硬盘io系能测试工具汇总

io测试工具

1.windows io测试工具

HD Tune

HD Tune 是一款小巧易用的硬盘工具软件，其主要功能有硬盘传输速率检测，健康状态检测，温度检测及 HD Tune 5.00

磁盘表面扫描存取时间、CPU占用率。另外，还能检测出硬盘的固件版本、序列号、容量、缓存大小以及当前的Ultra DMA

模式等。虽然这些功能其它软件也有，但难能可贵的是此软件把所有这些功能积于一身，而且非常小巧，速度又快，更重

要的是它是免费软件，可自由使用。

SANergy 充分利用SAN的潜力，测试盘阵

使用Tivoli SANergy，客户可以有效地集中他们的存储资源，已达到减少管理费用，提高系统系能和增加投资回报率（ROI）的

目的。Tivoli的SANergy使用户能够通过实施存储区域网络(SAN)透明的共享共用存储器，卷和文件的存取；可以使用UNIX

&reg；windows NT&reg；和Apple Macintosh 系统再卷，文件和字节级上跨平台共享存储资源。

共享的方式大大的增加吞吐量，降低管理费用。通过合并存储空间和消除多主机环境中常见的复制数据而获得的高性能共享式存储

环境，能显著降IT成本。

2.linux 下服务器测试工具

vps测试工具

unixbench 是一款不错的linux下的vps性能测试软件。unixbeach的目的是给类unix系统的性能提供一种基本的指标；因此，多个测试被用于测试系统性能的各种各样的方面。

然后将这些测试结果与从基础系统的分数比较，产生一个预定的值，这个预定值比原始的分数更容易处理。综合整套系列的预定值将得出一个系统全面的索引。unixbench包含了一些

非常简单的图形测试，实现了测量系统的2D和3D图形性能。

unixbeach也能处理多处理器系统。如果您的系统有多个cpus，缺省行为是两次进行选择的测试--一次雨每个测试程序的一个拷贝每次跑，和N一次复制，其中N是cpus的数量。

这些设计使您能评价：当运行唯一任务时您系统的性能；多任务时您系统的性能；实施并行处理后你的系统性能的提高。

请注意unixbeach是一个系统基准测试工具，不是cpu，ram或者磁盘基准测试工具。结果不仅依靠您的硬件，还要依靠您的操作系统，库甚至编译器。

vps（virtual private server 虚拟专用服务器）技术，将一部服务器分割成多个虚拟专享服务器的优质服务。每个vps都可以分配独立的公网ip地址，

独立操作系统，独立超大空间，独立内存，独立cpu资源，独立执行程序和独立系统配置。用户除了可以分配多个虚拟主机及无限企业邮箱外，更具有独立

服务器功能，可自行安装程序，单独重启服务器。高端虚拟主机用户的最佳选择。虚拟专用服务器确保所有资源为用户独享，给用户最高的服务品质保证，让用户

以虚拟主机的价格享受到独立主机的服务品质。

linux下io性能测试工具

http://cwind.blog.51cto.com/62582/809284

以下分别用dd/orion/iozone/bonnie++四种工具对磁盘io进行了测试

首先用自带dd命令先测一下，块大小为8k

dd只能提供一个大概的测试结果，而且是连续IO而不是随机IO

读测试

#time dd if=/dev/sda2 of=/dev/null bs=8k count=8388608

8388608+0 records in

8388608+0 records out

68719476736 bytes (69 GB) copied, 516.547 seconds, 133 MB/s

real 8m36.926s

user 0m0.117s

sys 0m55.216s

写测试

#time dd if=/dev/zero of=/opt/iotest bs=8k count=8388608

8388608+0 records in

8388608+0 records out

68719476736 bytes (69 GB) copied, 888.398 seconds, 77.4 MB/s

real 14m48.743s

user 0m3.678s

sys 2m47.158s

读写测试

#time dd if=/dev/sda2 of=/opt/iotest bs=8k count=8388608

8388608+0 records in

8388608+0 records out

68719476736 bytes (69 GB) copied, 1869.89 seconds, 36.8 MB/s

real 31m10.343s

user 0m2.613s

sys 3m25.548s

接下来用oracle的orion工具来测一下

解压即可使用

#gzip orion\_linux\_x86-64.gz

测异步IO时需要libaio库

#export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/lib64

#echo $LD\_LIBRARY\_PATH

:/opt/oracle/product/10.2.0/lib:/usr/lib64

创建配置文件mytest.lun,列出要测试的分区即可。注意文件明前缀要跟下面的testname一致

#vi mytest.lun

查看mytest.lun

/dev/sda2

先来个simple test

# ./orion\_linux\_x86-64 -run simple -testname mytest -num\_disk 8

查看测试结果

#cat mytest\_20081111\_1431\_summary.txt

ORION VERSION 11.1.0.7.0

commandline:

