

专栏首页 > 悠风的采坑日记 > 笔记 | Linux 性能监视与跑分测试

笔记 | Linux 性能监视与跑分测试

2019-08-28 阅读 2.3K

测试环境

系统: Kali Linux amd64

用户: yowfung

工具: htop, net-tools, ping, iperf, UnixBench 等

Linux 性能测试是运维和安全测试相关人员的一门必备技能。了解基本的性能查看命令可以帮助你更全面地了解服务器环境，掌握基本的性能测试能够让你对服务器的运行性能有更细致的了解。这里我整合了一些基本的性能查看与测试技巧，主要包含有基本信息查看、磁盘读写测试、网络通信测试和综合性能跑分测试等，以便日后查阅。

基本信息查看

查看 CPU 参数:

```
cat /proc/cpuinfo
```

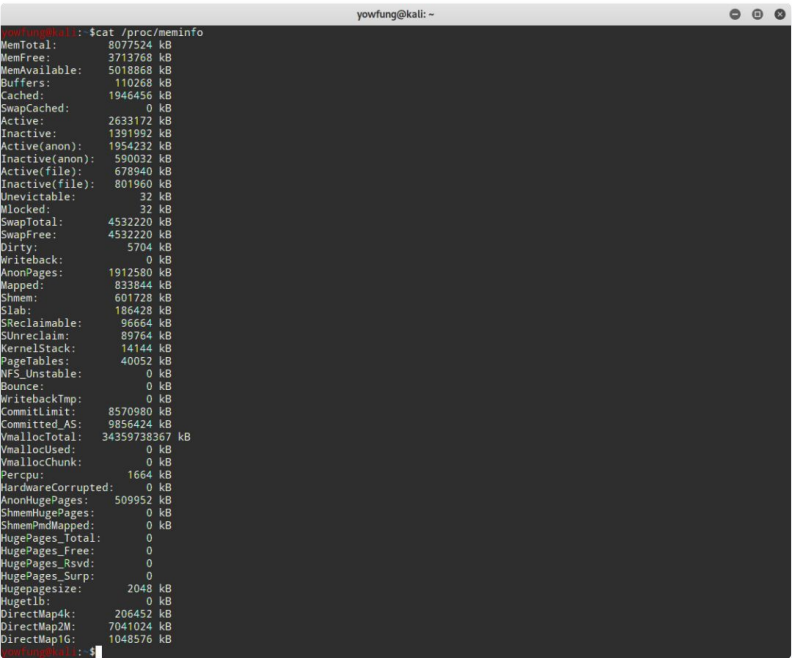
```
cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 742
model name    : Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz
stepping      : 19
microarch     : 0x66
cpu MHz       : 3800.000
cache size    : 8192 KB
physical id   : 0
siblings      : 4
core id       : 0
cpu cores     : 4
initial apicid : 0
fpu_exception : yes
bust level    : 2
mca           : 0
mca0_0        : 0
mca0_1        : 0
mca0_2        : 0
mca0_3        : 0
mca0_4        : 0
mca0_5        : 0
mca0_6        : 0
mca0_7        : 0
mca0_8        : 0
mca0_9        : 0
mca0_10       : 0
mca0_11       : 0
mca0_12       : 0
mca0_13       : 0
mca0_14       : 0
mca0_15       : 0
mca0_16       : 0
mca0_17       : 0
mca0_18       : 0
mca0_19       : 0
mca0_20       : 0
mca0_21       : 0
mca0_22       : 0
mca0_23       : 0
mca0_24       : 0
mca0_25       : 0
mca0_26       : 0
mca0_27       : 0
mca0_28       : 0
mca0_29       : 0
mca0_30       : 0
mca0_31       : 0
mca0_32       : 0
mca0_33       : 0
mca0_34       : 0
mca0_35       : 0
mca0_36       : 0
mca0_37       : 0
mca0_38       : 0
mca0_39       : 0
mca0_40       : 0
mca0_41       : 0
mca0_42       : 0
mca0_43       : 0
mca0_44       : 0
mca0_45       : 0
mca0_46       : 0
mca0_47       : 0
mca0_48       : 0
mca0_49       : 0
mca0_50       : 0
mca0_51       : 0
mca0_52       : 0
mca0_53       : 0
mca0_54       : 0
mca0_55       : 0
mca0_56       : 0
mca0_57       : 0
mca0_58       : 0
mca0_59       : 0
mca0_60       : 0
mca0_61       : 0
mca0_62       : 0
mca0_63       : 0
mca0_64       : 0
mca0_65       : 0
mca0_66       : 0
mca0_67       : 0
mca0_68       : 0
mca0_69       : 0
mca0_70       : 0
mca0_71       : 0
mca0_72       : 0
mca0_73       : 0
mca0_74       : 0
mca0_75       : 0
mca0_76       : 0
mca0_77       : 0
mca0_78       : 0
mca0_79       : 0
mca0_80       : 0
mca0_81       : 0
mca0_82       : 0
mca0_83       : 0
mca0_84       : 0
mca0_85       : 0
mca0_86       : 0
mca0_87       : 0
mca0_88       : 0
mca0_89       : 0
mca0_90       : 0
mca0_91       : 0
mca0_92       : 0
mca0_93       : 0
mca0_94       : 0
mca0_95       : 0
mca0_96       : 0
mca0_97       : 0
mca0_98       : 0
mca0_99       : 0
mca0_100      : 0
mca0_101      : 0
mca0_102      : 0
mca0_103      : 0
mca0_104      : 0
mca0_105      : 0
mca0_106      : 0
mca0_107      : 0
mca0_108      : 0
mca0_109      : 0
mca0_110      : 0
mca0_111      : 0
mca0_112      : 0
mca0_113      : 0
mca0_114      : 0
mca0_115      : 0
mca0_116      : 0
mca0_117      : 0
mca0_118      : 0
mca0_119      : 0
mca0_120      : 0
mca0_121      : 0
mca0_122      : 0
mca0_123      : 0
mca0_124      : 0
