

性能测试

--Linux 服务器监控

内容回顾

■ Controller介绍

- 作用：设计场景、运行场景并监控场景运行
- 主要菜单的功能

■ 场景设计

- 目标场景（验收测试，有非常明确的性能指标，结果是达到或达不到）
- **Group**:两个以上业务相互独立，用户量不成正比

内容回顾

- 百分比模式：模拟场景中用户分不同比例做不同的业务操作
- Scenario —Basic schedule：收集不同用户负载量下，各项性能指标
- Scenario—Real-world schedule
 - 做峰值测试
 - 模拟真实场景（拖拽的方式来模拟多少个用户启动，多少个用户停止）

■ 注意事项

- Run-Time Settings中Run Logic设置的迭代次数与Run Mode 中Duration设置持续时间，优先级问题

内容回顾

- 设置Run Mode后脚本执行过程
- 多机负载配置
 - 安装Load Generator模块
 - 开启LoadRunner Agent Runtime Settings Configuration
 - 配置Load Genrators
- 资源监控
 - CPU (Processor time)

内容回顾

- 内存（可用内存，内存错误量）
- 硬盘（读写速度，队列长度）
- 网络带宽
- 数据库计数器

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- Linux监控工具nmon

Linux服务器监控概述

■ 服务器端操作系统

- Windows
- Linux

■ 如果是Linux操作系统，怎样收集资源占用情况数据？

服务器资源监控

■ 测试范围及性能指标

- CPU
- 内存
- 硬盘
- 网络

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- linux监控工具nmon

服务器资源监控--top

■ top

- 能够实时监控系统的运行状态，并且可以按照CPU及内存使用量进行排序

```

KiB Mem : 10161264 total, 499364 free, 49524 used, 497248 buff/cache
KiB Swap: 1046524 total, 1046524 free, 0 used. 802132 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM     TIME+ COMMAND
 15745 edu       20   0   42224 3592 3056 R   0.3   0.4   0:00.02 top
    1 root      20   0   37816 6008 4128 S   0.0   0.6   0:03.05 systemd
    2 root      20   0      0  0  0 S   0.0   0.0   0:00.00 kthreadd
    3 root      20   0      0  0  0 S   0.0   0.0   0:00.00 ksoftirqd/0
    5 root      0 -20      0  0  0 S   0.0   0.0   0:00.00 kworker/0:0H
    7 root      20   0      0  0  0 S   0.0   0.0   0:03.68 rcu_sched
    8 root      20   0      0  0  0 S   0.0   0.0   0:00.00 rcu_bh
    9 root      rt    0      0  0  0 S   0.0   0.0   0:00.00 migration/0
   10 root      rt    0      0  0  0 S   0.0   0.0   0:00.15 watchdog/0
   11 root      20   0      0  0  0 S   0.0   0.0   0:00.00 kdevtmpfs
   12 root      0 -20      0  0  0 S   0.0   0.0   0:00.00 netns
   13 root      0 -20      0  0  0 S   0.0   0.0   0:00.00 perf
  
```

用户名 ↑
 nice值。负值表示高优先级，正值表示低优先级
 CPU使用率 ↑
 命令名称 ↑
 进程ID ↑
 优先级 ↑
 进程使用的虚拟内存总量 ↑
 共享内存 ↑
 内存使用率 ↑
 进程使用CPU时间总计 ↑
 进程使用的、未被换出的物理内存大小

服务器资源监控--top

- top结果中常用交互命令（注意大写）
 - 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/累计时间进行排序

