统监控可以实时了解系统的运行状态，快速发现系统异常，及时解决异常问题，保障系统服务的可靠性和稳定性。
- 系统监控的内容
 - 系统监控是对操作系统整体运行情况的监控，通常监控系统的CPU、负载、物理内存、虚拟内存、内核线程、磁盘、进程等方面。



网络与信息系统运维监控平台功能赏析

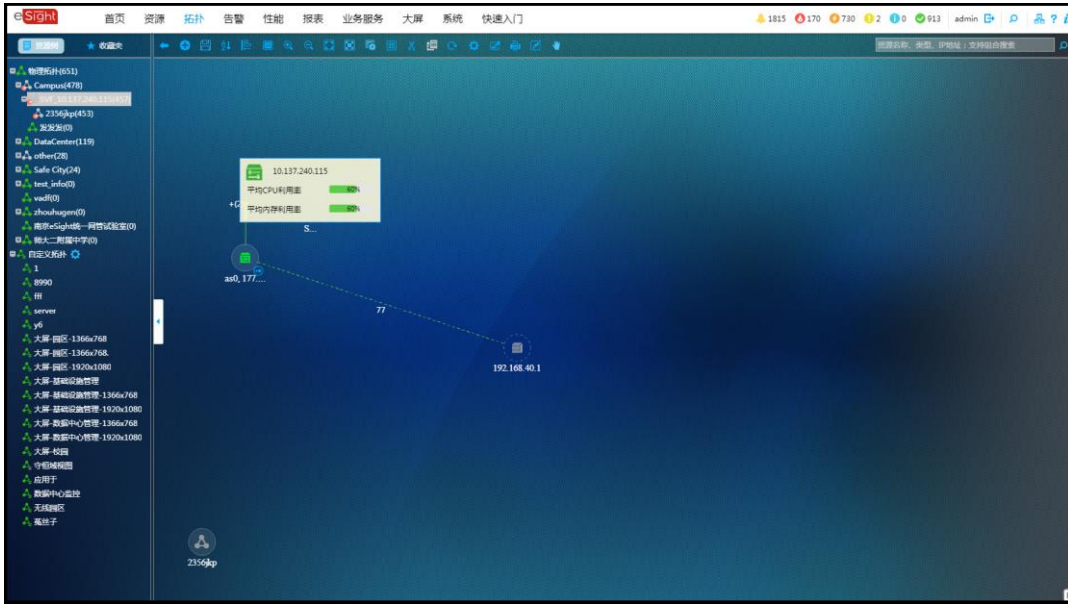


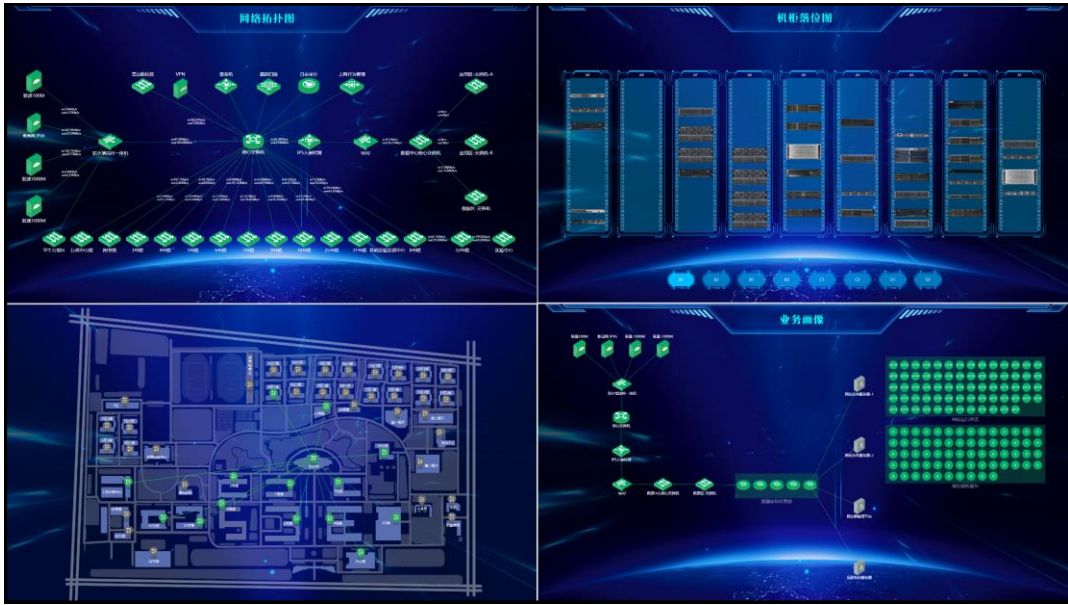


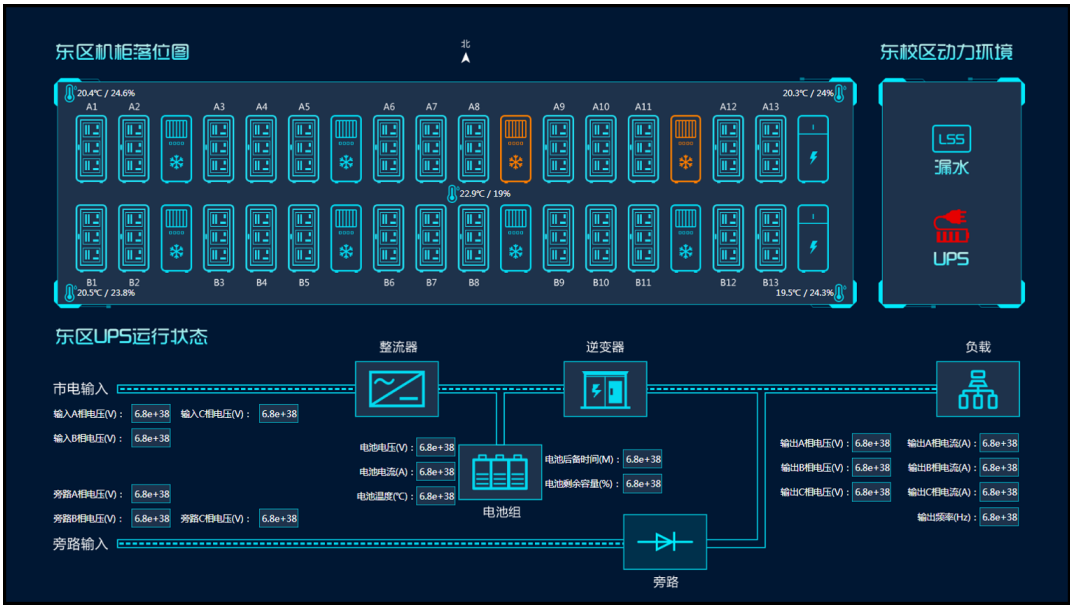


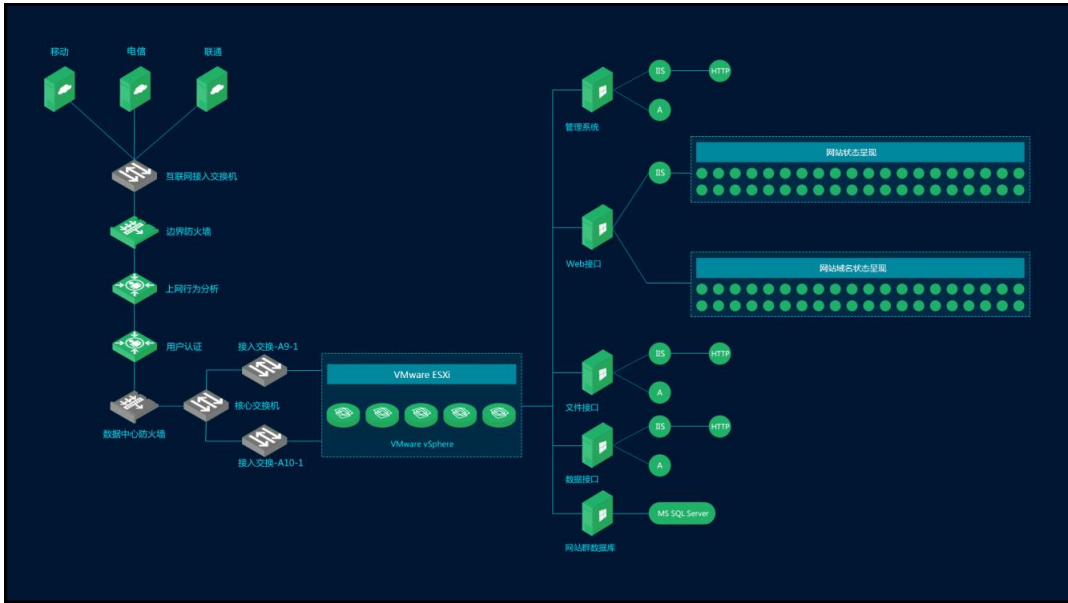
The screenshot displays a detailed table of server status and performance metrics. The table includes columns for various system parameters and their values.

操作日志	告警	操作	告警内容	到达时间	确认	清除	备注	第三方工单	报警规则	报警策略	次数	所属设备	报警时间	报警类型
→	已清除	Other	端口业务+web上端口 E=1	2017-12-12 08:57:19	未确认	清除	备注=0113772	未配置	端口业务	报警	1	业务主机	2017-12-12 08:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=2	2017-12-12 08:57:19	未确认	清除	备注=0113772	未配置	端口业务	报警	1	业务主机	2017-12-12 08:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=3	2017-12-12 10:57:49	未确认	清除	备注=0113787	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:49	业务主机
→	已清除	Other	端口业务+web上端口 E=4	2017-12-12 10:57:20	未确认	清除	备注=0113787	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=5	2017-12-12 10:57:20	未确认	清除	备注=0113774	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=6	2017-12-12 10:57:19	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=7	2017-12-12 10:57:19	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=8	2017-12-12 10:57:19	未确认	清除	备注=0113774	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=9	2017-12-12 10:57:19	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:19	业务主机
→	已清除	Other	端口业务+web上端口 E=10	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=11	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=12	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=13	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=14	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=15	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=16	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=17	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=18	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=19	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=20	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=21	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=22	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=23	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=24	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=25	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=26	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=27	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=28	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=29	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机
→	已清除	Other	端口业务+web上端口 E=30	2017-12-12 10:57:20	未确认	清除	备注=0113783	未配置	端口业务	报警	1	业务主机	2017-12-12 10:57:20	业务主机













2.查看系统的性能状态

2.1 内存与缓存监控

- 物理内存是CPU能直接寻址的存储空间，是系统硬件提供的内存：
 - 物理内存的存取速度快
 - 暂时存放CPU的运算数据
 - 存储硬盘等外部存储器交换的数据
 - 保障CPU计算的稳定性和高性能



2.查看系统的性能状态

- 虚拟内存是为了满足物理内存的不足而提出的策略，利用磁盘空间虚拟出一块逻辑内存。
- 用作虚拟内存的磁盘空间被称为交换空间（Swap Space）。
 - 获取更多的内存空间，且空间地址是连续的，使程序编写连接更加简单。
 - 不同进程的虚拟地址之间没有关系，单个进程操作不会对其他进程造成影响。
 - 每块虚拟内存都有相应的读写属性，保护程序的代码段不被修改，数据块不能被执行等，增加了系统的安全性。
 - 可直接映射磁盘上的文件到虚拟地址空间，从而做到物理内存长时间分配，只需要在读取相应文件的时候，才从虚拟内存加载到物理内存中。
 - 进程间的内存共享可以通过映射同一块物理内存到进程的不同虚拟内存空间来实现共享。
 - 使用虚拟内存后，可方便使用交换空间和COW（copy on write）等功能。



2.查看系统的性能状态

- 在Linux操作系统中，以应用程序读写文件数据为操作过程，系统内存执行过程如下。
 - 操作系统分配内存，将读取的数据从磁盘读入到内存中
 - 从内存中将数据分发给应用程序
 - 向文件中写数据时，操作系统分配内存接收用户数据
 - 接收完成后，内存将数据写入磁盘
- 如果有大量数据需要从磁盘读取到内存或者由内存写入磁盘时，系统的读写性能就变得非常低，因为无论是从磁盘读数据，还是写数据到磁盘，都是一个很消耗时间和资源的过程。



2.查看系统的性能状态

2.1 内存与缓存监控

表 12-1-2 物理内存常见指标及其含义

指标	说明
MMU	内存管理单元，是 CPU 用来将进程的虚拟内存转换为物理内存的模块。它的输入是进程的页表和虚拟内存，输出是物理内存。将虚拟内存转换成物理内存的速度直接影响着系统的速度，所有 CPU 均包含该硬件模块用于系统加速
TLB	查找缓存区，存在 CPU L1 cache 中，用于查找虚拟内存和物理内存的映射信息
Buffer Cache	缓冲区缓存，用来缓冲设备上的数据，当读写磁盘时，系统会将相应的数据存放到 Buffer Cache，等下次访问时，直接从缓存中拿数据，从而提高系统效率
Page Cache	页面缓存，用来加快读写磁盘上文件的速度，数据结构是文件 ID 和 offset 到文件内容的映射，根据文件 ID 和 offset 就能找到相应的数据



2.查看系统的性能状态

2.1 内存与缓存监控

□ free

- 查看当前主机操作系统的物理内存总量、使用量及剩余量等。

```
[root@Project-11-Task-01 ~]# free -h
Mem:  521M  200M  277M  11M  343M  464M
Swap:  819M  0B  819M
```

```
total      物理内存总大小
used      已使用内存大小，包括缓存和应用程序实际使用的内存大小
free      剩余未被使用的内存大小
shared    共享内存大小，进程间通信使用
buffers  被缓冲区占用的内存大小
cached   被缓存占用的内存大小
available  可被应用程序使用的内存大小
```



2.查看系统的性能状态

2.1 内存与缓存监控

□ free

- 查看当前主机操作系统的物理内存总量、使用量及剩余量等。

【语法】

free [选项]

【选项】

-b 以Byte为单位显示内存使用情况
 -k 以KB为单位显示内存使用情况
 -m 以MB为单位显示内存使用情况
 -o 不显示缓冲区调节列
 -s <间隔秒数> 持续观察内存使用状况，按照指定时间刷新数据
 -t 显示内存总和列



2.查看系统的性能状态

2.1 内存与缓存监控

□ vmstat

- 统计系统整体情况，包括内核进程、虚拟内存、磁盘和CPU活动信息。
- vmstat命令可查看procs（进程）、memory（内存）、swap（交换分区）、io（IO读写）、system（系统）以及cpu的运行信息。

```
[root@Project-11-Task-01 ~]# vmstat
procs-----memory-----swap-----io-----system-----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  st
2  0  0  284904  3268  348264  0  0  43  14  132  222  0  1  99  0  0
[root@Project-11-Task-01 ~]#
```

procs运行结果中选项内容

r 运行队列中进程的数量
 b 等待IO的进程数量

memory运行结果中选项内容

swpd 虚拟内存使用量
 free 空闲物理内存量
 buff 用于缓冲的内存量
 cache 用于缓存的内存量

swap运行结果选项内容

si 每秒从交换分区写入内存数据量大小
 so 每秒写入交换分区数据量大小

io运行结果选项内容

bi 每秒读取的磁盘块数
 bo 每秒写入的磁盘块数

system运行结果选项内容

in 每秒系统中断数
 cs 每秒上下文切换数

cpu运行结果选项内容

us 用户进程执行时间百分比
 sy 内核系统进程执行时间百分比
 wa IO等待时间（百分比）
 id CPU空闲时间（百分比）
 st Time stolen from a virtual machine