mca0_125      : 0
mca0_126      : 0
mca0_127      : 0
mca0_128      : 0
mca0_129      : 0
mca0_130      : 0
mca0_131      : 0
mca0_132      : 0
mca0_133      : 0
mca0_134      : 0
mca0_135      : 0
mca0_136      : 0
mca0_137      : 0
mca0_138      : 0
mca0_139      : 0
mca0_140      : 0
mca0_141      : 0
mca0_142      : 0
mca0_143      : 0
mca0_144      : 0
mca0_145      : 0
mca0_146      : 0
mca0_147      : 0
mca0_148      : 0
mca0_149      : 0
mca0_150      : 0
mca0_151      : 0
mca0_152      : 0
mca0_153      : 0
mca0_154      : 0
mca0_155      : 0
mca0_156      : 0
mca0_157      : 0
mca0_158      : 0
mca0_159      : 0
mca0_160      : 0
mca0_161      : 0
mca0_162      : 0
mca0_163      : 0
mca0_164      : 0
mca0_165      : 0
mca0_166      : 0
mca0_167      : 0
mca0_168      : 0
mca0_169      : 0
mca0_170      : 0
mca0_171      : 0
mca0_172      : 0
mca0_173      : 0
mca0_174      : 0
mca0_175      : 0
mca0_176      : 0
mca0_177      : 0
mca0_178      : 0
mca0_179      : 0
mca0_180      : 0
mca0_181      : 0
mca0_182      : 0
mca0_183      : 0
mca0_184      : 0
mca0_185      : 0
mca0_186      : 0
mca0_187      : 0
mca0_188      : 0
mca0_189      : 0
mca0_190      : 0
mca0_191      : 0
mca0_192      : 0
mca0_193      : 0
mca0_194      : 0
mca0_195      : 0
mca0_196      : 0
mca0_197      : 0
mca0_198      : 0
mca0_199      : 0
mca0_200      : 0
mca0_201      : 0
mca0_202      : 0
mca0_203      : 0
mca0_204      : 0
mca0_205      : 0
mca0_206      : 0
mca0_207      : 0
mca0_208      : 0
mca0_209      : 0
mca0_210      : 0
mca0_211      : 0
mca0_212      : 0
mca0_213      : 0
mca0_214      : 0
mca0_215      : 0
mca0_216      : 0
mca0_217      : 0
mca0_218      : 0
mca0_219      : 0
mca0_220      : 0
mca0_221      : 0
mca0_222      : 0
mca0_223      : 0
mca0_224      : 0
mca0_225      : 0
mca0_226      : 0
mca0_227      : 0
mca0_228      : 0
mca0_229      : 0
mca0_230      : 0
mca0_231      : 0
mca0_232      : 0
mca0_233      : 0
mca0_234      : 0
mca0_235      : 0
mca0_236      : 0
mca0_237      : 0
mca0_238      : 0
mca0_239      : 0
mca0_240      : 0
mca0_241      : 0
mca0_242      : 0
mca0_243      : 0
mca0_244      : 0
mca0_245      : 0
mca0_246      : 0
mca0_247      : 0
mca0_248      : 0
mca0_249      : 0
mca0_250      : 0
mca0_251      : 0
mca0_252      : 0
mca0_253      : 0
mca0_254      : 0
mca0_255      : 0
mca0_256      : 0
mca0_257      : 0
mca0_258      : 0
mca0_259      : 0
mca0_260      : 0
mca0_261      : 0
mca0_262      : 0
mca0_263      : 0
mca0_264      : 0
mca0_265      : 0
mca0_266      : 0
mca0_267      : 0
mca0_268      : 0
mca0_269      : 0
mca0_270      : 0
mca0_271      : 0
mca0_272      : 0
mca0_273      : 0
mca0_274      : 0
mca0_275      : 0
mca0_276      : 0
mca0_277      : 0
mca0_278      : 0
mca0_279      : 0
mca0_280      : 0
mca0_281      : 0
mca0_282      : 0
mca0_283      : 0
mca0_284      : 0
mca0_285      : 0
mca0_286      : 0
mca0_287      : 0
mca0_288      : 0
mca0_289      : 0
mca0_290      : 0
mca0_291      : 0
mca0_292      : 0