服务器资源监控--top

■ top -s 累积模式查看

```
edu@software:~$ top -s
top - 16:01:38 up 6:21, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 102 total, 1 running, 101 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.3 sy, 0.0 ni, 99.0 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
KiB Mem : 1016136 total, 469092 free, 49756 used, 497288 buff/cache
KiB Swap: 1046524 total, 1046524 free, 0 used. 801892 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1513	root	20	0	0	0	0	S	0.3	0.0	0:07.62	kworker/0:2
1	root	20	0	37816	6008	4128	S	0.0	0.6	0:03.06	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.55	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
7	root	20	0	0	0	0	S	0.0	0.0	0:03.95	rcu_sched

服务器资源监控--top

■ top -s 补充说明

- **load average**: 系统的运行队列的平均利用率, 也可以认为是可运行进程的平均数。三个值分别表示在最后的1分钟、5分钟、15分钟的平均负载值
- **top使用说明**: 在单核CPU中load average的值为1时, 表示满负荷状态。多核CPU中满负荷load average的值为 $1 * \text{CPU核数}$

服务器资源监控--top

- **CPU(s):** 用户空间占用CPU百分比
- **sy:**内核空间占用CPU百分比
- **ni:** 用户进程空间内改变过优先级的进程占用CPU百分比
- **id:** 空闲CPU百分比
- **wa:** 等待输入输出的CPU时间百分比
- **hi:**硬件中断
- **si:**软件中断
- **st:**实时

服务器资源监控--top

■ memory:内存

- total: 物理内存总量
- used: 使用的物理内存总量
- free: 空闲内存总量
- buffers: 用作内核缓存的内存量
- Swap: 交换区总量
- used: 使用的交换区总量
- free: 空闲交换区总量
- cached: 缓冲的交换区总量

服务器资源监控----vmstat

■ vmstat

- 展现给定时间间隔的服务器的状态值,包括服务器的CPU使用率, 内存使用, 虚拟内存交换情况,I/O读写情况
- 一般使用两个数字参数来完成的, 第一个参数是采样的时间间隔数, 单位是秒, 第二个参数是采样的次数
- 如: `vmstat 2 1` (2表示每隔两秒采集一次服务器状态, 1表示只采集一次)

服务器资源监控----vmstat

- **r** 表示运行队列(就是说多少个进程真的分配到CPU) 如果运行队列过大，表示你的CPU很繁忙，一般会造成CPU使用率很高。
- **b** 表示阻塞的进程

服务器资源监控----vmstat

- swpd 虚拟内存已使用的大小，如果大于0，表示你的机器物理内存不足了，如果不是程序内存泄露的原因，那么该升级内存了或者把耗内存的任务迁移到其他机器
- free 空闲的物理内存的大小

■ buff 缓存

服务器资源监控----vmstat

- **cache** 用来记忆我们打开的文件,给文件做缓冲, (这里是Linux/Unix的聪明之处, 把空闲的物理内存的一部分拿来做文件和目录的缓存, 是为了提高 程序执行的性能, 当程序使用内存时, **buffer/cached**会很快地被使用)
- **si** 每秒从磁盘读入虚拟内存的大小, 如果这个值大于0, 表示物理内存不够用或者内存泄露了, 要查找耗内存进程解决掉

服务器资源监控----vmstat

- **so** 每秒虚拟内存写入磁盘的大小，如果这个值大于0，同上
- **bi** 块设备每秒接收的块数量，这里的块设备是指系统上所有的磁盘和其他块设备，默认块大小是1024byte(I/O操作时查看读写速度)
- **bo** 块设备每秒发送的块数量，例如我们读取文件，bo就要大于0。
bi和bo一般都要接近0，不然就是I/O过于频繁，需要调整
- **in** 每秒CPU的中断次数，包括时间中断

服务器资源监控----vmstat

- **cs** 每秒上下文切换次数，例如我们调用系统函数，就要进行上下文切换，线程的切换，也要进程上下文切换，这个值要越小越好，太大了，要考虑调低线程或者进程的数目
- **us** 用户CPU时间
- **sy** 系统CPU时间，如果太高，表示系统调用时间长，例如是IO操作频繁

服务器资源监控----vmstat

- **id** 空闲 CPU时间，一般来说， $id + us + sy = 100$ ，一般id是空闲 CPU使用率，us是用户CPU使用率，sy是系统CPU使用率
- **wt** CPU等待I/O 的时间

服务器资源监控----vmstat

■ 实际使用中，有些场景会在一段时间内一直监控

— 使用命令：vmstat 2

■ 不监控时，直接结束此命令即可

```
edu@software:~$ vmstat 2
```

procs		memory				swap		io		system		cpu				
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
0	0	0	469380	18696	478620	0	0	17	20	47	100	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	53	104	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	47	92	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	49	94	0	1	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	47	92	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	50	94	0	0	100	0	0

top VS vmstat

- 两者都可以监控服务器资源使用情况，但监控的点不同

top VS vmstat