2.查看系统的性能状态

2.1 内存与缓存监控

□ vmstat

- 统计系统整体情况，包括内核进程、虚拟内存、磁盘和CPU活动信息。

【语法】

vmstat [选项] [参数]

【选项】

-a 显示活动和非活动内存
 -f 显示启动后创建的进程总数
 -m 显示slab（内存分配机制）信息
 -n 只在开始时显示一次各字段头信息
 -s 以表格方式显示事件计数器和内存状态
 -d 显示磁盘相关统计信息
 -p 显示指定磁盘分区统计信息
 -S 使用指定单位显示，可使用k、K、m、M

【参数】

时间间隔 状态信息刷新的时间间隔
 次数 显示报告的次数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.1 内存与缓存监控

□ swapon

- 查看当前主机操作系统的交换分区运行情况。

```
[root@Project-11-Task-01 ~]# swapon
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition 8396 0    -2
[root@Project-11-Task-01 ~]#
[root@Project-11-Task-01 ~]# swapon -s
文件名      类型      大小      已用      程度
/dev/dm-1   partition 839676 0         -2
[root@Project-11-Task-01 ~]#
```

Filename 交换分区对应的设备文件名称
 Type 文件类型，Partition表示为分区
 Size 交换分区大小
 Used 交换分区目前使用量
 Priority 交换分区使用的优先级

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.1 内存与缓存监控

□ swapon

- 查看当前主机操作系统的交换分区运行情况。

【语法】

swapon [选项] [参数]

【选项】

-a 将/etc/fstab文件中所有设置为swap的设备，启动为交换分区
 -p <优先顺序> 指定交换分区的使用优先级顺序
 -s 显示交换分区的使用状况

【参数】

交换空间

指定需要激活的交换文件或交换分区。
 如果是交换分区则指定交换分区对应的设备文件。

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.1 内存与缓存监控

□ 案例：使用命令记录系统内存运行情况

- 通过vmstat命令动态监控系统内存，实现2个目标。
 - 每秒检测一次内存运行情况，持续记录5分钟
 - 输出检测结果至用户主目录的~/memory.txt文件中

```
[root@Project-11-Task-01 ~]# vmstat 1 300 >> memory.txt
^C
[root@Project-11-Task-01 ~]# tail -n 20 memory.txt
 3 0 11064 63888 4372 485968 20 0 20 0 1108 4380 33 67 0 0 0
 2 0 11064 63888 4372 485972 0 0 0 0 1095 4413 33 67 0 0 0
 2 0 11064 65080 4372 485972 0 0 0 0 1050 4331 29 71 0 0 0
 1 1 12032 179072 4384 376240 0 900 340 20132 822 1943 13 32 1 54 0
 0 1 12032 172980 4384 381444 0 0 5780 757 1809 1527 9 11 13 60 0
 0 0 12032 166908 4384 382728 0 0 1508 293 538 804 9 0 60 21 0
 1 2 12032 161192 4384 389356 0 0 5536 341 1567 2716 11 23 19 46 0
 4 1 12032 151544 4384 399500 28 0 9952 0 1467 1055 24 26 0 51 0
 0 2 12032 131888 4384 418352 36 0 10000 9036 1163 1327 39 12 0 40 0
procs-----memory-----swap-----io-----system-----cpu-----
 r b swpd free buff cache si so bi bo in cs us sy id wa st
 2 1 12032 122200 4384 424496 0 0 6276 905 1468 2093 35 17 0 48 0
 1 1 12032 104800 4392 426852 0 0 3088 1271 1751 4088 16 29 13 43 0
 0 1 12032 192520 4392 429200 0 0 876 1849 1362 2855 3 12 43 42 0
 0 0 12032 198380 4392 427712 0 0 104 235 600 798 2 3 91 4 0
 0 0 12032 198380 4392 427712 0 0 0 0 346 291 0 1 99 0 0
 0 0 12032 198532 4392 427632 0 0 0 0 124 191 0 1 99 0 0
 0 0 12032 198552 4392 427600 0 0 0 0 137 208 0 0 100 0 0
 0 0 12032 198552 4392 427600 0 0 0 0 103 178 0 0 100 0 0
 0 0 12032 198552 4392 427600 0 0 0 0 109 199 0 1 99 0 0
[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

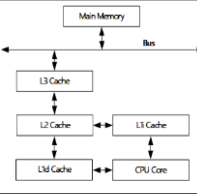
2.2 CPU监控

表 12-2-1 CPU 基本概念及其含义

概念	说明
物理 CPU	主板上实际接入的 CPU 个数, 在 Linux 中用 “physical id” 确定
CPU 核数	每个物理 CPU 上实际接入的芯片组数量, 如双核、四核等
逻辑 CPU	一般情况下, 逻辑 CPU 数 = 物理 CPU 数量 * CPU 核数, 如果逻辑 CPU 多于物理 CPU, 说明该 CPU 支持超线程技术

表 12-2-2 CPU 缓存指标及其含义

指标	说明
Main Memory	物理运行内存信息
Bus	Linux 系统总线
L3 Cache	CPU 三级缓存
L2 Cache	CPU 二级缓存
L1l Cache	CPU 一级缓存, 用于存储指令
L1d Cache	CPU 一级缓存, 用于存储数据
CPU Core	CPU 内核



河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

2.2 CPU监控

□ lscpu

- 查看CPU架构、数量、型号、主频等详细信息。

```

[root@Project-11-Task-01 ~]# lscpu
#CPU架构          x86_64
#CPU指令集        x86_64
#CPU多字节存储顺序 Little Endian
#CPU核心数        1
#当前在线CPU数量  0
#每个核心的线程数 1
#每个物理CPU核心数 1
#主板上CPU插槽数  1
#NUMA节点数      1
#CPU厂商          GenuineIntel
#CPU系列          6
#CPU型号标识      94
#CPU型号名称      Intel(R) Core(TM) i5-6600T CPU @ 2.70GHz
#CPU更新版本      3
#CPU主频          CPU MHz: 2712.004
#在系统内核启动时检测到的物理CPU逻辑CPU连接
#HyperThreading技术
#CPU支持的虚拟化技术
#CPU一级数据缓存大小
#CPU一级指令缓存大小
#CPU二级缓存大小
#CPU三级缓存大小
#NUMA的节点数    0
#当前CPU支持的功能
[root@Project-11-Task-01 ~]#
  
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

2.2 CPU监控

□ lscpu

- 查看CPU架构、数量、型号、主频等详细信息。

【语法】

lscpu [选项]

【选项】

-e 以扩展可读的格式显示
-p 以可解析的格式显示



2.查看系统的性能状态

2.2 CPU监控

□ mpstat

安装方法: yum install sysstat

- 实时监控主机系统的CPU, 了解系统的运行状态。

```
[root@Project-11-Task-01 ~]# mpstat
Linux 4.18.0-147.5.1.el8_1.x86_64 (Project-11-Task-01) 2020年04月19日  _x86_64_ (1 CPU)
14# 2675 05# CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %gnice   %idle
14# 2675 05# all      0.53    0.14    0.17    0.32    0.44    0.11    0.00    0.00    0.00  98.30
[root@Project-11-Task-01 ~]#
```

usr 用户操作占用CPU的时间百分比
nice 进程占用CPU的时间百分比
sys 系统内核处理占用CPU的时间百分比
iowait 磁盘IO等待的时间百分比
irq CPU硬中断的时间百分比
soft CPU软中断的时间百分比
steal 虚拟CPU处在非自愿等待下占用的时间百分比
quest 运行虚拟处理器时CPU的时间百分比
gnice 低优先级进程占用CPU的时间百分比
idle 除磁盘IO等待外, CPU空闲的时间百分比



2.查看系统的性能状态

□ mpstat

- 实时监控主机系统的CPU，了解系统的运行状态。

【语法】

mpstat [选项] [参数]

【选项】

-P 指定CPU核心编号

【参数】

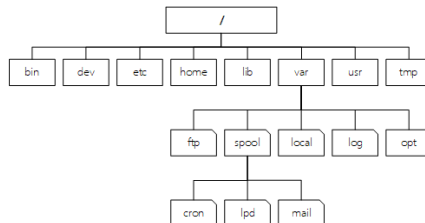
时间间隔 指定监控报告执行输出的时间间隔（秒）
次数 显示系统CPU检测的执行次数



2.查看系统的性能状态

□ 文件系统概述

- 文件和目录的操作命令、存储、组织和控制的总体结构统称为文件系统。文件系统是指格式化后用于存储文件的设备（如硬盘分区、光盘、软盘、闪存及其他存储设备）。
- 文件系统会对存储空间进行组织和分配，并对文件的访问进行保护和控制。不同的操作系统对文件的组织方式会有所区别，其所支持的文件系统类型也不一样。
- 文件系统的组织方式是树状的层次式目录结构，在这个结构中处于最顶层的是根目录，用“/”代表，往下延伸就是其各级子目录。



2.查看系统的性能状态

□ 文件系统类型

- Linux操作系统支持的文件系统类型很多，除了UNIX所能支持的常见文件系统类型外，还支持包括FAT16、FAT32、NTFS在内的各种Windows文件系统。
- Linux可通过“加载”的方式把Windows操作系统的分区挂载到Linux的某个目录下进行访问。



表 12-4-1 Linux 支持的文件系统类型

文件系统	说明
ext	第一个专门针对 Linux 的文件系统，为 Linux 的发展做出了重要贡献，但由于性能和兼容性上存在许多缺陷，现在已很少使用
ext2	为解决 ext 文件系统的缺陷而设计的高性能、可扩展的文件系统，在 1993 年发布，其特点是存取文件的性能好，在中小型的文件方面的优势尤其明显
ext3	日志文件系统，是 ext2 的升级版，用户可以方便地从 ext2 文件系统迁移到 ext3 文件系统，ext3 在 ext2 的基础上加入了日志功能，即使系统因为故障导致宕机，ext3 文件系统也只需要数十秒中即可恢复，避免了意外宕机对数据的破坏
ext4	ext4 是 ext3 的改进版，修改了 ext3 中部分重要的数据结构，也提供了更佳的性能和可靠性，还有更为丰富的功能
zfs	动态文件系统（Dynamic File System），是第一个 128 位文件系统，最初是由 Sun 公司为 Solaris 10 操作系统开发的文件系统，Linux 发行版正在逐步默认使用该文件系统
swap	Linux 中一个专门用于交换分区的文件系统（类似与 Windows 上的虚拟内存）
NFS	网络文件系统，可支持不同的操作系统，实现不同系统间的文件共享，其通信协议设计与主机及操作系统无关
smb	SMB 协议的网络文件系统，可用于实现 Linux 和 Windows 操作系统之间的文件共享
cifs	通用网络文件系统，是 SMB 协议的网络文件系统的增强版本，是计算机用户在企业内部和因特网上共享文件的标准方法
vfat	与 Windows 系统兼容的 Linux 文件系统，可作为 Windows 分区的交换文件
minix	Minix 操作系统使用的文件系统，也是 Linux 最初使用的文件系统之一



2. 查看系统的性能状态

□ 文件系统常见目录

- Linux操作系统在安装过程中会创建一些默认的目录，这些默认目录是有特殊功能的。
- 用户在不确定的情况下最好不要更改这些目录下的文件，以免造成系统错误。



表 12-4-2 常见的 Linux 系统中的默认目录及说明

目录	说明
/	Linux 文件系统的入口，也是整个文件系统的最顶层目录
/bin	存放可执行的命令文件，供系统管理员和普通用户使用，例如 cp、mv、rm、cat 和 ls 等。此外，该目录还包含诸如 bash、csh 等 Shell 程序
/boot	存放内核影像及引导系统所需要的文件，比如 vmlinuz、initrd.img 等内核文件以及 GRUB 等系统引导管理程序
/dev	存放设备文件，Linux 中每个设备都有对应的设备文件
/etc	存放系统配置文件
/etc/init.d	存放系统中以 System V Init 模式启动的程序脚本
/etc/xinit.d	存放系统中以 xinetd 模式启动的程序脚本
/etc/rc.d	存放系统中不同运行级别的启动和关闭脚本
/home	存放普通用户的个人主目录
/lib	存放库文件
/lost+found	存放因系统意外崩溃或机器意外关机而产生的文件碎片，当系统启动的过程中 fsck 工具会检查这个目录，并修复受损的文件系统
/media	存放即插即用型存储设备自动创建的挂载点
/mnt	存放存储设备的挂载目录
/opt	存放较大的第三方软件



表 12-4-2 常见的 Linux 系统中的默认目录及说明

目录	说明
/proc	该目录并不存在磁盘上，而是一个实时的、驻留在内存中的文件系统，用于存放操作系统、运行进程以及内核等信息
/root	root 用户默认主目录
/sbin	存放大多数涉及系统管理的命令，这些命令只有 root 用户才有权限执行
/tmp	临时文件目录，用户运行程序时所产生的临时文件就存放在这个目录下
/usr	存放用户自行编译安装的软件及数据，也存放字体、帮助文件等
/usr/bin	存放普通用户有权限执行的可执行程序，以及安装系统时自动安装的可执行文件
/usr/sbin	存放可执行程序，但大多是系统管理的命令，只有 root 权限才能执行
/usr/local	存放用户自行编译安装的软件
/usr/share	存放系统共用的文件，如字体文件、帮助文件等
/usr/src	存放内核源码
/var	存放系统运行时要改变的数据
/var/log	存放系统日志
/var/spool	存放打印机、邮件等脱机文件



2. 查看系统的性能状态

2.3 磁盘与IO监控

□ tree

安装方法: yum install tree

- 以树状图形样式列出目录内容，可查看文件系统中所有目录、文件等信息。

```
[root@Project-11-Task-01 ~]# tree /etc/ -L 2
/etc/
├── adjtime
├── aliases
├── alternatives
│   ├── cifs-idmap-plugin -> /usr/lib64/cifs-utils/cifs_idmap_sss.so
│   ├── ifdown -> /usr/libexec/nm-ifdown
│   ├── ifup -> /usr/libexec/nm-ifup
│   ├── libnssckbi-no_nss_db -> /usr/lib64/pkcs11/p11-kit-trust.so
│   ├── nssm -> /usr/share/man/man1/nssm.1.gz
│   ├── nmap -> /usr/bin/nmap
│   ├── python -> /usr/libexec/no-python
│   └── unversioned-python-man -> /usr/share/man/man1/unversioned-python.1.gz
├── anacrontab
├── audit
│   ├── auditd.conf
│   ├── audit.rules
│   ├── audit-stop.rules
│   ├── plugins.d
│   └── rules.d
├── authselect
│   ├── custom
│   └── user-nsswitch.conf
├── bash_completion.d
├── authselect-completion.sh
├── ifconfig
├── basdir
├── bindresvport.blacklist
├── binfmt.d
├── centos-release
└── centos-release-upstream
```