mca0_293      : 0
mca0_294      : 0
mca0_295      : 0
mca0_296      : 0
mca0_297      : 0
mca0_298      : 0
mca0_299      : 0
mca0_300      : 0
mca0_301      : 0
mca0_302      : 0
mca0_303      : 0
mca0_304      : 0
mca0_305      : 0
mca0_306      : 0
mca0_307      : 0
mca0_308      : 0
mca0_309      : 0
mca0_310      : 0
mca0_311      : 0
mca0_312      : 0
mca0_313      : 0
mca0_314      : 0
mca0_315      : 0
mca0_316      : 0
mca0_317      : 0
mca0_318      : 0
mca0_319      : 0
mca0_320      : 0
mca0_321      : 0
mca0_322      : 0
mca0_323      : 0
mca0_324      : 0
mca0_325      : 0
mca0_326      : 0
mca0_327      : 0
mca0_328      : 0
mca0_329      : 0
mca0_330      : 0
mca0_331      : 0
mca0_332      : 0
mca0_333      : 0
mca0_334      : 0
mca0_335      : 0
mca0_336      : 0
mca0_337      : 0
mca0_338      : 0
mca0_339      : 0
mca0_340      : 0
mca0_341      : 0
mca0_342      : 0
mca0_343      : 0
mca0_344      : 0
mca0_345      : 0
mca0_346      : 0
mca0_347      : 0
mca0_348      : 0
mca0_349      : 0
mca0_350      : 0
mca0_351      : 0
mca0_352      : 0
mca0_353      : 0
mca0_354      : 0
mca0_355      : 0
mca0_356      : 0
mca0_357      : 0
mca0_358      : 0
mca0_359      : 0
mca0_360      : 0
mca0_361      : 0
mca0_362      : 0
mca0_363      : 0
mca0_364      : 0
mca0_365      : 0
mca0_366      : 0
mca0_367      : 0
mca0_368      : 0
mca0_369      : 0
mca0_370      : 0
mca0_371      : 0
mca0_372      : 0
mca0_373      : 0
mca0_374      : 0
mca0_375      : 0
mca0_376      : 0
mca0_377      : 0
mca0_378      : 0
mca0_379      : 0
mca0_380      : 0
mca0_381      : 0
mca0_382      : 0
mca0_383      : 0
mca0_384      : 0
mca0_385      : 0
mca0_386      : 0
mca0_387      : 0
mca0_388      : 0
mca0_389      : 0
mca0_390      : 0
mca0_391      : 0
mca0_392      : 0
mca0_393      : 0
mca0_394      : 0
mca0_395      : 0
mca0_396      : 0
mca0_397      : 0
mca0_398      : 0
mca0_399      : 0
mca0_400      : 0
mca0_401      : 0
mca0_402      : 0
mca0_403      : 0
mca0_404      : 0
mca0_405      : 0
mca0_406      : 0
mca0_407      : 0
mca0_408      : 0
mca0_409      : 0
mca0_410      : 0
mca0_411      : 0
mca0_412      : 0
mca0_413      : 0
mca0_414      : 0
mca0_415      : 0
mca0_416      : 0
mca0_417      : 0
mca0_418      : 0
mca0_419      : 0
mca0_420      : 0
mca0_421      : 0
mca0_422      : 0
mca0_423      : 0
mca0_424      : 0
mca0_425      : 0
mca0_426      : 0
mca0_427      : 0
mca0_428      : 0
mca0_429      : 0
mca0_430      : 0
mca0_431      : 0
mca0_432      : 0
mca0_433      : 0
mca0_434      : 0
mca0_435      : 0
mca0_436      : 0
mca0_437      : 0
mca0_438      : 0
mca0_439      : 0
mca0_440      : 0
mca0_441      : 0
mca0_442      : 0
mca0_443      : 0
mca0_444      : 0
mca0_445      : 0
mca0_446      : 0
mca0_447      : 0
mca0_448      : 0
mca0_449      : 0
mca0_450      : 0
mca0_451      : 0
mca0_452      : 0
mca0_453      : 0
mca0_454      : 0
mca0_455      : 0
mca0_456      : 0
mca0_457      : 0
mca0_458      : 0
mca0_459      : 0
mca0_460      : 0