```
edu@software:~$ top -s
top - 16:01:38 up 6:21, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 102 total, 1 running, 101 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.3 sy, 0.0 ni, 99.0 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
KiB Mem : 1016136 total, 469092 free, 49756 used, 497288 buff/cache
KiB Swap: 1046524 total, 1046524 free, 0 used. 801892 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1513	root	20	0	0	0	0	S	0.3	0.0	0:07.62	kworker/0:2
1	root	20	0	37816	6008	4128	S	0.0	0.6	0:03.06	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.55	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
7	root	20	0	0	0	0	S	0.0	0.0	0:03.95	rcu_sched

```
edu@software:~$ vmstat 2
```

procs		memory				swap		io		system		cpu				
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
0	0	0	469380	18696	478620	0	0	17	20	47	100	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	53	104	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	47	92	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	49	94	0	1	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	47	92	0	0	100	0	0
0	0	0	469364	18696	478620	0	0	0	0	50	94	0	0	100	0	0

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- linux监控工具nmon

服务器CPU监控--mpstat

■ mpstat

- 安装： `sudo apt-get install mpstat` (Ubuntu 15及以下版本)
- 安装： `sudo apt install sysstat` (Ubuntu 16及以上版本)
- 可以查看多核CPU中每个计算核心的统计数据
- `-P` (大写) 表示监控哪个CPU，CPU在 **【0，CPU个数-1】** 中取值
- `internal`相邻的两次采样的间隔时间
- `count`采样的次数
- 如： `mpstat -P 0 2 3` (监控第一个CPU，每隔2s采集一次数据，共3次)

服务器CPU监控--mpstat

```
edu@software:~$ mpstat
Linux 4.4.0-87-generic (software)          2019年03月12日  _x86_64_          (1 CPU)

16时44分50秒  CPU      %usr    %nice    %sys  %iowait    %irq    %soft    %steal    %guest    %gnice   %idle
16时44分50秒  all      0.15    0.03    0.14    0.07    0.00    0.01    0.00    0.00    0.00    99.59
```

```
edu@software:~$ mpstat -P ALL 2 3
Linux 4.4.0-87-generic (software)          2019年03月12日  _x86_64_          (1 CPU)

16时45分32秒  CPU      %usr    %nice    %sys  %iowait    %irq    %soft    %steal    %guest    %gnice   %idle
16时45分34秒  all      0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00   100.00
16时45分34秒    0      0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00   100.00

16时45分34秒  CPU      %usr    %nice    %sys  %iowait    %irq    %soft    %steal    %guest    %gnice   %idle
16时45分36秒  all      0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    99.00
16时45分36秒    0      0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    99.00

16时45分36秒  CPU      %usr    %nice    %sys  %iowait    %irq    %soft    %steal    %guest    %gnice   %idle
16时45分38秒  all      0.00    0.00    0.00    3.00    0.00    0.00    0.00    0.00    0.00    97.00
16时45分38秒    0      0.00    0.00    0.00    3.00    0.00    0.00    0.00    0.00    0.00    97.00

Average:      CPU      %usr    %nice    %sys  %iowait    %irq    %soft    %steal    %guest    %gnice   %idle
Average:      all      0.00    0.00    0.33    1.00    0.00    0.00    0.00    0.00    0.00    98.67
Average:      0      0.00    0.00    0.33    1.00    0.00    0.00    0.00    0.00    0.00    98.67
```

服务器CPU监控--mpstat

■ mpstat注意

- 当没有参数时，mpstat则显示系统启动以后所有信息的平均值。有interval时，第一行的信息自系统启动以来的平均信息。从第二行开始，输出为前一个interval时间段的平均信息

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- linux监控工具nmon

服务器内存监控—free

■ free

- 监控系统的内存使用状态
- **total**: 总计物理内存的大小
- **used**: 已使用多大
- **free**: 可用有多少
- **shared**: 多个进程共享的内存总额
- **buffer/cached**磁盘缓存的大小

■ 常用选项

- **-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

服务器内存监控—free

```
edu@software:~$ free -m
```

	total	used	free	shared	buff/cache	available
Mem:	992	48	458	4	485	783
Swap:	1021	0	1021			