2.查看系统的性能状态

2.3 磁盘与IO监控

□ tree

- 以树状图形样式列出目录内容，可查看文件系统中所有目录、文件等信息。

【语法】

```
tree [选项] [参数]
```

【选项】

```
-a    显示所有文件和目录
-d    输出目录名称
-f    在每个文件或目录之前，显示完整的相对路径名称
-g    列出文件或目录的所属组名称
-i    不以阶梯状列出文件和目录名称
-l    不显示符号样本样式的文件或目录名称
-l    针对映射连接的目录，直接列出该连接所指向的原始目录
-n    不在文件和目录清单加上颜色
.....
```

【参数】

```
目录  列出该指定目录下的所有文件，包括子目录里的文件
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.3 磁盘与IO监控

□ df

- 查看主机文件系统磁盘的使用情况。

```
[root@Project-11-Task-01 ~]# df -h
文件系统      容量  已用  可用  已用% 挂载点
devtmpfs      392M  0  392M   0% /dev
tmpfs         413M  0  413M   0% /dev/shm
tmpfs         413M  16M  397M   4% /run
tmpfs         413M  0  413M   0% /sys/fs/cgroup
/dev/mapper/cl-root 6.2G  1.7G  4.6G  28% /
/dev/sda1     976M  228M  648M  23% /boot
tmpfs         63M  0  63M   0% /run/user/0
[root@Project-11-Task-01 ~]#
```

Filesystem	主机文件系统名称
1K-blocks	1K大小文件块数量
Used	已使用的磁盘大小
Available	可用磁盘大小
Use%	已用磁盘大小百分比
Mounted on	文件系统挂载点

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

□ df

- 查看主机文件系统磁盘的使用情况。

【语法】

df [选项] [参数]

【选项】

-a 显示全部文件系统列表
 -h 以合适的单位来显示，提高可读性
 -H 等于“-h”，但是计算时1K=1000，而不是1K=1024
 -i 用索引节点信息替代磁盘信息
 -k 指定区块的大小
 -l 只显示本地文件系统
 -m 指定区块大小

 【参数】

文件系统 指定文件系统查看信息



2.查看系统的性能状态

□ iostat

安装方法: yum install sysstat

- 监视主机磁盘IO活动情况，查看存储设备的性能，也输出当前CPU的使用情况。

```
[root@Project-11-Task-01 ~]# iostat
Linux 4.18.0-147.5.1.el8_1.x86_64 (Project-11-Task-01) 2020年04月19日 _x86_64_ (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.89    0.08    0.71    0.46    0.00   97.87

Device            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
sda                 3.17         65.81         58.75     665355     593968
dm-9                 3.56         69.63         53.68     632966     541862
dm-1                 0.33          0.38          1.20         3824     12116

[root@Project-11-Task-01 ~]#
```

Device 检测磁盘设备名称
 tps 设备每秒的传输次数
 kB_read/s 每秒从设备读取的数据量
 kB_wrtn/s 每秒向设备写入的数据量
 kB_read 从设备读取的总数据量
 kB_wrtn 从设备写入的总数据量



2.查看系统的性能状态

□ iostat

安装方法: yum install sysstat

- 每隔2秒检测磁盘IO情况, 设置单位为MB, 并显示最近4秒内的2次数据信息。

```
[root@Project-11-Task-01 ~]# iostat -m 2 2
Linux 4.18.0-147.5.1.el8_1.x86_64 (Project-11-Task-01) 2020年04月19日 _x86_64_ (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.91    0.07    0.71    0.46    0.00   97.84

Device:            tps    MB_read/s    MB_wrtn/s    MB_read    MB_wrtn
sda                  3.12         0.06         0.06      652      584
dm-0                  3.21         0.06         0.05      601      533
dm-1                   0.32         0.00         0.00         3         11

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           11.11    0.00    4.55    0.00    0.00   84.34

Device:            tps    MB_read/s    MB_wrtn/s    MB_read    MB_wrtn
sda                  0.00         0.00         0.00         0         0
dm-0                  0.00         0.00         0.00         0         0
dm-1                  0.00         0.00         0.00         0         0

[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

□ iostat

- 每隔2秒检测磁盘IO情况, 设置单位为MB, 并显示最近4秒内的2次数据信息。

【语法】

iostat [选项] [参数]

【选项】

-c 仅显示CPU使用情况
 -d 仅显示磁盘设备IO情况
 -k 显示状态以千字节每秒为单位, 而不使用块每秒
 -m 显示状态以兆字节每秒为单位
 -p 仅显示块设备和所有被使用的其他分区状态
 -t 显示每个报告产生时的时间
 -x 显示扩展状态信息

【参数】

时间间隔 每次报告产生的间隔时间 (秒)
 次数 显示报告的次数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.3 磁盘与IO监控

□ iotop

- 监控磁盘IO使用状况，可对进程、用户、IO等相关信息。

```

Total DISK READ : 0.00 B/s | Total DISK WRITE : 7.69 K/s
Actual DISK READ: 0.00 B/s | Actual DISK WRITE: 0.00 B/s
  PID USER   DISK B/s  DISK %IOPS  SW/PR  IO  COMMAND
19183 be/4 root    0.00 B/s   0.00 %  0.00 % 0.03 % [kworker/0:1-events_power_efficient]
1467  be/4 root    0.00 B/s   3.84 K/s   0.00 %  0.00 % rsyslogd -n [in:imjournal]
1471  be/4 root    0.00 B/s   3.84 K/s   0.00 %  0.00 % rsyslogd -n [rsain:imQlog]
  1  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % systemd --system --deserialize 20
  2  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kthread]
  3  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [rcu_gp]
  4  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [rcu_par_gp]
  6  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kworker/0:0H-kblockd]
  8  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [sm_percpu_wq1]
  9  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [ksoftirqd/0]
10  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [rcu_sched]
11  rt/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [rtirq/0]
12  rt/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [watchdog/0]
13  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [cpuhp/0]
15  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [bdsvemrfs]
16  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [netns]
17  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kauditd]
18  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [khugepaged]
19  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [oom_reaper]
20  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [writeback]
21  be/4 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kcompactd]
22  be/5 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [ksmd]
23  be/7 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [khugepaged]
24  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [cgroup]
25  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kintegrityd]
26  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [kblockd]
27  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [tmp_dev_wq]
28  be/0 root    0.00 B/s   0.00 B/s   0.00 %  0.00 % [md]
  
```

安装方法: yum install iotop

iotop命令执行过程中常用的快捷键

左右箭头 改变排序方式，默认是按IO大小排序
r 改变排序顺序
o 只显示有IO输出的进程
p 进程/线程显示方式的切换
a 显示累计使用量
q 退出

主机磁盘IO总览结果中选项内容

Total DISK READ 每秒磁盘总读取大小
Total DISK WRITE 每秒磁盘总写入大小
Actual DISK READ 实际每秒磁盘读取大小
Actual DISK WRITE 实际每秒磁盘写入大小

主机磁盘IO进程结果中选项内容

TID 线程ID
PRIO 线程优先级
USER 所属用户
DISK READ 每秒中磁盘读取大小
DISK WRITE 每秒中磁盘写入大小
SWAPIN 写入交换分区占比
IO 使用率大小
COMMAND 线程执行命令

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

2.3 磁盘与IO监控

□ iotop

- 监控磁盘IO使用状况，可对进程、用户、IO等相关信息。

【语法】

iotop [选项] [参数]

【选项】

-o 只显示有IO操作的进程
-b 批量显示，无交互，主要用作记录到文件
-n NUM 显示NUM次，主要用于非交互式模式
-d SEC 间隔SEC秒显示一次
-p PID 针对进程进行输出
-u USER 根据进程执行用户进行输出

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

□ badblocks

安装方法: `yum install iotop`

- 硬盘出现坏道会严重影响主机运行, badblocks工具可检测硬盘是否存在坏道。
- 对主机/dev/sda1进行磁盘检测, 查看是否存在损坏。

```
[root@Project-11-Task-01 ~]# badblocks /dev/sda1
[root@Project-11-Task-01 ~]#
[root@Project-11-Task-01 ~]# badblocks -b 4096 -c 16 /dev/sda1 -o badblock-list
[root@Project-11-Task-01 ~]#
[root@Project-11-Task-01 ~]# cat badblock-list
[root@Project-11-Task-01 ~]#
[root@Project-11-Task-01 ~]#
```

#若存在坏道则进行输出, 若未输出则说明文件系统未出现坏道。
#在检测主机文件系统前, 需要卸载目标分区, 再进行检测, 检测完成再进行挂载。



2.查看系统的性能状态

□ badblocks

- 硬盘出现坏道会严重影响主机运行, badblocks工具可检测硬盘是否存在坏道。

【语法】

badblocks [选项] [参数]

【选项】

-b <区块大小> 指定磁盘的区块大小, 单位为字节
-o <输出文件> 将检查的结果写入指定的文件
-s 在检查时显示进度
-v 执行时显示详细的信息
-w 在检查时, 执行写入测试

【参数】

文件系统 需要检测是否存在坏道的磁盘分区文件系统



2.查看系统的性能状态

- 程序是存储在磁盘上包含可执行机器指令和数据的静态实体，进程是在操作系统中执行特定任务的动态实体。
 - 一个程序允许有多个进程，而每个运行中的程序至少由一个进程组成。
 - 以FTP服务器为例，有多个用户使用FTP服务，则系统会开启多个服务进程以满足用户需求。
- Linux操作系统作为多用户多任务操作系统，每个进程与其他进程都是彼此独立的，都有独立的权限与职责，用户的应用程序不会干扰到其他用户的程序或操作系统本身。



2.查看系统的性能状态

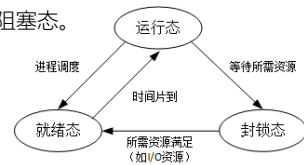
- 进程间有并列关系，也有父进程和子进程的关系，进程间的父子关系实际上是管理和被管理的关系，当父进程终止时，子进程也随之而终止，但子进程终止，父进程并不一定终止。
- Linux操作系统包括如下3种不同类型的进程，每种进程都有其自己的特点和属性。
 - 交互进程：
 - 由Shell启动的进程，可在前台运行，也可在后台运行。
 - 批处理进程：
 - 该进程和终端没有关联，是一个进程序列。
 - 守护进程：
 - 操作系统启动时，随之启动并持续运行的进程。



2.查看系统的性能状态

2.4 进程监控与管理

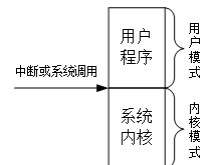
- Linux操作系统进程具有3类状态，分别为：运行态、就绪态和封锁态。
 - 运行态：
 - 当前进程已分配到CPU，正在处理器上执行时的状态。
 - 就绪态：
 - 进程已具备运行条件，但因为其他进程正占用CPU，暂时不能运行而等待分配CPU的状态。
 - 封锁态：
 - 进程因等待某种事件发生而暂时不能运行的状态，也被称为阻塞态。
 - 进程的状态可依据一定的条件和原因而变化。



2.查看系统的性能状态

2.4 进程监控与管理

- 在Linux操作系统中，进程执行模式划分为用户模式和内核模式。
 - 用户模式。当前运行的是用户程序、应用程序或者内核之外的系统程序，则对应进程就在用户模式下运行；
 - 内核模式。在用户程序执行过程中出现系统调用或者发生中断事件，就要运行操作系统（即核心）程序，进程模式就变成内核模式。
- 按照进程的功能和运行程序分类，进程可划分为两大类：
 - 一类是系统进程，只运行在内核模式，执行操作系统代码，完成一些管理性的工作，例如内存分配、进程切换；
 - 一类是用户进程，通常在用户模式中执行，并通过系统调用或在出现中断、异常进入内核模式。



2.查看系统的性能状态

2.4 进程监控与管理

□ 进程优先级

- 在Linux操作系统中，进程在执行时都会赋予一个优先等级，等级越高，进程获得CPU时间就会越多，所以级别越高的进程，运行的时间就会越短，反之则需要较长的运行时间。
- 进程的优先等级范围为-20~19，其中，-20表示最高等级，而19则是最低。等级-1~-20只有root用户可以设置，进程运行的默认优先等级为0。

□ 进程启动

- 在Linux操作系统中，启动进程有两个主要途径：前台启动和后台启动。
- 前台启动。
 - 手工启动一个进程的最常用方式（例如，用户输入一个ls命令，就会启动一个前台进程）。前台启动进程的特点就是会一直占据着终端窗口，除非前台进程运行完毕，否则用户无法在该终端窗口中再执行其他命令。前台启动进程的方式一般比较适合运行时间较短、需要与用户交互的程序。
- 后台启动。
 - 后台启动进程在运行后，不管是否已经完成，都会立即返回到Shell提示符下，不会占用终端窗口，用户可以在终端窗口上继续运行其他程序，后台启动进程会由系统继续调度执行。后台启动进程的方法是，在执行的命令后面加上'&'字符。

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

2.4 进程监控与管理

□ ps

- 查看命令执行时运行的进程信息。

```
[root@Project-11-Task-01 ~]# ps
  PID TTY          TIME CMD
 1971 pts/0    00:00:00 bash
 19699 pts/0    00:00:00 ps
[root@Project-11-Task-01 ~]#
```

PID 运行命令 (CMD) 的进程编号
 TTY 命令所运行的位置 (终端)
 TIME 运行该命令所占用的CPU处理时间
 CMD 进程所运行的命令

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

2.查看系统的性能状态

2.4 进程监控与管理

□ ps

- 查看命令执行时运行的进程信息。

【语法】

ps [选项]

【选项】

-a 显示所有终端下执行的程序
 a 显示现行终端下的所有程序，包括其他用户的程序
 c 列出程序时，显示每个程序真正的指令名称，不包含路径
 e 列出程序时，显示每个程序所使用的环境变量
 g 显示现行终端机下的所有程序
 h 不显示标题列
 n 以数字标识USER和WCHAN列信息
 r 只列出现行终端机正在执行中的程序
 s 采用程序信号的格式显示进程运行状态
 v 采用虚拟内存的格式显示进程状态
 x 显示所有进程，不以终端来区分