mca0_461      : 0
mca0_462      : 0
mca0_463      : 0
mca0_464      : 0
mca0_465      : 0
mca0_466      : 0
mca0_467      : 0
mca0_468      : 0
mca0_469      : 0
mca0_470      : 0
mca0_471      : 0
mca0_472      : 0
mca0_473      : 0
mca0_474      : 0
mca0_475      : 0
mca0_476      : 0
mca0_477      : 0
mca0_478      : 0
mca0_479      : 0
mca0_480      : 0
mca0_481      : 0
mca0_482      : 0
mca0_483      : 0
mca0_484      : 0
mca0_485      : 0
mca0_486      : 0
mca0_487      : 0
mca0_488      : 0
mca0_489      : 0
mca0_490      : 0
mca0_491      : 0
mca0_492      : 0
mca0_493      : 0
mca0_494      : 0
mca0_495      : 0
mca0_496      : 0
mca0_497      : 0
mca0_498      : 0
mca0_499      : 0
mca0_500      : 0
mca0_501      : 0
mca0_502      : 0
mca0_503      : 0
mca0_504      : 0
mca0_505      : 0
mca0_506      : 0
mca0_507      : 0
mca0_508      : 0
mca0_509      : 0
mca0_510      : 0
mca0_511      : 0
mca0_512      : 0
mca0_513      : 0
mca0_514      : 0
mca0_515      : 0
mca0_516      : 0
mca0_517      : 0
mca0_518      : 0
mca0_519      : 0
mca0_520      : 0
mca0_521      : 0
mca0_522      : 0
mca0_523      : 0
mca0_524      : 0
mca0_525      : 0
mca0_526      : 0
mca0_527      : 0
mca0_528      : 0
mca0_529      : 0
mca0_530      : 0
mca0_531      : 0
mca0_532      : 0
mca0_533      : 0
mca0_534      : 0
mca0_535      : 0
mca0_536      : 0
mca0_537      : 0
mca0_538      : 0
mca0_539      : 0
mca0_540      : 0
mca0_541      : 0
mca0_542      : 0
mca0_543      : 0
mca0_544      : 0
mca0_545      : 0
mca0_546      : 0
mca0_547      : 0
mca0_548      : 0
mca0_549      : 0
mca0_550      : 0
mca0_551      : 0
mca0_552      : 0
mca0_553      : 0
mca0_554      : 0
mca0_555      : 0
mca0_556      : 0
mca0_557      : 0
mca0_558      : 0
mca0_559      : 0
mca0_560      : 0
mca0_561      : 0
mca0_562      : 0
mca0_563      : 0
mca0_564      : 0
mca0_565      : 0
mca0_566      : 0
mca0_567      : 0
mca0_568      : 0
mca0_569      : 0
mca0_570      : 0
mca0_571      : 0
mca0_572      : 0
mca0_573      : 0
mca0_574      : 0
mca0_575      : 0
mca0_576      : 0
mca0_577      : 0
mca0_578      : 0
mca0_579      : 0
mca0_580      : 0
mca0_581      : 0
mca0_582      : 0
mca0_583      : 0
mca0_584      : 0
mca0_585      : 0
mca0_586      : 0
mca0_587      : 0
mca0_588      : 0
mca0_589      : 0
mca0_590      : 0
mca0_591      : 0
mca0_592      : 0
mca0_593      : 0
mca0_594      : 0
mca0_595      : 0
mca0_596      : 0
mca0_597      : 0
mca0_598      : 0
mca0_599      : 0
mca0_600      : 0
mca0_601      : 0
mca0_602      : 0
mca0_603      : 0
mca0_604      : 0
mca0_605      : 0
mca0_606      : 0
mca0_607      : 0
mca0_608      : 0
mca0_609      : 0
mca0_610      : 0
mca0_611      : 0
mca0_612      : 0
mca0_613      : 0
mca0_614      : 0
mca0_615      : 0
mca0_616      : 0
mca0_617      : 0
mca0_618      : 0
mca0_619      : 0
mca0_620      : 0
mca0_621      : 0
mca0_622      : 0
mca0_623      : 0
mca0_624      : 0
mca0_625      : 0
mca0_626      : 0
mca0_627      : 0
mca0_628      : 0