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- linux监控工具nmon

服务器网络监控---netstat

■ netstat

- 用于显示本机网络连接、运行端口、路由表等信息
- -s 显示网络统计信息
- -l: 仅列出有在Listen（监听）的服务状态
- -p: 显示建立相关链接的程序名
- -t (tcp) : 显示tcp相关选项
- -u (udp) : 显示udp相关选项
- -c: 每隔一个固定时间，执行该netstat命令

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- linux监控工具nmon

服务器磁盘监控---iostat

■ iostat

- 是对系统磁盘IO操作进行监控，它的输出主要显示磁盘的读写操作的统计信息，同时给出CPU使用情况
- `iostat -x` （指定设备名称）
- `iostat -x sda 1 3`(每隔1s种采集一次数据，共3次)

目录

- Linux服务器监控概述
- 服务器资源监控
- 服务器CPU监控
- 服务器内存监控
- 服务器网络监控
- 服务器磁盘监控
- **linux监控工具nmon**

linux 监控工具nmon

- nmon是一种各种Linux操作系统上广泛使用的监控与分析工具，相对于其它一些系统资源监控工具来说，nmon所记录的信息是比较全面的，它能在系统运行过程中实时地捕捉系统资源的使用情况，并且能输出结果到文件中，然后通过nmon_analyzer工具产生数据文件与图形化结果

linux 监控工具nmon

■ nmon安装：

- `sudo apt-get install nmon` (`sudo apt install nmon`(高版本))

■ nmon主要参数：

- `-f` 输出文件，必选参数，nmon输出的文件名是默认名称

`localhost_date_time.nmon`

- `-F <filename>` 这个参数和`-f`相同，只不过用户可以自己定义文件名称

linux 监控工具nmon

- **-s**采集数据频率，也就是保存数据的频率
- **-c** 采集数据次数
- **-t** 输出最消耗资源的进程数据
- **-m** 生成的数据文件的存放目录

■ 使用举例

- **nmon -s 2 -c 3 -m /home/liu -F test.nmon**

Linux 与 Windows 操作系统互传文件

1 两台主机能ping通；Linux主机有SSH服务，并且打开22端口；

- （如果防火墙打开，设置22 端口不被拦截，`$sudo ufw allow 22 sudo ufw allow ssh`）

2 Linux 传 Windows :

- 从linux向windows传文件：
- 在windows的cmd中输入 `pscp 用户名@: linux中源文件路径 windows中目的路径`

Linux 与 Windows 操作系统互传文件

- 如： `pscp root@192.168.0.100:/home/aaa.jpg G:/putty/`
- 注意：使用pscp只能传文件，无法传目录，要传目录的话可以用
`psftp`

■ 3 Windows 向 Linux传文件：

- 打开cmd窗口，输入命令：
`pscp 源文件路径 root@hostIp:目的路径`
- 如将aaa.jpg传到/home下，则使用如下命令：
`pscp aaa.jpg root@192.168.0.101:/home`

linux 监控工具nmon

■ nmon analyser Sheet

- **SYS_SUMM** 系统汇总,蓝线为cpu占有率变化情况,粉线为磁盘IO的变化情况
- **AAA** 关于操作系统以及nmon本身的一些信息
- **CPU_ALL** 所有CPU概述, 显示所有CPU平均占用情况
- **CPU_SUMM** 每一个CPU在执行时间内的占用情况

linux 监控工具nmon

■ nmon analyser Sheet

- **DISKBUSY** 磁盘组每个disk设备平均占用情况
- **DISKREAD** 每个磁盘组的平均读情况
- **DISKSIZE** 每个磁盘组的平均读写情况（块大小）
- **DISKWRITE** 每个磁盘组的平均写情况
- **DISKXFER** 每个磁盘组的I/O每秒操作
- **MEM** 内存相关的主要信息，使用、空闲内存大小等

内容总结

■ Linux服务器监控概述

■ 服务器资源监控

- vmstat

- top 命令的使用

■ 服务器CPU监控

- vmstat

内容总结

■ 服务器内存监控

- free

■ 服务器网络监控

- netstat

■ 服务器磁盘监控

- iostat

内容总结

■ linux监控工具nmon

- 安装
- **nmon**
- 输出到文件
- 分析工具分析（**nmon analyser Sheet**）



Question