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

2.4 进程监控与管理

□ ps aux

- 案例：以用户为主，显示所有用户相关的进程运行情况信息。

进程常见的状态字符选项内容

D 无法运行的休眠状态
 R 正在运行状态
 S 处于休眠状态
 T 处于停止或被追踪状态
 W 进入内存交换状态
 X 死掉的进程状态
 Z “僵尸”进程
 < 优先级高的进程
 N 优先级低的进程
 L 部分被锁进内存
 s 具有多个子进程
 l 多进程
 + 位于后台的进程组

```
[root@Project-11-Task-01 ~]# ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STA START  TIME COMMAND
root         1  0.0  1.5 96732 13444 ?        Ss   13:07  0:03 /usr/sbin/systemd/systemd --system --d
root         2  0.0  0.0   0     0 ?        S    13:07  0:00 [kthreadd]
root         3  0.0  0.0   0     0 ?        I<   13:07  0:00 [rcu_gp]
root         4  0.0  0.0   0     0 ?        I<   13:07  0:00 [rcu_par_gp]
root         6  0.0  0.0   0     0 ?        I<   13:07  0:00 [kworker/8:0H-kblockd]
root         8  0.0  0.0   0     0 ?        I<   13:07  0:00 [mm_percpu_wq]
root         9  0.0  0.0   0     0 ?        S    13:07  0:00 [ksoftirqd/0]
root        10  0.0  0.0   0     0 ?        R    13:07  0:00 [rcu_sched]
root        11  0.0  0.0   0     0 ?        S    13:07  0:00 [migration/0]
root        12  0.0  0.0   0     0 ?        S    13:07  0:00 [watchdog/0]
root        13  0.0  0.0   0     0 ?        S    13:07  0:00 [cpuhp/0]
root        15  0.0  0.0   0     0 ?        S    13:07  0:00 [kdevtmpfs]
root        16  0.0  0.0   0     0 ?        I<   13:07  0:00 [netns]
root        17  0.0  0.0   0     0 ?        S    13:07  0:00 [kauditd]
root        18  0.0  0.0   0     0 ?        S    13:07  0:00 [khungtaskd]
root        19  0.0  0.0   0     0 ?        S    13:07  0:00 [oom_reaper]
root        20  0.0  0.0   0     0 ?        I<   13:07  0:00 [writeback]
root        21  0.0  0.0   0     0 ?        S    13:07  0:00 [kcompactd0]
root        22  0.0  0.0   0     0 ?        SN   13:07  0:00 [kamd]
root        23  0.0  0.0   0     0 ?        SN   13:07  0:00 [khugepaged]
root        24  0.0  0.0   0     0 ?        I<   13:07  0:00 [crypto]
root        25  0.0  0.0   0     0 ?        I<   13:07  0:00 [kintegrityd]
root        26  0.0  0.0   0     0 ?        I<   13:07  0:00 [kblockd]
root        27  0.0  0.0   0     0 ?        I<   13:07  0:00 [tpm_dev_wq]
root        28  0.0  0.0   0     0 ?        I<   13:07  0:00 [md]
root        29  0.0  0.0   0     0 ?        I<   13:07  0:00 [edac-poller]
root        30  0.0  0.0   0     0 ?        S    13:07  0:00 [watchdog]
root        39  0.0  0.0   0     0 ?        S    13:07  0:00 [kswapd0]
root       900  0.0  0.0   0     0 ?        I<   13:07  0:00 [kthrotld]
```

USER 进程属主
 PID 进程ID
 %CPU 进程占用CPU的百分比
 %MEM 进程占用内存的百分比
 VSZ 进程使用的虚拟内存量 (KB)
 RSS 进程占用的固定内存量 (KB)
 TTY 进程在哪个终端上运行
 STAT 进程当前运行状态
 START 进程启动的时间
 TIME 进程使用CPU的时间
 COMMAND 进程执行命令的名称和参数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

□ ps lax

- 案例：以进程为主，显示进程详细运行情况信息。

```
[root@Project-11-Task-01 ~]# ps lax
F UID PID PPID PRI NI VSZ RSS WCHAN STAT TTY TIME COMMAND
4 0 1 0 20 0 96712 13444 do_oppo S ? 0:00 /usr/lib/systemd/systemd --system
1 0 2 0 20 0 0 0 - S ? 0:00 [kthread]
1 0 3 2 0 -20 0 0 0 - I ? 0:00 [rcu_gp]
1 0 4 2 0 -20 0 0 0 - I ? 0:00 [rcu_gar_gp]
1 0 6 2 0 -20 0 0 0 - I ? 0:00 [kworker/0:0H-kblockd]
1 0 8 2 0 -20 0 0 0 - I ? 0:00 [m_percpu_wq1]
1 0 9 2 20 0 0 0 - S ? 0:00 [ksftirqd/0]
1 0 10 2 20 0 0 0 - R ? 0:00 [rcu_sched]
1 0 11 2 -100 - 0 0 - S ? 0:00 [migration/0]
5 0 12 2 -100 - 0 0 - S ? 0:00 [watchdog/0]
1 0 13 2 20 0 0 0 - S ? 0:00 [cpuhp/0]
5 0 15 2 20 0 0 0 - S ? 0:00 [kdevtmpfs]
1 0 16 2 0 -20 0 0 - I ? 0:00 [netns]
1 0 17 2 20 0 0 0 - S ? 0:00 [kauditd]
1 0 18 2 20 0 0 0 - S ? 0:00 [khungtaskd]
1 0 19 2 20 0 0 0 - S ? 0:00 [oom_reaper]
1 0 20 2 0 -20 0 0 - I ? 0:00 [writeback]
1 0 21 2 20 0 0 0 - S ? 0:00 [kcompactd0]
1 0 22 2 25 5 0 0 - SM ? 0:00 [kmem]
1 0 23 2 39 19 0 0 - SM ? 0:00 [khugepaged]
1 0 24 2 0 -20 0 0 - I ? 0:00 [crypto]
1 0 25 2 0 -20 0 0 - I ? 0:00 [kintegrityd]
1 0 26 2 0 -20 0 0 - I ? 0:00 [kblockd]
1 0 27 2 0 -20 0 0 - I ? 0:00 [pm_dev_wq]
1 0 28 2 0 -20 0 0 - I ? 0:00 [ed]
1 0 29 2 0 -20 0 0 - I ? 0:00 [edac-poller]
1 0 30 2 -100 - 0 0 - S ? 0:00 [watchdogd]
1 0 30 2 20 0 0 0 - S ? 0:00 [kswapd0]
1 0 90 2 0 -20 0 0 - I ? 0:00 [kthrotld]
```

ps lax查看结果与ps aux不同的选项内容
 F 进程的属主
 UID 进程使用者ID
 PPID 父进程ID
 PRI 内核调度优先级
 NI 进程优先级标识
 WCHAN 正在等待的进程资源

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2.查看系统的性能状态

□ kill

- 对系统运行中的进程进行管理，例如中断某个进程的运行等。

```
[root@Project-11-Task-01 ~]# ps aux | grep httpd
root 830 0.0 1.2 280348 10652 ? Ss 13:07 0:00 /usr/sbin/httpd -DFOREGROUND
apache 10459 0.0 0.0 292904 8032 ? S 16:11 0:00 /usr/sbin/httpd -DFOREGROUND
apache 10460 0.0 1.3 1350336 11448 ? S1 16:11 0:00 /usr/sbin/httpd -DFOREGROUND
apache 10461 0.0 1.6 1481472 13488 ? S1 16:11 0:00 /usr/sbin/httpd -DFOREGROUND
apache 10462 0.1 1.3 1350336 11436 ? S1 16:11 0:03 /usr/sbin/httpd -DFOREGROUND
root 10709 0.0 0.1 12320 1064 pts/0 S+ 17:00 0:00 grep --color=auto httpd
[root@Project-11-Task-01 ~]# kill -9 830
[root@Project-11-Task-01 ~]# ps aux | grep httpd
root 10725 0.0 0.1 12320 1064 pts/0 R+ 17:01 0:00 grep --color=auto httpd
[root@Project-11-Task-01 ~]#
```

kill命令可根据预设值进行进程操作。
 SIGTERM (15) : 可将指定程序终止
 SIGKILL (9) : 可尝试强制删除程序

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



2. 查看系统的性能状态

2.4 进程监控与管理

kill

- 对系统运行中的进程进行管理，例如中断某个进程的运行等。

【语法】

kill [选项] [参数]

【选项】

-a 当处理当前进程时，不限制命令名和进程号的对应关系
 -l <编号> 若不加信息编号选项，则-l参数会列出全部的信息名称
 -p 指定kill命令只打印相关进程的进程号，不发送任何信号
 -s <编号> 指定要发送的消息
 -u 指定用户

【参数】

进程号 指定要终止的进程

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3. 实时监控系统的运行状态

3.1 实时监控工具

top

- 是综合检测系统运行信息的工具，可实时查看系统运行状态。

```
top - 17:55:07 up 4147, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 87 total, 2 running, 85 sleeping, 0 stopped, 0 zombie
%cpu(s):  6.0 us,  0.0 sy,  0.0 ni, 99.7 id,  0.0 wa,  0.2 hi,  0.0 si,  0.0 st
MiB Mem :  821.4 total,  169.8 free,  184.2 used,  468.2 buff/cache
MiB Swap:  820.0 total,  810.7 free,  9.2 used,  491.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	MEM	TIME	COMMAND
19741	root	20	0	63972	4288	3660	R	0.7	0.5	0:00.02	top
1	root	20	0	176640	13564	9188	S	0.0	1.6	0:03.03	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
9	root	20	0	0	0	0	S	0.0	0.0	0:00.22	ksfttqg/0
10	root	20	0	0	0	0	I	0.0	0.0	0:00.54	rcu_sched
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.04	watchdog/0
13	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpu/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
16	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper
20	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	writeback
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kcompactd0
22	root	25	0	0	0	0	S	0.0	0.0	0:00.04	watchdog/0
23	root	39	19	0	0	0	S	0.0	0.0	0:00.04	khugepaged
24	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	crypto
25	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	krtgtrtyd
26	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kblockd

在执行top命令时，可使用交互命令进行快捷操作。

k 终止一个进程
 i 忽略闲置和僵死进程
 q 退出程序
 r 重新设置一个进程的优先级
 S 切换到累积模式
 s 改变刷新时间（单位为秒），如果有小数，按毫秒计。输入0则系统将不断刷新，默认值是5s

每F 从当前显示中添加或删除项目
 o或O 改变显示项目的顺序
 l 切换显示平均负载和启动时间信息
 L 切换显示内存信息
 m 切换显示内存信息
 t 切换显示进程和CPU状态信息
 T 切换显示命令名称和完整命令行
 c 切换显示命令名称和完整命令行
 M 根据驻留内存大小进行排序
 P 根据CPU使用百分比大小进行排序
 T 根据时间/累积时间进行排序

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

□ top

- 查看指定进程的运行状态信息。

```
top - 17:58:59 up 4:51, 2 users, load average: 0.00, 0.00, 0.00
Tasks: 1 total, 0 running, 1 sleeping, 0 stopped, 0 zombie
MEm(s): 0.0 us, 0.0 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.2 hi, 0.0 si, 0.0 st
MiB Mem:  821.4 total,  141.5 free,  211.1 used,  464.7 buff/cache
MiB Swap:  820.0 total,  810.7 free,    9.2 used,  464.4 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S %CPU  MEM  TIME+ COMMAND
  1 root        20   0 17640 13600 9188 S  0.0  1.6  0:01.05 systemd
```

在执行top命令时,可使用交互命令进行快捷操作。

- k 终止一个进程
- i 忽略闲置和僵死进程
- q 退出程序
- r 重新设置一个进程的优先级
- S 切换到累积模式
- s 改变刷新时间(单位为秒)。如果有小数就换算成ms。输入0值则系统将不断刷新。默认值为5s
- 键F 从当前显示中添加或删除项目
- o或O 改变显示项目的顺序
- l 切换显示平均负载和启动时间信息
- m 切换显示内存信息
- t 切换显示进程和CPU状态信息
- c 切换显示命令名称和完整命令行
- M 根据驻留内存大小进行排序
- P 根据CPU使用百分比大小进行排序
- T 根据时间/累积时间进行排序

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

□ top

- 查看指定进程的运行状态信息。

【语法】

top [选项]

【选项】

- b 以批处理模式操作
- c 显示整个命令行
- d 屏幕刷新间隔时间
- l 忽略失效过程
- s 保密模式
- S 累积模式
- I <时间> 设置间隔时间
- u <用户名> 指定用户名
- p <进程号> 指定进程号
- n <次数> 循环显示的次数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.1 实时监控工具

□ sar

安装方法: yum install sysstat

- 对系统当前的状态进行取样, 然后通过计算数据和比例来分析系统的当前状态。

```
[root@Project-11-Task-01 ~]# sar 1
Linux 4.18.0-147.5.1.el8_1.x86_64 (Project-11-Task-01) 2020年04月19日 _x86_64_ (1 CPU)

1M07:05分 07秒 CPU      %user   %nice    %system %iowait  %steal   %idle
1M07:05分 08秒 all      0.00    0.00    0.00    0.00    0.00   100.00
1M07:05分 09秒 all      0.00    0.00    1.00    0.00    0.00    99.00
1M07:05分 10秒 all      0.00    0.00    0.00    0.00    0.00   100.00
1M07:05分 11秒 all      0.00    0.00    0.00    0.00    0.00   100.00
^C

平均时间:  all      0.00    0.00    0.25    0.00    0.00    99.75
[root@Project-11-Task-01 ~]# sar 1 5 -o
Linux 4.18.0-147.5.1.el8_1.x86_64 (Project-11-Task-01) 2020年04月19日 _x86_64_ (1 CPU)

1M07:06分 01秒 CPU      %user   %nice    %system %iowait  %steal   %idle
1M07:06分 02秒 all      0.00    0.00    0.00    0.00    0.00   100.00
1M07:06分 03秒 all      0.00    0.00    1.00    0.00    0.00    99.00
1M07:06分 04秒 all      1.00    0.00    1.00    0.00    0.00    98.00
1M07:06分 05秒 all      0.00    0.00    0.00    0.00    0.00   100.00
1M07:06分 06秒 all      0.00    0.00    1.98    0.00    0.00    98.02
平均时间:  all      0.20    0.00    0.00    0.00    0.00    99.00
[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.1 实时监控工具

□ sar

- 对系统当前的状态进行取样, 然后通过计算数据和比例来分析系统的当前状态。

【语法】

sar [选项] [参数]

【选项】

-A 显示所有的报告信息
 -b 显示I/O速率
 -B 显示换页状态
 -c 显示进程创建活动
 -d 显示每个块设备的状态
 -e 设置显示报告的结束时间
 -f 从指定文件提取报告
 -i 设置状态信息刷新的间隔时间
 -P 显示每个CPU的状态
 -R 显示内存状态

【选项】

-U 显示CPU利用率
 -V 显示索引节点, 文件和其他内核表的状态
 -W 显示交换分区状态
 -X 显示给定进程的状态

【参数】

时间间隔 设置每次获取数据展示的时间间隔 (秒)
 次数 显示报告的次数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.1 实时监控工具