mca0_629      : 0
mca0_630      : 0
mca0_631      : 0
mca0_632      : 0
mca0_633      : 0
mca0_634      : 0
mca0_635      : 0
mca0_636      : 0
mca0_637      : 0
mca0_638      : 0
mca0_639      : 0
mca0_640      : 0
mca0_641      : 0
mca0_642      : 0
mca0_643      : 0
mca0_644      : 0
mca0_645      : 0
mca0_646      : 0
mca0_647      : 0
mca0_648      : 0
mca0_649      : 0
mca0_650      : 0
mca0_651      : 0
mca0_652      : 0
mca0_653      : 0
mca0_654      : 0
mca0_655      : 0
mca0_656      : 0
mca0_657      : 0
mca0_658      : 0
mca0_659      : 0
mca0_660      : 0
mca0_661      : 0
mca0_662      : 0
mca0_663      : 0
mca0_664      : 0
mca0_665      : 0
mca0_666      : 0
mca0_667      : 0
mca0_668      : 0
mca0_669      : 0
mca0_670      : 0
mca0_671      : 0
mca0_672      : 0
mca0_673      : 0
mca0_674      : 0
mca0_675      : 0
mca0_676      : 0
mca0_677      : 0
mca0_678      : 0
mca0_679      : 0
mca0_680      : 0
mca0_681      : 0
mca0_682      : 0
mca0_683      : 0
mca0_684      : 0
mca0_685      : 0
mca0_686      : 0
mca0_687      : 0
mca0_688      : 0
mca0_689      : 0
mca0_690      : 0
mca0_691      : 0
mca0_692      : 0
mca0_693      : 0
mca0_694      : 0
mca0_695      : 0
mca0_696      : 0
mca0_697      : 0
mca0_698      : 0
mca0_699      : 0
mca0_700      : 0
mca0_701      : 0
mca0_702      : 0
mca0_703      : 0
mca0_704      : 0
mca0_705      : 0
mca0_706      : 0
mca0_707      : 0
mca0_708      : 0
mca0_709      : 0
mca0_710      : 0
mca0_711      : 0
mca0_712      : 0
mca0_713      : 0
mca0_714      : 0
mca0_715      : 0
mca0_716      : 0
mca0_717      : 0
mca0_718      : 0
mca0_719      : 0
mca0_720      : 0
mca0_721      : 0
mca0_722      : 0
mca0_723      : 0
mca0_724      : 0
mca0_725      : 0
mca0_726      : 0
mca0_727      : 0
mca0_728      : 0
mca0_729      : 0
mca0_730      : 0
mca0_731      : 0
mca0_732      : 0
mca0_733      : 0
mca0_734      : 0
mca0_735      : 0
mca0_736      : 0
mca0_737      : 0
mca0_738      : 0
mca0_739      : 0
mca0_740      : 0
mca0_741      : 0
mca0_742      : 0
mca0_743      : 0
mca0_744      : 0
mca0_745      : 0
mca0_746      : 0
mca0_747      : 0
mca0_748      : 0
mca0_749      : 0
mca0_750      : 0
mca0_751      : 0
mca0_752      : 0
mca0_753      : 0
mca0_754      : 0
mca0_755      : 0
mca0_756      : 0
mca0_757      : 0
mca0_758      : 0
mca0_759      : 0
mca0_760      : 0
mca0_761      : 0
mca0_762      : 0
mca0_763      : 0
mca0_764      : 0
mca0_765      : 0
mca0_766      : 0
mca0_767      : 0
mca0_768      : 0
mca0_769      : 0
mca0_770      : 0
mca0_771      : 0
mca0_772      : 0
mca0_773      : 0
mca0_774      : 0
mca0_775      : 0
mca0_776      : 0
mca0_777      : 0
mca0_778      : 0
mca0_779      : 0
mca0_780      : 0
mca0_781      : 0
mca0_782      : 0
mca0_783      : 0
mca0_784      : 0
mca0_785      : 0
mca0_786      : 0
mca0_787      : 0
mca0_788      : 0
mca0_789      : 0
mca0_790      : 0
mca0_791      : 0
mca0_792      : 0
mca0_793      : 0
mca0_794      : 0
mca0_795      : 0
mca0_796      : 0