-run simple -testname mytest -num\_disk 8

This maps to this test:

Test:mytest

Small IO size:8kB

Large IO size:1024kB

IO Types:Small Random IOs, Large Random IOs

Simulated Array Type:CONCAT

Write:0%

Cache Size:Not Entered

Duration for each Data Point:60 seconds

Small Columns:,0

Large Columns:,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16

Total Data Points:38

Name:/dev/sda2 Size:629143441920

1 FILEs found.

Maximum Large MBPS=56.97 @Small=0 and Large=7

Maximum Small IOPS=442 @Small=40 and Large=0

Minimum Small Latency=14.62 @Small=1 and Large=0

最大MBPS位56.97，最大IOPS位442

再测试一下8k随机读操作

# ./orion\_linux\_x86-64 -run advanced -testname mytest -num\_disks 8 -size\_small 8 -size\_large 8 -type rand&

查看结果

#cat mytest\_20081111\_1519\_summary.txt

ORION VERSION 11.1.0.7.0

commandline:

-run advanced -testname mytest -num\_disks 8 -size\_small 8 -size\_large 8 -type rand

This maps to this test:

Test:mytest

Small IO size:8KB

Large IO size:8KB

IO Types:Small Random IOs,Large Random IOs

Simulated Array Type:CONCAT

Write:%

Cache Size:Not Entered

Duration for each Data Point:60 sedonds

Small Columns:, 0

Large Columns:, 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16

Total Data Points:38

Name:/dev/sda2 Size:629143441920

1 FILEs found.

Maximum Large MBPS=3.21 @ Small=0 and Large=13

Maximum Small IOPS=448 @Small=38 and Large=0

Minimum Small Latency=15.16 @Small=1 and Large=0

最大MBPS为3.21（这么低？？），最大IOPS位448

再测试一下1M顺序读操作，失败了，原因不明...

#./orion\_linux\_x86-64 -run advanced -testname mytest -num\_disk 8 -size\_small 1024 -size\_large 1024 -type seq

ORION:ORacle IO Numbers --Version 11.1.0.7.0

mytest\_20081114\_1349

Test will take approximately 73 minutes

Larger caches may take longer

rwbase\_run\_test:rwbase\_reap\_req failed

rwbase\_run\_process:rwbase\_run\_test failedd

rwbase\_rwluns:rwbase\_run\_process failed

orion\_warm\_cache:Warming cache failed.Continuing

看看结果

#cat mytest\_20081111\_1620\_summary.txt

ORION VERSION 11.1.0.7.0

Commandline:

-run advanced -testname mytest -num\_disk 8 -size\_small 1024 -size\_large 1024 -type seq

This maps to this test:

Test:mytest

Small IO size:1024KB

Large IO size:1024KB

IO Types:Small Random IOs,Large Sequential Streams

Number of Concurrent IOs Per Stream:4

Force streams to separate disks:No

Simulated Array Type:CONCAT

Write:0%

Cache Size:Not Entered

Duration for each Data Point:60 seconds

没结果，失败

3.用iozone来测试一下

安装

#tar -xvf iozone3\_345.tar

#make linux-AMD64

指定64G的文件，只测read/reread 和write/rewrite,记录大小从4k-16k，同时生成一个excel文件iozone.wks

#./iozone -Rab iozone.wks -s64G -i 0 -i 1 -y 4k -q 16k

Iozone:Performance Test of File I/O

Version $Revision:3.345$

Compiled for 64bit mode

Build:linux-AMD64

Contributors:William Norcott,Don Capps,Isom Crawford,Kirby Collins

Al Slater,Scott Rhine,Mike Wisner,Ken Goss

Steve Landherr,Brad Smith,Mark Kelly,Dr.Alain CYR,

Randy Dunlap,Mark Montague,Dan Million,Gavin Brebner,

Jean=Marc Zucconi,Jeff Blomberg,Benny Halevy,Dave Boone,

Erik Habbinga,Kris Strecker,Walter Wong,Joshua Root,

Fabrice Bacchella,Zhenghua Xue,Qin Li.