□ htop

- 互动进程查看器，可实时查看运行进程信息。

安装方法: yum install htop

```

CPU:  1.3%  Tasks: 31, 224 thr: 1 running
Mem:  231M/812M  Load average: 0.27 0.18 0.09
Swap:  14.3M/820M  Uptime: 05:08:03

PID  PPID  CPU%  MEM%  VSZ  RSS  TTY  STAT  COMMAND
2386  root   0    0 2760 3768 3152  R  1.3  0.4  0:00.00 htop
19781 apache 20  0 1310M 7768 3072  S  0.7  0.9  0:00.21 /usr/sbin/httpd -DFOREGROUND
1 root   20  0 170M 13360 1024  S  0.0  1.6  0:01.42 /usr/lib/systemd/systemd --system --des
615 root  20  0 100M 10520 1480  S  0.0  1.3  0:02.47 /usr/lib/systemd/systemd-journald
767 root  16 -4 61240 2348 1856  S  0.0  0.3  0:00.00 /sbin/auditd
766 root  16 -4 61240 2348 1856  S  0.0  0.3  0:00.02 /sbin/auditd
795 root  20  0 150M 5180 4620  S  0.0  0.6  0:01.40 /sbin/rngd -f
790 root  20  0 150M 5180 4620  S  0.0  0.6  0:01.50 /sbin/rngd -f
788 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.00 /usr/lib/polkit-1/polkitd --no-debug
799 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.02 /usr/lib/polkit-1/polkitd --no-debug
800 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.00 /usr/lib/polkit-1/polkitd --no-debug
801 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.00 /usr/lib/polkit-1/polkitd --no-debug
804 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.00 /usr/lib/polkit-1/polkitd --no-debug
791 polkitd 20  0 150M 1824 12184  S  0.0  2.2  0:00.10 /usr/lib/polkit-1/polkitd --no-debug
786 dbus  20  0 81944 5096 4256  S  0.0  0.6  0:00.00 /usr/lib/dbus-daemon --system --address
792 dbus  20  0 81944 5096 4256  S  0.0  0.6  0:01.70 /usr/bin/dbus-daemon --system --address
808 root  20  0 17572 7916 4032  S  0.0  0.9  0:00.03 /usr/lib/systemd/systemd-logind
824 root  20  0 300M 1256 11652  S  0.0  1.5  0:00.23 /usr/sbin/NetworkManager --no-daemon
825 root  20  0 300M 1256 11652  S  0.0  1.5  0:00.00 /usr/sbin/NetworkManager --no-daemon
819 root  20  0 300M 1256 11652  S  0.0  1.5  0:00.36 /usr/sbin/NetworkManager --no-daemon
1245 root  20  0 414M 18844 1612  S  0.0  2.4  0:02.13 /usr/libexec/platform-python -E /usr/s
1254 root  20  0 414M 18844 1612  S  0.0  2.4  0:00.00 /usr/libexec/platform-python -E /usr/s
1260 root  20  0 414M 18844 1612  S  0.0  2.4  0:00.00 /usr/libexec/platform-python -E /usr/s
827 root  20  0 414M 18844 1612  S  0.0  2.4  0:02.47 /usr/libexec/platform-python -E /usr/s

```

执行htop命令时，可使用交互命令进行快捷操作。

上/下键或PgUP/PgDn	选定想要的进程
左/右键或Home/End	移动字段
Space	编辑/取消标记一个进程。命令可作用域多个进程
U	取消标记所有进程
s	选择某一进程，按s用strace追踪进程的系统调用
l	显示进程打开的文件，如果安装了lsuf，按此键可以显示进程所打开的文件
I	倒序排序，如果排序是正序的，则反转成倒序的，反之亦然
+, -	在树视图模式下，展开或折叠子树
a	在有多个处理器核心上，设置CPU affinity，标记一个进程允许使用哪些CPU
u	显示特定用户进程
M	按内存使用顺序
P	按CPU使用顺序
T	按Time+使用顺序
F	跟踪进程，如果排序引起选定的进程在列表上到处移动，让选定条跟随该进程，通过这种方式，可以让一个进程在屏幕上一直可见，使用方向键会停止该功能
K	显示/隐藏内核线程
H	显示/隐藏用户线程
Ctrl-L	刷新
Numbers	用户PID查找，输入PID号，光标将移动到相应的进程上

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.1 实时监控工具

□ htop

- 互动进程查看器，可实时查看运行进程信息。

【语法】

htop [选项]

【选项】

-C或--no-color	使用一个单色的配色方案
-d或--delay=DELAY	设置延迟更新时间，单位秒
-u或--user=USERNAME	只显示一个给定的用户的过程
-p或--pid= PID,[PID,PID..]	只显示给定的PIDs (进程号组信息)
-s或--sort-key COLUMN	以给定的列进行排序

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.1 实时监控工具

atop

安装方法: yum install atop

- 监控系统资源与进程, 并以一定的频率记录系统的运行状态。

The screenshot shows the atop tool output. The top section displays system-level statistics such as CPU usage (sys 11.76%), memory usage (MEM 821.0M), and disk usage (DISK 1.73815). The middle section lists network statistics (NET) for various protocols like transport, network, and endpoints. The bottom section is a table of running processes, including columns for PID, %CPU, %MEM, %IOPS, %IO, %DISK, %NET, %S, %EXC, %TH, %S, %CPUB, %CPU, and %MEM. Processes like systemd, tuned, and rsyslogd are visible.

atop工具可查看系统PRC (进程)、CPL (当前执行进程的特权等级)、LVM (逻辑卷)、DSK (磁盘)、NET (网络) 等运行信息。

PRC运行结果中选项内容如下。

- sys 过去10s所有的进程在内核运行时间总和
- usr 过去10s所有的进程以用户状态运行时间总和
- #zombie 过去10s僵死进程的数量
- #exit 在10s采样周期内退出的进程数量

CPL运行结果中选项内容如下。

- avg1/avg5/avg15 过去1/5/15分钟进程等待队列数
- cws 上下文交换次数
- intr 中断发生的次数

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.1 实时监控工具

atop

安装方法: yum install atop

- 监控系统资源与进程, 并以一定的频率记录系统的运行状态。

This screenshot is identical to the one on page 75, showing the same atop tool output with system statistics, network data, and a list of running processes.

atop工具可查看系统PRC (进程)、CPL (当前执行进程的特权等级)、LVM (逻辑卷)、DSK (磁盘)、NET (网络) 等运行信息。

LVM和DSK运行结果中选项内容如下。

- busy 磁盘繁忙占比
- read 每秒读请求数
- write 写请求数
- avio 磁盘的平均I/O时间

NET展示传输层 (transport)、网络层 (network)、网络接口 (ens192, lo) 的传输信息。运行结果中选项内容如下。

- tcp/tcpo 传入/传出的TCP数据包的大小
- udp/udp0 传入/传出的UDP数据包的大小
- ip/ipo 接收/发送IP数据包数量
- ipfwr IP数据包转发数量
- deliv 网络传送数据包数量
- pcki/pck0 传入/传出的数据包大小
- sp 网卡的带宽
- si/so 每秒传入/传出的数据大小

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.1 实时监控工具

□ atop

- 监控系统资源与进程，并以一定的频率记录系统的运行状态。

【语法】

atop [选项]

【选项】

-a 展示所有的进程信息
-P 计算每个进程的比例集大小
-L 非屏幕输出情况下的备用行长度
-f 用系统统计显示固定的行数
-F 禁止系统资源的排序
-G 在输出中禁止退出进程
-l 限制显示某些资源的行数
-y 显示单个线程运行状态信息
-l 显示系统平均每秒S.O进程总值
-x 系统进程高使用时也单色显示
-g 显示一般或默认进程信息
-m 显示与内存相关的进程信息

【选项】

-d 显示与磁盘相关的进程信息
-n 显示与网络相关的进程信息
-s 显示与调度相关的进程信息
-v 显示与进程ID、用户、用户组、日期等进程信息
-c 显示每个进程的命令行信息
-o 显示用户自定义的进程信息
-u 显示每个用户累计的进程信息
-p 显示每个应用程序累计的进程信息（即同名）
-j 显示每个容器累计的进程信息
-C 按照CPU使用量大小顺序排序
-M 按照内存使用率大小顺序排序
-D 按照磁盘活动顺序排序
-N 按照网络活动顺序排序
-A 按最活跃资源顺序排序
-w 将原始数据压缩并写入文件
-r 从压缩文件中读取原始数据

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.2 盯屏监控工具

□ dstat

安装方法: `yum install dstat`

- 全能系统信息监控工具，可实时监控主机CPU、磁盘、网络、IO、内存的使用情况。

```
[root@Project-11-Task-01 ~]# dstat
You did not select any stats, using -cdngy by default.
-----total-----disk-----net-----paging-----system-----
usr  syz  ldt  wai  atli  read  writi  recv  sendi  in  out  |  int  csw
1  0  0  0  0  0  0  60  650  0  0  0  98  174
1  1  98  0  0  0  0  120  326  0  0  0  93  174
0  0  99  0  0  0  0  120  326  0  0  0  103  179
1  0  99  0  0  0  0  120  326  0  0  0  114  176
0  0  99  0  0  0  0  120  326  0  0  0  105  177
1  0  99  0  0  0  0  120  326  0  0  0  115  182
1  0  94  4  0  12k  40958  2100  326  13.00  0  0  154  244
1  98  0  0  0  0  1200  342  0  0  0  150  252
6  1  90  4  0  330k  0  2027  324  0  0  0  240  338
84  5  1  8  0  4870k  0  1147  344  0  0  0  1140  347
50  2  34  6  0  2077k  1860  342  0  0  0  932  556
1  0  98  0  0  0  0  1380  342  0  0  0  114  193
1  0  98  0  0  0  0  1979  326  0  0  0  130  211
1  0  98  0  0  0  0  1200  326  0  0  0  127  198
1  0  99  0  0  0  0  1679  326  0  0  0  136  207
1  0  98  0  0  0  0  1441  326  0  0  0  132  202
1  0  98  0  0  0  0  300  326  0  0  0  137  204
0  0  99  0  0  0  1k  180  325  0  0  0  166  310
2  0  95  3  0  420k  0  803  717  0  0  0  208  301
1  0  91  6  0  550k  218  425  0  0  0  138  232
1  1  97  0  0  0  0  120  326  0  0  0  410  333
2  0  98  0  0  0  0  120  326  0  0  0  232  234
1  0  98  0  0  0  0  240  326  0  0  0  120  195
1  0  99  0  0  0  0  180  446  0  0  0  94  172
0  0  99  0  0  0  0  60  326  0  0  0  108  168
2  0  98  0  0  0  0  120  326  0  0  0  106  180
0  0  100  0  0  0  0  180  326  0  0  0  115  185
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.2 盯屏监控工具

□ dstat

安装方法: yum install dstat

- 对主机进行监控, 并将监控结果保存在dstat.txt文件中。

```
[root@Project-11-Task-01 ~]# dstat -tsp --socket --fs --output dstat.txt
-----system-----swap-----procs-----sockets-----filesystem-----
time | used | free | run | blk | new | tat | tcp | udp | rwm | frg | files | inodes
19-04 18:35:46 | 15M | 845M | 1.0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:47 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:48 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:49 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:50 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:51 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:52 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566
19-04 18:35:53 | 15M | 845M | 0 | 0 | 0 | 187 | 3 | 2 | 1 | 0 | 1824 | 32566

[root@Project-11-Task-01 ~]# cat dstat.txt
"pcp-dstat 4.3.2 CSV Output"
"Author": "PCP team <pcp@groups.io> and Dag Wieers <dag@wieers.com>,,, "URL": "https://pcp.io/ and ht
tp://dag.wieers.com/home-made/dstat/"
"Host": "Project-11-Task-01",, "User": "root"
"CdLine": "dstat -tsp --socket --fs --output dstat.txt",, "Date": "19 Apr 2020 18:35:46 CST"
"system", "swap",, "procs",, "sockets",, "filesystem"
"time", "swapused", "swapfree", "run", "blk", "new", "tat", "tcp", "udp", "raw", "frg", "files", "inodes"
19-04 18:35:46,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:47,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:48,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:49,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:50,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:51,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:52,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
19-04 18:35:53,15824128,844804896,0,0,0,187,3,2,1,0,1824,32566
[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / https://internet.hactcm.edu.cn / http://www.51xueweb.cn



3.实时监控系统的运行状态

3.2 盯屏监控工具

□ dstat

- 全能系统信息监控工具, 可实时监控主机CPU、磁盘、网络、IO、内存的使用情况。

【语法】

dstat [选项]

【选项】

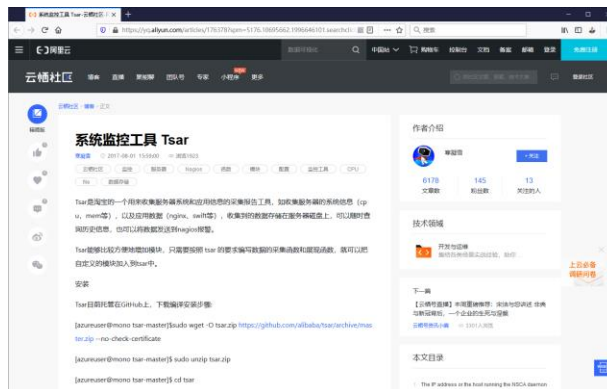
-c	显示CPU系统占用、 用户占用、空闲、等待、中断等信息		
-C	当有多个CPU时候, 此参数可按需分别显示CPU状态		
-d	显示磁盘读写数据大小		
-n	显示网络状态		
-l	显示系统负载情况		
-m	显示内存使用情况	-N <网卡>	可指定显示网卡的信息
-g	显示页面使用情况	--ipc	显示ipc消息队列、信号等信息
-p	显示进程状态	--socket	用来显示tcp、udp端口状态
-s	显示交换分区使用情况	--output 文件	可以将状态信息以csv格式重定向 指定文件中
-r	显示I/O请求情况		
-y	系统状态		

河南中医药大学 / 阮晓龙 / 13938213680 / https://internet.hactcm.edu.cn / http://www.51xueweb.cn



3.实时监控系统的运行状态

3.2 盯屏监控工具



河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

3.实时监控系统的运行状态

3.3 PROC

- Linux系统上的proc是一种伪文件系统（即虚拟文件系统），只存在内存当中，是存储当前内核运行状态的一系列特殊文件，用户可通过该类型文件查看主机以及当前正在运行进程的信息，甚至可以通过更改其中某些文件来改变内核的运行状态。
 - 鉴于proc文件的特殊性，其目录下的文件也常被称作为虚拟文件，其中大多数文件的时间及日期属性通常为当前系统时间和日期，虚拟文件随时刷新。
 - 为了查看和使用上的方便，通常会按照相关性分类存储于不同的目录甚至子目录中，大多数虚拟文件都可使用文件查看命令（如cat、more、less等）查看，有些文件信息表述的内容是一目了然的，但也有文件的信息不具备可读性，不过，可读性较差的文件可使用一些命令（如apm、free、lspci或top等）来提高文件的可读性。

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

表 12-6-1 /proc 下常见的目录

目录	描述
/proc/apm	高级电源管理 (APM) 版本信息及电池相关状态信息, 通常由 apm 命令使用
/proc/buddyinfo	用于诊断内存碎片问题的相关信息
/proc/cmdline	在启动时传递至内核的相关参数信息, 这些信息通常由 lilo (Linux 加载程序) 或 grub (Linux 引导管理程序) 等工具进行传递
/proc/cpuinfo	处理器的相关信息文件
/proc/crypto	系统上已安装内核使用的密码算法及每个算法的详细信息列表
/proc/devices	系统已经加载的所有块设备和字符设备的信息, 包含主设备号和设备组 (与主设备号对应的设备类型) 名
/proc/diskstats	每块磁盘设备的 I/O 统计信息列表 (内核 2.5.69 以后的版本支持此功能)
/proc/dma	每个正在使用且注册的 ISA DMA 通道信息列表
/proc/execdomains	内核当前支持的执行域信息列表
/proc/fb	帧缓冲设备列表文件, 包含帧缓冲设备的设备号和相关驱动信息
/proc/filesystems	当前被内核支持的文件系统类型列表文件, 被标示为 nodev 的文件系统表示不需要该块设备的支持; 通常 "mount" 设备时, 如果没有指定文件系统类型, 将通过此文件来决定其所需文件系统的类型
/proc/interrupts	X86 或 X86_64 体系架构系统上每个 IRQ (Interrupt Request, 中断请求) 相关的中断信息列表
/proc/iomem	每个物理设备上的记忆体 (RAM 或者 ROM) 在系统内存中的映射信息
/proc/ioports	当前正在使用且已经被注册过的与物理设备进行通讯的输入-输出口范围信息列表
/proc/kallsyms	模块管理工具, 用来动态链接或绑定可装模块的符号定义, 由内核输出 (内核 2.5.71 以后的版本支持此功能), 通常这个文件中的信息量较大