mca0_797      : 0
mca0_798      : 0
mca0_799      : 0
mca0_800      : 0
mca0_801      : 0
mca0_802      : 0
mca0_803      : 0
mca0_804      : 0
mca0_805      : 0
mca0_806      : 0
mca0_807      : 0
mca0_808      : 0
mca0_809      : 0
mca0_810      : 0
mca0_811      : 0
mca0_812      : 0
mca0_813      : 0
mca0_814      : 0
mca0_815      : 0
mca0_816      : 0
mca0_817      : 0
mca0_818      : 0
mca0_819      : 0
mca0_820      : 0
mca0_821      : 0
mca0_822      : 0
mca0_823      : 0
mca0_824      : 0
mca0_825      : 0
mca0_826      : 0
mca0_827      : 0
mca0_828      : 0
mca0_829      : 0
mca0_830      : 0
mca0_831      : 0
mca0_832      : 0
mca0_833      : 0
mca0_834      : 0
mca0_835      : 0
mca0_836      : 0
mca0_837      : 0
mca0_838      : 0
mca0_839      : 0
mca0_840      : 0
mca0_841      : 0
mca0_842      : 0
mca0_843      : 0
mca0_844      : 0
mca0_845      : 0
mca0_846      : 0
mca0_847      : 0
mca0_848      : 0
mca0_849      : 0
mca0_850      : 0
mca0_851      : 0
mca0_852      : 0
mca0_853      : 0
mca0_854      : 0
mca0_855      : 0
mca0_856      : 0
mca0_857      : 0
mca0_858      : 0
mca0_859      : 0
mca0_860      : 0
mca0_861      : 0
mca0_862      : 0
mca0_863      : 0
mca0_864      : 0
mca0_865      : 0
mca0_866      : 0
mca0_867      : 0
mca0_868      : 0
mca0_869      : 0
mca0_870      : 0
mca0_871      : 0
mca0_872      : 0
mca0_873      : 0
mca0_874      : 0
mca0_875      : 0
mca0_876      : 0
mca0_877      : 0
mca0_878      : 0
mca0_879      : 0
mca0_880      : 0
mca0_881      : 0
mca0_882      : 0
mca0_883      : 0
mca0_884      : 0
mca0_885      : 0
mca0_886      : 0
mca0_887      : 0
mca0_888      : 0
mca0_889      : 0
mca0_890      : 0
mca0_891      : 0
mca0_892      : 0
mca0_893      : 0
mca0_894      : 0
mca0_895      : 0
mca0_896      : 0
mca0_897      : 0
mca0_898      : 0
mca0_899      : 0
mca0_900      : 0
mca0_901      : 0
mca0_902      : 0
mca0_903      : 0
mca0_904      : 0
mca0_905      : 0
mca0_906      : 0
mca0_907      : 0
mca0_908      : 0
mca0_909      : 0
mca0_910      : 0
mca0_911      : 0
mca0_912      : 0
mca0_913      : 0
mca0_914      : 0
mca0_915      : 0
mca0_916      : 0
mca0_917      : 0
mca0_918      : 0
mca0_919      : 0
mca0_920      : 0
mca0_921      : 0
mca0_922      : 0
mca0_923      : 0
mca0_924      : 0
mca0_925      : 0
mca0_926      : 0
mca0_927      : 0
mca0_928      : 0
mca0_929      : 0
mca0_930      : 0
mca0_931      : 0
mca0_932      : 0
mca0_933      : 0
mca0_934      : 0
mca0_935      : 0
mca0_936      : 0
mca0_937      : 0
mca0_938      : 0
mca0_939      : 0
mca0_940      : 0
mca0_941      : 0
mca0_942      : 0
mca0_943      : 0
mca0_944      : 0
mca0_945      : 0
mca0_946      : 0
mca0_947      : 0
mca0_948      : 0
mca0_949      : 0
mca0_950      : 0
mca0_951      : 0
mca0_952      : 0
mca0_953      : 0
mca0_954      : 0
mca0_955      : 0
mca0_956      : 0
mca0_957      : 0
mca0_958      : 0
mca0_959      : 0
mca0_960      : 0
mca0_961      : 0
mca0_962      : 0
mca0_963      : 0
mca0_964      : 0