Run began:Tue Nov 11 10:23:25 2008

Excel chart generation enabled

Auto Mode

File size set to 67108864KB

Using Minimum Record Size 4KB

Using Maximum Recourd Size 16KB

Command line used:./iozone -Rab iozone.wks -s64G -i 0 -i 1 -y 4k -q 16k

Output is in Kbytes/secc

Time Resolution=0.000001 seconds.

Processor cache size set to 1024Kbytes.

Processor cache line size set to 32 bytes.

File stride size set to 17\* record size

random random bkwd record stride

KB reclen write rewrite read reread read write read rewrite read fwrite frewrite fread freread

67108864 4 72882 69470 104898 125512

67108864 8 72083 69256 133689 109061

67108864 16 73375 69155 142019 116034

iozone test complete.

Excel output is below:

"Write report"

"4" "8" "16"

"67108864" 72882 72083 73375

"Re-writer report"

"4""8""16"

"67108864" 104898 133689 142019

"Re-Reader report"

"4" "8" "16"

"67108864" 125512 109061 116034

可以看到，8k的写是72M/s左右，读是133M/s左右，跟dd的结果比较接近

测试一下64G文件8k随机读写

#./iozone -Rab iozone.wks -s64G -i 2 -y 8k -q 8k

Iozone:Performance Test of File I/O

Version $Revision:3.345$

compiled for 64bit mode.

Build:linux-AMD64

Contributors:William Norcott,Don Capps,Isom Crawford,Kirby Collins

Al Slater,Scott Rhine,Mike Wisner,Ken Goss

Steve Landherr,Brad Smith,Mark Kelly,Dr.Alain CYR,

Randy Dunlap,Mark Montague,Dan Million,Gavin Brebner,

Jean-Marc Zuccone,Jeff Blomberg,Benny Halevy,Dave Boone,

Erik Habbinga,Kris Strecker,Walter Wong,joshua Root,

Fabrice Bacchella,Zhenghua Xue,Qin Li.

Run began:Fri Nov 14 15:52:01 2008

Excel chart generation enabled

Auto Mode

File size set to 67108864 KB

Using Minimum Record Size 8 KB

Using Maximum Record Size 8 KB

Command line used: ./iozone -Rab iozone.wks -s64G -i 2 -y 8k -q 8k

Output is in Kbytes/sec

Time Resolution = 0.000001 seconds.

Processor cache size set to 1024 Kbytes.

Processor cache line size set to 32 bytes.

File stride size set to 17 \* record size.

random random bkwd record stride

KB reclen write rewrite read reread read write read rewrite read fwrite frewrite fread freread

67108864 8

Error reading block at 6501007360

read: Success

出错了(??)

最后用bonnie++测一下

安装

#export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/lib64

#./configure

#make

#make install

开始测试，默认文件大小是内存2倍

#bonnie++ -d /opt/IOTest/ -m sva17 -u root

Using uid:0,gid:0

Writing with putc()...done

Writing intelligently...done

Rewriting...done

Reading with getc()...done

Reading intelligently...done

start 'em...done...done...done...

Create files in sequential order...done.

Stat files in sequential order...done.

Delete files in sequential order...done.

Create files in random order...done.

Stat files in random order...done.

Delete files in random order...done.

Version 1.03e ------Sequential Output------ --Sequential Input- --Random-

-Per Chr- --Block-- -Rewrite- -Per Chr- --Block-- --Seeks--

Machine Size K/sec %CP K/sec %CP K/sec %CP K/sec %CP K/sec %CP /sec %CP

sva17 63G 52391 84 35222 7 34323 6 56362 88 131568 10 176.7 0

------Sequential Create------ --------Random Create--------

-Create-- --Read--- -Delete-- -Create-- --Read--- -Delete--

files /sec %CP /sec %CP /sec %CP /sec %CP /sec %CP /sec %CP

16 +++++ +++ +++++ +++ +++++ +++ +++++ +++ +++++ +++ +++++ +++

sva17,63G,52391,84,35222,7,34323,6,56362,88,131568,10,176.7,0,16,+++++,+++,+++++,+++,+++++,+++,+++++,+++,+++++,+++,+++++,+++

结论：不同测试工具构建出来的测试环境不同，侧重点也不一样，得到的结果可能相差比较大。

MBPS:

dd和iozone比较接近，读写分别是130+和70+

orion读57左右，写没有测（会删掉分区内所有的文件！）

bonnie++按块读是130左右，写是35左右；按字符是56左右，写是52左右

IOPS：

dd无结果

orion 440左右（只读）

iozone出错

bonnie++ 176.7（读写）