表 12-6-1 /proc 下常见的目录

目录	描述
/proc/core	系统使用的物理内存以 ELF 核心文件 (core file) 格式存储, 其文件大小为已使用物理内存加上 4KB, 此文件用来检查内核数据结构的当前状态, 通常由 GDB 调试工具使用, 但不能使用文件查看命令打开此文件
/proc/kmsg	此文件用来保存由内核输出的信息, 通常由/sbin/klogd 或/bin/dmsg 等程序使用, 不能使用文件查看命令打开此文件
/proc/loadavg	保存关于 CPU 和磁盘 I/O 的负载平均值, 其前三列分别表示每 1 分钟、每 5 分钟及每 15 分钟的负载平均值, 类似于 uptime 命令输出的相关信息; 第四列是由斜线隔开的两个数值, 前者表示当前正由内核调度的实体 (进程和线程) 的数目, 后者表示系统当前存活的内核调度实体的数目; 第五列表示此文件被查看前最近一个由内核创建的进程 PID
/proc/locks	保存当前由内核锁定的文件相关信息, 包含内核内部的调试数据; 每个锁定占据一行, 且具有一个唯一的编号; 输出信息中每行的第二列表示当前锁定使用的锁定类别, POSIX 表示目前较新类型的文件锁, 有 lockf 系统调用产生, FLOCK 是传统的 UNIX 文件锁, 由 flock 系统调用产生; 第三列也通常由两种类型, ADVISORY 表示不允许其他用户锁定此文件, 但允许读取, MANDATOR Y 表示此文件锁定期间不允许其他用户以任何形式的访问
/proc/mdstat	保存 RAID 相关的多块磁盘的当前状态信息, 在没有使用 RAID 机器上, 其显示为 <none>
/proc/meminfo	系统中关于当前内存的利用状况等的信息, 常由 free 命令使用; 可以使用文件查看命令直接读取, 其内容显示为两列, 前者为统计属性, 后者为对应的值
/proc/mounts	在内核 2.4.29 版本以前, 此文件的内容为系统当前挂载的所有文件系统, 在 2.4.29 以后的内核中引进了每个进程使用独立挂载名称空间的方式, 此文件则随之变成了指向 /proc/self/mounts (每个进程自身挂载名称空间中的所有挂载点列表) 文件的符号链接。



表 12-6-1 /proc 下常见的目录

目录	描述
/proc/partitions	块设备每个分区的主设备号 (major) 和次设备号 (minor) 等信息, 同时包括每个分区所包含的块 (block) 数目
/proc/pci	内核初始化时发现的所有 PCI 设备及其配置信息列表, 其配置信息多为某 PCI 设备相关 IRQ 信息, 可读性不高, 可以用 “/sbin/lspci -vb” 命令获得较易理解的相关信息。在内核 2.6 版本以后, 此文件已为 /proc/bus/pci 目录及其下的文件代替
/proc/slabinfo	在内核中频繁使用的对象 (如 inode、dentry 等) 都有相应的 cache, 即 slab pool, 而 /proc/slabinfo 文件列出了这些对象相关 slab 信息
/proc/stat	实时追踪自系统上次启动以来的多种统计信息, 其中具体每行含义如表 12-6-2 所示
/proc/swaps	当前系统上的交换分区及其空间利用信息, 如果有多个交换分区的话, 则会将每个交换分区的信息分别存储于 /proc/swap 目录中的单独文件中, 而其优先级数字越低, 被使用到的可能性越大
/proc/uptime	系统上次启动以来的运行时间, 其第一个数字表示系统运行时间, 第二个数字表示系统空闲时间, 单位是秒
/proc/version	当前系统运行的内核版本号
/proc/vmstat	当前系统虚拟内存的统计数据, 可读性较好 (内核 2.6 版本以后支持此文件)
/proc/zoneinfo	内存区域 (zone) 的详细信息列表



表 12-6-2 /proc/stat 信息内容

行名	描述
cpu	该行后的八个值分别表示以 1/100 (jiffies) 秒为单位的统计值 (包括系统运行于用户模式、低优先级用户模式、运系统模式、空闲模式、I/O 等待模式的时间等)
intr	该行给出中断的信息, 第一个为自系统启动以来, 发生的所有的中断的次数; 然后每个数对应一个特定的中断自系统启动以来所发生的次数
ctxt	该行展示从系统启动以来 CPU 发生的上下文交换的次数
btime	该行展示从系统启动到现在为止的时间, 单位为秒
processes (total_forks)	该行展示从系统启动以来所创建的任务的个数目
procs_running	该行展示当前运行队列的任务数目
procs_blocked	该行展示当前被阻塞的任务数目



表 12-6-3 /proc/sys 系统目录内容

目录	描述
/proc/sys/abi	此目录主要记录应用程序二进制接口，涉及了程序的多个方面，如目标文件格式、数据类型、函数调用以及函数传递参数等信息
/proc/sys/crypto	此目录主要记录系统中已经安装的相关服务使用的信息加密处理配置
/proc/sys/debug	此目录主要记录系统运行中的调试信息，此目录通常是一空目录
/proc/sys/dev	为系统上特殊设备提供参数信息文件的目录，其不同设备的信息文件分别存储于不同的子目录中，如大多数系统上都具有的/proc/sys/dev/cdrom 和/proc/sys/dev/raid（如果内核编译时开启了支持 raid 的功能）目录，其内存储的通常是系统上 cdrom 和 raid 的相关参数信息文件
/proc/sys/fs	该目录包含一系列选项以及有关文件系统的各个方面信息，包括配额、文件句柄、索引以及系统登录信息
/proc/sys/kernel	此目录文件可用于监视和调整 Linux 操作中的内核相关参数
/proc/sys/net	主要包括了许多网络相关的操作，如 appletalk/、ethernet/、ipv4/、ipx/ 及 ipv6/等，通过改变这些目录中文件，能够在系统运行时调整相关网络参数
/proc/sys/vm	该目录下文件主要用来优化系统中的虚拟内存



表 12-6-4 /proc/1 信息内容

目录或文件	描述
cmdline	启动当前进程的完整命令，但僵尸进程目录中的此文件不包含任何信息
cwd	指当前进程运行目录的一个符号连接
environ	进程的环境变量列表，彼此间用空符号（NULL）隔开；变量用大写字母表示，其值用小写字母表示
exe	指向启动进程的可执行文件（完整路径）的符号链接，通过/proc/N/exe 可以启动当前进程的一个拷贝
/fd	包含当前进程打开的每一个文件的描述符（file descriptor），这些文件描述符是指向实际文件的一个符号链接
limits	当前进程所使用的每一个受限资源的软限制、硬限制和管理单元；此文件仅可由实际启动当前进程的 UID 用户读取
maps	当前进程关联到的每个可执行文件和库文件在内存中的映射区域及其访问权限所组成的列表
mem	当前进程所占用的内存空间，有 open、read、lseek 等系统调用使用，不能被用户读取
root	指向当前进程运行根目录的符号链接；在 Linux 和 UNIX 系统上，通常采用“chroot”命令使每个进程运行于独立的根目录
stat	当前进程的状态信息，包含系统格式化后的数据列，可读性差，通常由“ps”命令使用
statm	当前进程占用内存的状态信息，通常以“页面”（page）表示
/task	包含由当前进程所运行的每一个线程的相关信息，每个线程的相关信息文件均保存在一个由线程号（tid）命名的目录中，其内容类似于每个进程目录中的内容



3.实时监控系统的运行状态

3.3 PROC

□ PROC

- 案例：查看操作系统与内核版本

```
[root@Project-11-Task-01 ~]# cat /proc/version
Linux version 4.10.0-147.5.1.el9_1.x86_64 (mockbuil1@builder.bsys.centos.org) (gcc version 8.3.1 2019
0907 (Red Hat 8.3.1-4)) (GCC) #1 SMP Wed Feb 9 02:00:30 UTC 2020
[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.3 PROC

□ PROC

- 案例：查看主机的内存运行信息

```
[root@Project-11-Task-01 ~]# cat /proc/meminfo
MemTotal: 841104 kB
MemFree: 246904 kB
MemAvailable: 476436 kB
Buffers: 1508 kB
Cached: 341564 kB
SwapCached: 0 kB
Active: 244472 kB
Inactive: 194608 kB
Active(anon): 46964 kB
Inactive(anon): 54888 kB
Active(file): 197588 kB
Inactive(file): 139720 kB
Unevictable: 9684 kB
Mlocked: 9684 kB
SwapTotal: 839676 kB
SwapFree: 825268 kB
Dirty: 12 kB
Writeback: 0 kB
AnonPages: 95068 kB
Mapped: 79484 kB
Shmem: 10800 kB
KReclaimable: 41608 kB
Slab: 98928 kB
SReclaimable: 41608 kB
SUnreclaim: 56412 kB
KernelStack: 5084 kB
PageTables: 9620 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.3 PROC

□ PROC

- 案例：查看主机的CPU运行信息

```
[root@Project-11-Task-01 ~]# cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 94
model name    : Intel(R) Core(TM) i5-6600T CPU @ 2.70GHz
stepping      : 3
cpu MHz      : 2712.004
cache size   : 6144 KB
physical id  : 0
siblings     : 1
core id      : 0
cpu cores    : 1
apicid       : 0
initial apicid : 0
fpu          : yes
fpu_exception : yes
cpuid level  : 22
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx
fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_good nopl xtopology nonstop_tsc cpuid tsc_known
_freq pni pclmulqdq monitor ssse3 cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdRand hyp
ervisor lahf_lm abm 3dnowprefetch impcid_single pt1 fsgsbase avx2 impcid rdseed clflushopt flush_lid
bugs        : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapper itlb_multihit
bogomips    : 5424.00
clflush size : 64
cache_alignm : 64
address sizes : 39 bits physical, 48 bits virtual
power management:

[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.3 PROC

□ PROC

- 案例：查看主机的进程运行信息

```
[root@Project-11-Task-01 ~]# ll /proc/
总用量 0
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 1
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 10
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 11
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 12
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 13
dr-xr-xr-x. 9 systemd-resolve systemd-resolve 0 4月 19 13:07 1448
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 1453
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 15
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 16
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 17
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 18
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 19
dr-xr-xr-x. 9 root      root           0 4月 19 13:21 1968
dr-xr-xr-x. 9 root      root           0 4月 19 16:04 1964
dr-xr-xr-x. 9 root      root           0 4月 19 16:12 19674
dr-xr-xr-x. 9 root      root           0 4月 19 18:15 19750
dr-xr-xr-x. 9 root      root           0 4月 19 17:58 19775
dr-xr-xr-x. 9 apache    apache        0 4月 19 17:58 19776
dr-xr-xr-x. 9 apache    apache        0 4月 19 17:58 19777
dr-xr-xr-x. 9 apache    apache        0 4月 19 17:58 19778
dr-xr-xr-x. 9 apache    apache        0 4月 19 17:58 19779
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 2
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 20
dr-xr-xr-x. 9 root      root           0 4月 19 18:15 20725
dr-xr-xr-x. 9 root      root           0 4月 19 13:07 21
dr-xr-xr-x. 9 root      root           0 4月 19 18:19 21071
dr-xr-xr-x. 9 root      root           0 4月 19 18:20 21401
dr-xr-xr-x. 9 root      root           0 4月 19 18:40 21444
dr-xr-xr-x. 9 root      root           0 4月 19 19:00 21453
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



3.实时监控系统的运行状态

3.3 PROC

□ PROC

- 案例：查看主机的磁盘信息

```
[root@Project-11-Task-01 ~]# cat /proc/diskstats
      8      0 sda 52583 247 5764783 163884 26283 11132 1737535 512835 0 36639 651813 0 0 0 0
      8      1 sda1 33812 2 4276578 8995 117 45 115232 3748 0 7473 8295 0 0 0 0
      8      2 sda2 19438 245 1484253 152453 21899 11887 1622383 382715 0 28697 518896 0 0 0 0
     11      0 srb 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    253      0 dm-0 18638 0 1460485 131236 32938 0 1637743 758675 0 28348 909911 0 0 0 0
    253      1 dm-1 912 0 12656 5482 4312 0 34496 5317 0 926 10719 0 0 0 0
[root@Project-11-Task-01 ~]#
```

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



操作视频 / 现场演示

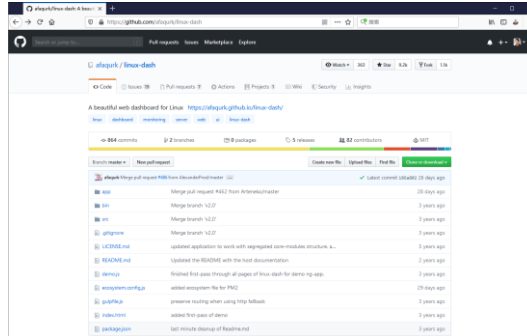
- ✓ 操作演示：
 - 实时监控工具：top htop atop
 - 盯屏监控工具：sar dstat
 - 虚拟文件系统：proc



4.使用Linux-dash实现可视化监控

4.1 linux-dash

- Linux-dash是基于Web的系统状态监控工具，通过Linux-dash可实现对主机进程基本信息、CPU、内存、网络、磁盘、负载等性能监控。



河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

4.使用Linux-dash实现可视化监控

4.2 任务1

任务1：使用Linux-dash实现可视化监控

步骤1：准备Linux-dash部署所需的基本环境

步骤2：获取Linux-dash程序

步骤3：配置Apache发布Linux-dash

步骤4：访问Linux-dash

步骤5：Linux-dash监控信息导读

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



操作视频 / 现场演示

✓ 任务1: 使用Linux-dash实现可视化监控

■ 任务目标:

- 本地主机通过浏览器访问Linux-dash
- 阅读Linux-dash的监控信息

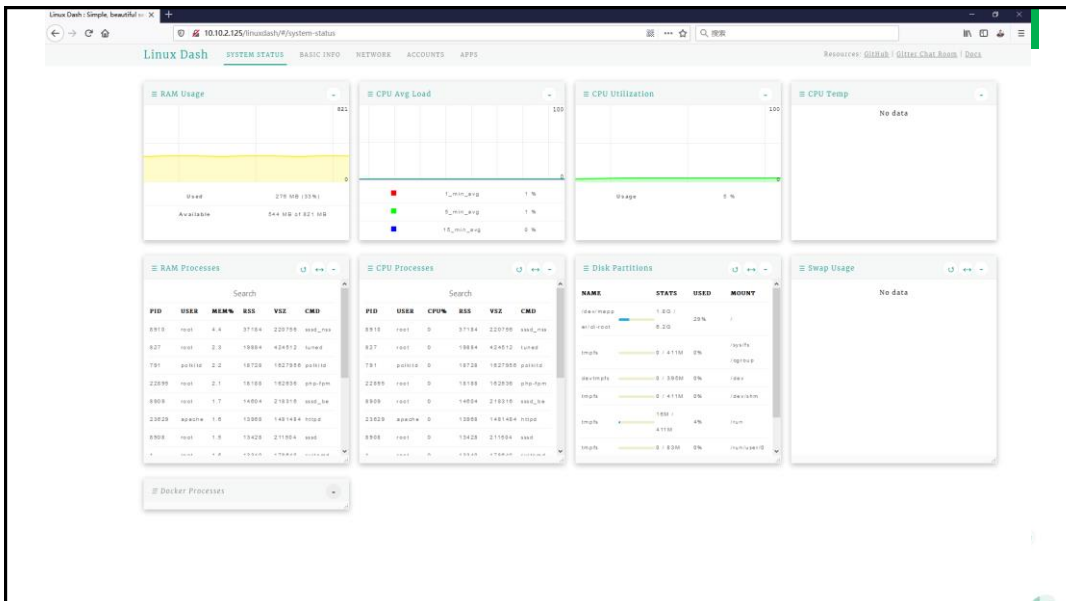


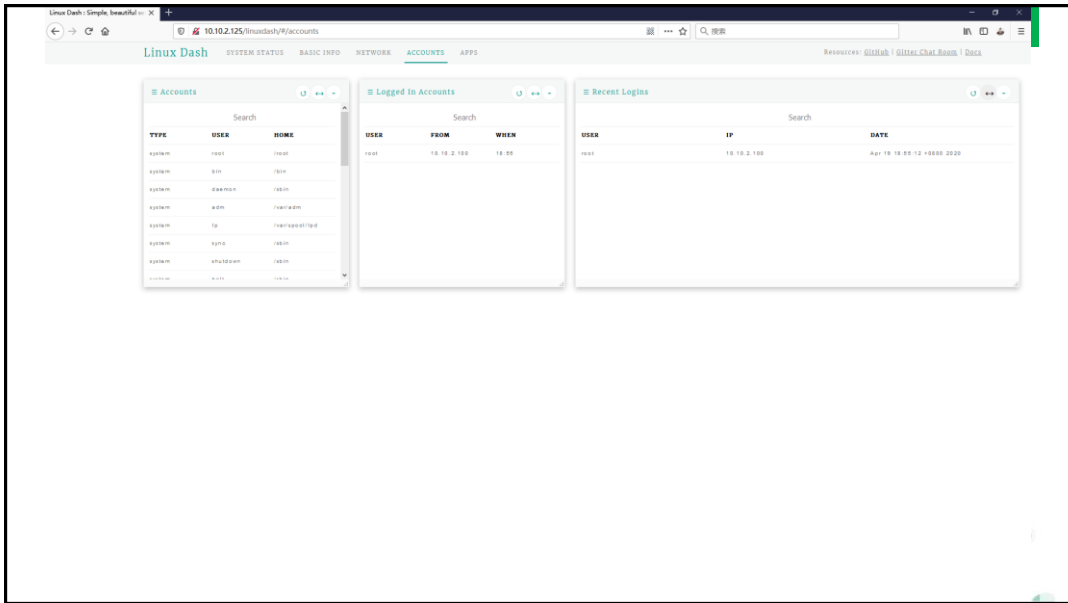
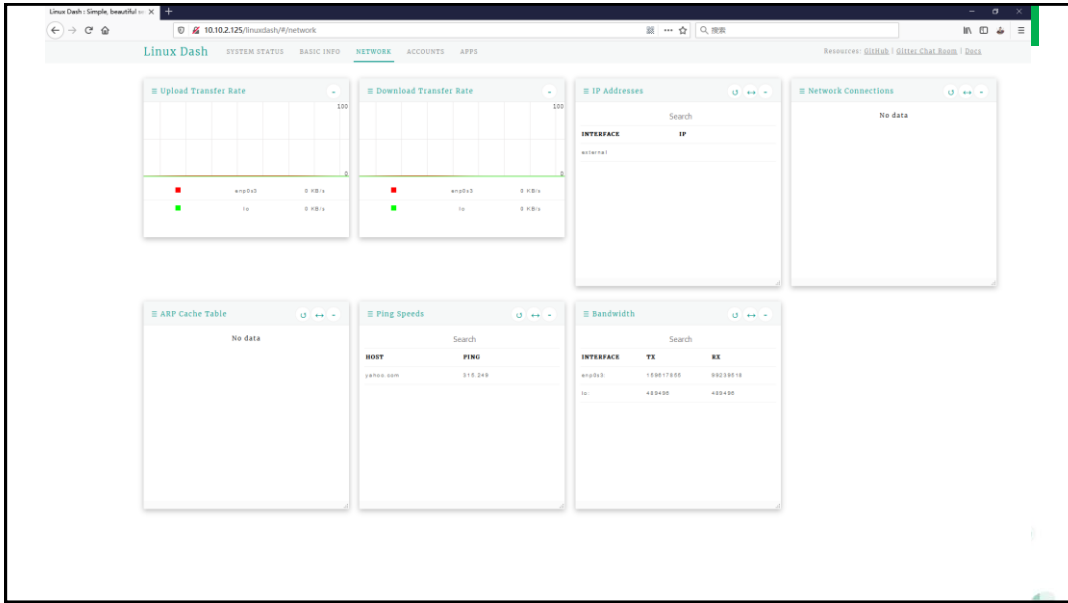
表 12-7-2 主机状态监控内容

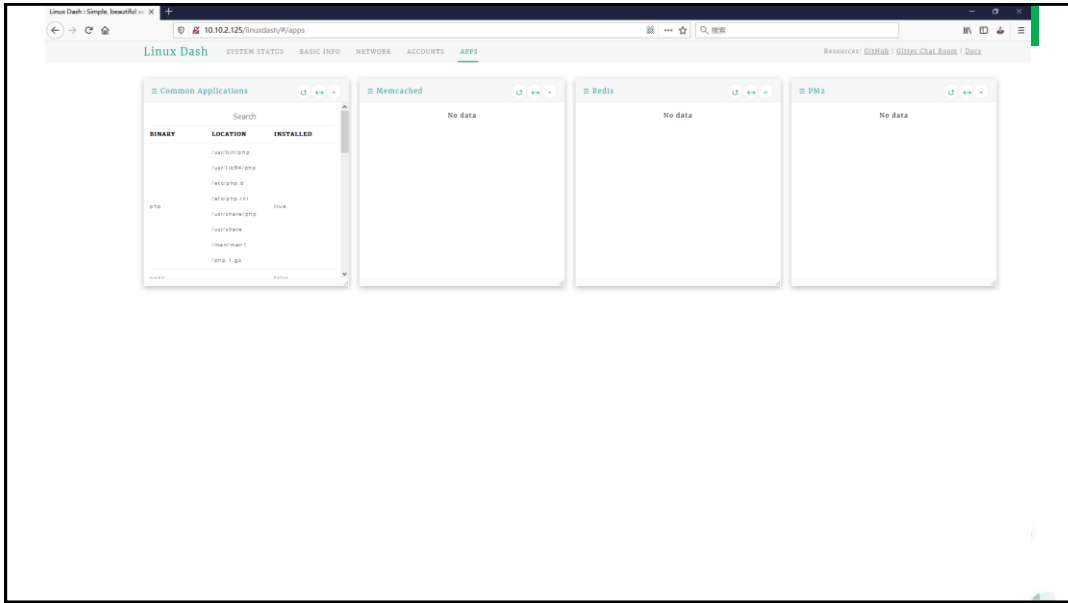
监控类型	监控内容	监控说明
RAM Usage 内存使用监控	Used	已使用的内存大小, 以及所占总内存比例, 该值包含了缓存和应用系统实际使用的内存大小
	Available	目前主机中还剩余可以被应用程序使用的物理内存大小
CPU Avg Load CPU 负载	1_min_avg	最近 1 分钟内平均 CPU 负载
	5_min_avg	最近 5 分钟内平均 CPU 负载
	15_min_avg	最近 15 分钟内平均 CPU 负载
CPU Utilization CPU 利用率	Usage	一段时间内 CPU 资源占用情况
Disk Partitions 磁盘分区	NAME	磁盘中文件系统分区名称
	STATS	文件系统磁盘使用状态
	USED	文件系统存储磁盘使用率
	MOUNT	文件系统分区挂载目录
RAM Processes 内存进程	PID	进程执行编号
	USER	进程执行属主
	MEM%	进程占用内存的百分比
	RSS	进程占用的固定内存量
	VSZ	进程占用的虚拟内存量
	CMD	进程执行命令的名称和参数
CPU Processes CPU 进程	PID	进程执行编号
	USER	进程执行属主
	CPU%	进程占用 CPU 运行执行的百分比
	RSS	进程占用的固定内存量
	VSZ	进程占用的虚拟内存量
	CMD	进程执行命令的名称和参数



The screenshot displays the Linux Dash dashboard with the following sections:

- General Info:** Hostname: Project-11-Team-01, OS: 4.18.0-147.5.1.el8_2_100_24, Server Time: Sun Apr 19 20:45:14 CDT 2020, Uptime: 7 hours 42 minutes and 4 seconds.
- Memory Info:** Active: 276432 kB, Active(anon): 84076 kB, Active(file): 191356 kB, AnonHugePages: 38896 kB, AnonPages: 114880 kB, Buffer: 0 kB, Buffers: 1984 kB, Cached: 493456 kB, Committed: 1260228 kB, Free: 1851128 kB.
- CPU Info:** Architecture: x86_64, SmpCPUs: 1624, SmpCores: 1610, SmpNodes: 1, CPU MHz: 2712.004, CPU Family: 0, CPU op-mode(s): 32-bit, 64-bit, CPU(s): 1, Core(s) per socket: 1.
- Scheduled Cron Jobs:** A table with columns: MIN, HRS, DAY, MONTH, WEDAY, USER, CMD. It shows cron jobs for root, root, root, root, root, root, root, root, root, root.
- Cron Job History:** No data.
- IO Stats:** A table with columns: DEVICE, READS, WRITES, IN_PROG, TIME. It shows IO statistics for sda, sda1, sda2, dm-0, dm-1.







命令指南 / 操作引导

104

```

1. [root@Project-11-Task-01 ~]# yum install httpd
2. [root@Project-11-Task-01 ~]# systemctl start httpd
3. [root@Project-12-Task-01 ~]# systemctl enable httpd.service
4. [root@Project-11-Task-01 ~]# systemctl is-enabled httpd
5. [root@Project-11-Task-01 ~]# yum module -y enable php:7.3
6. [root@Project-11-Task-01 ~]# yum install php
7. [root@Project-11-Task-01 ~]# php -v
8. [root@Project-11-Task-01 ~]# httpd -v
9. [root@Project-11-Task-01 ~]# seenforce 0
10. [root@Project-11-Task-01 ~]# sestatus
11. [root@Project-11-Task-01 ~]# firewall-cmd --permanent --add-rich-rule='rule family=ipv4 service name=http accept'
12. [root@Project-11-Task-01 ~]# firewall-cmd --reload
13. [root@Project-11-Task-01 ~]# firewall-cmd --list-service

14. [root@Project-11-Task-01 ~]# yum install wget
15. [root@Project-11-Task-01 ~]# yum install unzip
16. [root@Project-11-Task-01 ~]# wget https://github.com/afaqurk/linux-dash/archive/master.zip
17. [root@Project-11-Task-01 ~]# unzip -d /var/www/ master.zip
18. [root@Project-11-Task-01 ~]# chown -R apache:apache /var/www/linux-dash-master
19. [root@Project-11-Task-01 ~]# chmod -R 755 /var/www/linux-dash-master

20. [root@Project-11-Task-01 ~]# vi /etc/httpd/conf.d/linuxdash.conf
21. #/etc/httpd/cron.d/linuxdash.conf file info
22. Alias /linuxdash "/var/www/linux-dash-master/app"
23. <Directory "/var/www/linux-dash-master/app">
24.     AllowOverride None
25.     Options None
26.     Require ip 10.10.2.0/24
27. </Directory>
28. [root@Project-11-Task-01 ~]# systemctl reload httpd

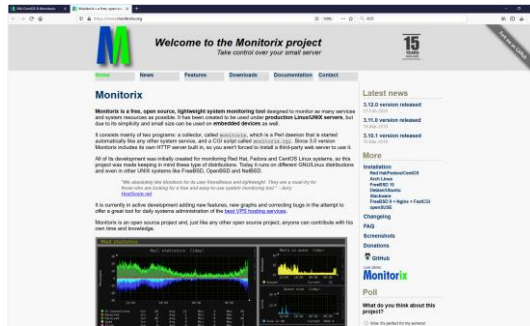
```



5. 使用Monitorix实现系统监控

5.1 Monitorix

- Monitorix is a free, open source, lightweight system monitoring tool designed to monitor as many services and system resources as possible.



河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

5. 使用Monitorix实现系统监控

5.2 任务2

任务2：使用Monitorix实现系统监控

步骤1：准备Monitorix部署所需的基本环境

步骤2：获取Monitorix程序

步骤3：安装配置并发布

步骤4：配置监控对象

步骤5：查看监控信息

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



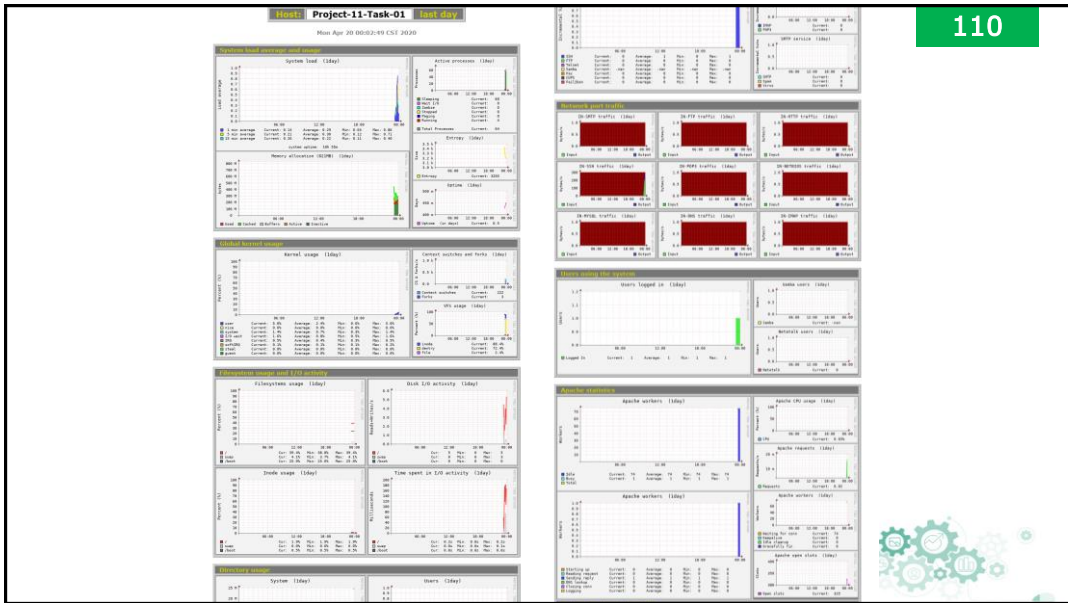
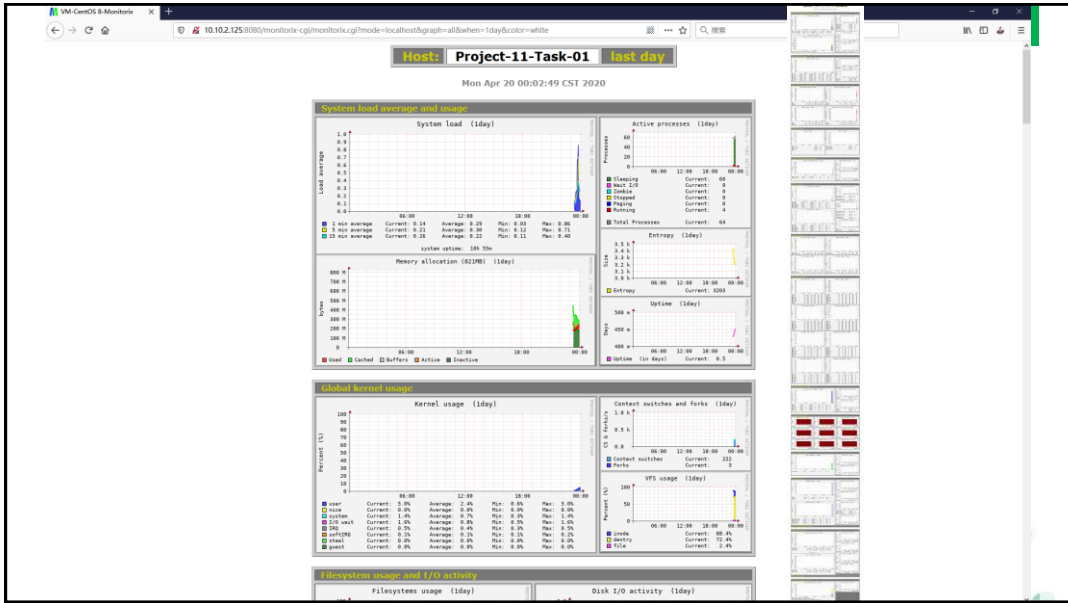
操作视频 / 现场演示

✓ 任务2：使用Monitorix实现系统监控

■ 任务目标：

- 本地主机通过浏览器访问Monitorix
- 实现对系统的全面监控
- 阅读Monitorix的监控信息





110





命令指南 / 操作引导

1. [root@Project-11-Task-01 ~]# yum install epel-release
2. [root@Project-11-Task-01 ~]# yum install rrdtool rrdtool-perl
3. [root@Project-11-Task-01 ~]# yum install perl-libwww-perl perl-MailTools
4. [root@Project-11-Task-01 ~]# yum install perl-CGI perl-DBI perl-XML-Simple
5. [root@Project-11-Task-01 ~]# yum install perl-Config-General perl-HTTP-Server-Simple
6. [root@Project-11-Task-01 ~]# yum install monitorix
7. [root@Project-11-Task-01 ~]# vi /etc/monitorix/monitorix.conf
8. [root@Project-11-Task-01 ~]# systemctl restart monitorix



6. 使用Cacti建设网络监控服务

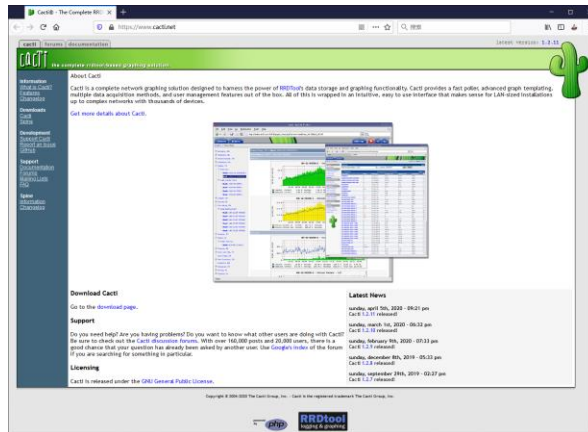
6.1 Cacti

- Cacti is a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality.
- Cacti provides a fast poller, advanced graph templating, multiple data acquisition methods, and user management features out of the box.
- All of this is wrapped in an intuitive, easy to use interface that makes sense for LAN-sized installations up to complex networks with thousands of devices.



6. 使用Cacti建设网络监控服务

6.1 Cacti



河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>

6. 使用Cacti建设网络监控服务

6.2 任务3

任务3：使用Cacti实现监控服务

- 步骤1：准备Cacti部署所需的基本环境
- 步骤2：部署Cacti并提供服务
- 步骤3：系统初始化配置
- 步骤4：添加受监控对象
- 步骤5：查看监控信息

河南中医药大学 / 阮晓龙 / 13938213680 / <https://internet.hactcm.edu.cn/> / <http://www.51xueweb.cn>



操作视频 / 现场演示

✓ 任务3：使用Cacti实现监控服务

■ 任务目标：

- 本地主机通过浏览器访问Cacti
- 实现对系统的全面监控
 - 监控CentOS
 - 监控Windows
- 阅读Cacti的监控信息



安装部署并发布服务





命令指南 / 操作引导

```

1. #-----了解系统基本信息并进行操作系统升级-----
2. [root@Project-12-Task-02 ~]# nmcli
3. [root@Project-12-Task-02 ~]# yum update -y

4. #-----安装Apache Http Server并完成配置-----
5. [root@Project-12-Task-02 ~]# yum install -y httpd
6. [root@Project-12-Task-02 ~]# systemctl start httpd
7. [root@Project-12-Task-02 ~]# systemctl enable httpd

8. #-----安装php， 并进行参数配置-----
9. [root@Project-12-Task-02 ~]# yum module list php
10. [root@Project-12-Task-02 ~]# yum module -y enable php:7.3
11. [root@Project-12-Task-02 ~]# yum install -y php
12. [root@Project-12-Task-02 ~]# vi /etc/php.ini
13. #*****配置文件/etc/php.ini修改的内容*****
14. date.timezone = "Asia/Shanghai"
15. max_execution_time = 60
16. memory_limit = 640M
17. #*****
18. [root@Project-12-Task-02 ~]# systemctl restart php-fpm

```



命令指南 / 操作引导

```

1. #-----安装MariaDB数据库管理系统并进行配置-----
2. [root@Project-12-Task-02 ~]# yum install -y mariadb-server
3. [root@Project-12-Task-02 ~]# cp /etc/my.cnf.d/mariadb-server.cnf /etc/mariadb-server.cnf.bak
4. [root@Project-12-Task-02 ~]# vi /etc/my.cnf.d/mariadb-server.cnf
5. #*****配置文件/etc/my.cnf.d/mariadb-server.cnf*****
6. character_set_server=utf8mb4
7. collation_server=utf8mb4_unicode_ci
8. init_connect='SET NAMES utf8mb4'
9. skip-character-set-client-handshake=true

10. join_buffer_size=64M
11. innodb_buffer_pool_size=640M
12. innodb_buffer_pool_instances=5
13. innodb_flush_log_at_timeout=3
14. innodb_read_io_threads=32
15. innodb_write_io_threads=16
16. innodb_io_capacity=5000
17. innodb_io_capacity_max=10000
18. innodb_file_format=Barracuda
19. innodb_large_prefix=1
20. #*****

21. [root@Project-12-Task-02 ~]# systemctl start mariadb
22. [root@Project-12-Task-02 ~]# systemctl enable mariadb
23. [root@Project-12-Task-02 ~]# mysqladmin -uroot password 'mariadb@centos#123'
24. [root@Project-12-Task-02 ~]# mysql_tzinfo_to_sql /usr/share/zoneinfo/Asia/Shanghai | mysql -u root -p mysql

```





命令指南 / 操作引导

```

1. #-----安装监控必须的数据存储工具RRDTool, 数据采集协议SNMP, 组件模块-----
2. [root@Project-12-Task-02 ~]# yum install -y wget patch tar
3. [root@Project-12-Task-02 ~]# yum install -y rrdtool net-snmp*
4. [root@Project-12-Task-02 ~]# yum install -y php php-common php-bcmath php-cli \
5.     php-mysqlnd php-gd php-gmp php-intl \
6.     php-json php-ldap php-mbstring \
7.     php-pdo php-pear php-snmp php-process \
8.     php-xml php-zip

9. #-----配置Firewall防火墙, 安装阶段关闭SELinux-----
10. [root@Project-12-Task-02 ~]# firewall-cmd --permanent --zone=public --add-port=80/tcp
11. [root@Project-12-Task-02 ~]# firewall-cmd --reload
12. [root@Project-12-Task-02 ~]# firewall-cmd --zone=public --list-all
13. [root@Project-12-Task-02 ~]# setenforce 0
14. [root@Project-12-Task-02 ~]# sestatus
15. [root@Project-12-Task-02 ~]# getenforce

```



命令指南 / 操作引导

```

1. #-----获取Cacti程序并进行部署和发布, 最新版本请查看官网cacti.net-----
2. [root@Project-12-Task-02 ~]# wget https://www.cacti.net/downloads/cacti-1.2.10.tar.gz
3. [root@Project-12-Task-02 ~]# tar -xzf cacti-1.2.10.tar.gz -C /var/www/
4. [root@Project-12-Task-02 ~]# mv /var/www/cacti-1.2.10 /var/www/cacti
5. [root@Project-12-Task-02 ~]# chown -R apache:apache /var/www/cacti
6. [root@Project-12-Task-02 ~]# chmod -R 755 /var/www/cacti
7. [root@Project-12-Task-02 ~]# chmod 777 /var/www/cacti/log/cacti.log

8. #-----创建Cacti所需的数据库, 并导入初始数据表-----
9. [root@Project-12-Task-02 ~]# mysql -uroot -pmariadb@centos#123
10. #*****在MariaDB中创建数据库db_cacti*****
11. MariaDB [(none)]> show databases;
12. MariaDB [(none)]> create database db_cacti;
13. MariaDB [(none)]> use db_cacti;
14. MariaDB [db_cacti]> source /var/www/cacti/cacti.sql;
15. MariaDB [(none)]> GRANT ALL ON db_cacti.* TO 'cactiroot'@'localhost' IDENTIFIED BY 'cacti@mariadb#123';
16. MariaDB [(none)]> GRANT SELECT ON mysql.time_zone_name TO 'cactiroot'@'localhost';
17. MariaDB [(none)]> flush privileges;
18. MariaDB [(none)]> exit
19. #*****

```





命令指南 / 操作引导

```

1. [root@Project-12-Task-02 ~]# vi /var/www/cacti/include/config.php
2. #*****cacti-1.2.1/include/config.php配置文件修改数据库信息*****
3. $database_type = 'mysql';
4. $database_default = 'db_cacti';
5. $database_hostname = 'localhost';
6. $database_username = 'cactiroot';
7. $database_password = 'cacti@mariaadb#123';
8. $database_port = '3306';
9. $database_retries = 5;
10. $database_ssl = false;
11. $database_ssl_key = '';
12. $database_ssl_cert = '';
13. $database_ssl_ca = '';
14. $url_path = '/';
15. #*****

```

```

16. #-----配置Apache Http Server，发布Cacti服务-----
17. [root@Project-12-Task-02 ~]# vi /etc/httpd/conf/httpd.conf
18. #*****/etc/httpd/conf/httpd.conf配置文件信息*****
19. DocumentRoot "/var/www/cacti"
20. <Directory "/var/www/cacti">
21.     AllowOverride None
22.     Require all granted
23. </Directory>
24. #*****

```

```

25. #-----创建网络与系统监控数据采集的任务计划-----
26. [root@Project-12-Task-02 ~]# echo '*/* * * * * root php /var/www/cacti/poller.php > /dev/null 2>&1' >> /etc/crontab

```



对安装部署的服务进行正确性验证





命令指南 / 操作引导

```

1. #-----冗余操作，进行服务重启和状态验证-----
2. [root@Project-12-Task-02 ~]# systemctl reload crond
3. [root@Project-12-Task-02 ~]# systemctl reload php-fpm
4. [root@Project-12-Task-02 ~]# systemctl reload httpd
5. [root@Project-12-Task-02 ~]# systemctl restart mariadb
6. [root@Project-12-Task-02 ~]# systemctl is-enabled httpd
7. [root@Project-12-Task-02 ~]# systemctl is-enabled mariadb
8. [root@Project-12-Task-02 ~]# systemctl status httpd
9. [root@Project-12-Task-02 ~]# systemctl status mariadb
10. [root@Project-12-Task-02 ~]# httpd -v
11. [root@Project-12-Task-02 ~]# php -v
12. [root@Project-12-Task-02 ~]# firewall-cmd --reload
13. [root@Project-12-Task-02 ~]# firewall-cmd --zone=public --list-all
14. [root@Project-12-Task-02 ~]# systemctl is-enabled firewalld
15. [root@Project-12-Task-02 ~]# systemctl status firewalld
16. [root@Project-12-Task-02 ~]# sestatus
17. [root@Project-12-Task-02 ~]# getenforce

18. #-----在本地主机使用浏览器访问http://vm-host-ip，根据提示进行安装-----
19. #-----cacti 1.2.10版本Bug：选择载入模板的页面中，取消所有选项，仅选择5个模板-----

```



初始化安装

(初始账号密码均为admin，首次使用需要修改密码)



实现对CentOS与Windows的监控

(配置监控后需要一段时间方可查看数据)



The screenshot displays the Zabbix configuration interface for a server named 'VM Project 02 Task 01 CentOS (10.10.2.138)'. The interface is divided into several sections:

- Host Information:** Shows the host name, IP address (10.10.2.138), and various system details.
- Configuration Parameters:** A table of key-value pairs for monitoring settings.
- Item List:** A table of monitoring items with their names, units, and current values.

Item Name	Unit	Value
1) Server Linux CPU Load	0.000000	0.000000
2) Server Linux CPU Usage	0.000000	0.000000
3) Server Linux Disk I/O	0.000000	0.000000
4) Server Linux Memory Physical Memory	0.000000	0.000000
5) Server Linux Memory Swap Space	0.000000	0.000000
6) Server Linux Memory Virtual Memory	0.000000	0.000000
7) Server Linux Time Ping Time	0.000000	0.000000
8) Server Linux Time Pinging Time	0.000000	0.000000
9) Server Linux Time System	0.000000	0.000000

Main Console

VM Management (4 VMs)

VM	名称	IP	OS	状态	CPU	内存	磁盘	网络	快照	操作
VM-Project-01	VM-Project-01	192.168.1.10	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-02	VM-Project-02	192.168.1.11	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-03	VM-Project-03	192.168.1.12	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-04	VM-Project-04	192.168.1.13	Windows	Running	100%	100%	100%	100%	1	更多操作

Main Console

VM Management (4 VMs)

VM	名称	IP	OS	状态	CPU	内存	磁盘	网络	快照	操作
VM-Project-01	VM-Project-01	192.168.1.10	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-02	VM-Project-02	192.168.1.11	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-03	VM-Project-03	192.168.1.12	Windows	Running	100%	100%	100%	100%	1	更多操作
VM-Project-04	VM-Project-04	192.168.1.13	Windows	Running	100%	100%	100%	100%	1	更多操作