```

```
# 查看 CPU 总线线程数:
cat /proc/cpuinfo | grep 'processor'
```

查看内存参数:

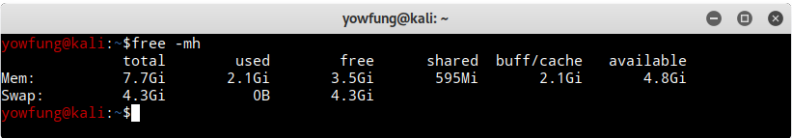
```
cat /proc/meminfo
```



该命令可以查看内存的详细信息，包括内存容量、交换空间、高速缓存等。

查看内存使用情况:

```
free -mh
```



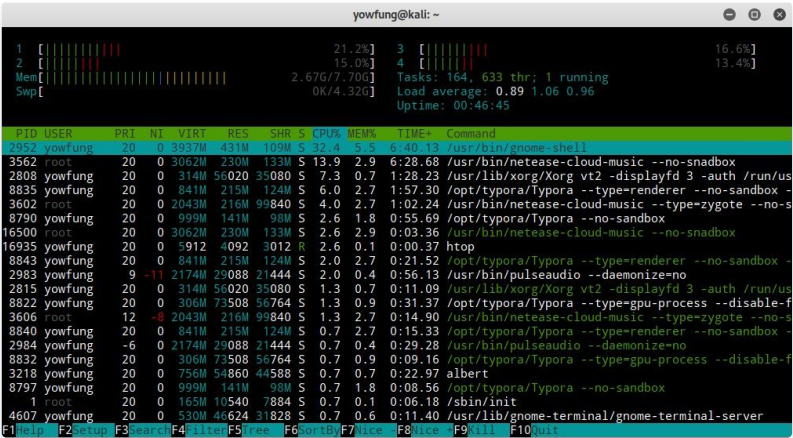
该命令可查看当前 Linux 对内存和交换空间的占用情况。

查看进程情况:

```
# 安装 htop
sudo apt install htop

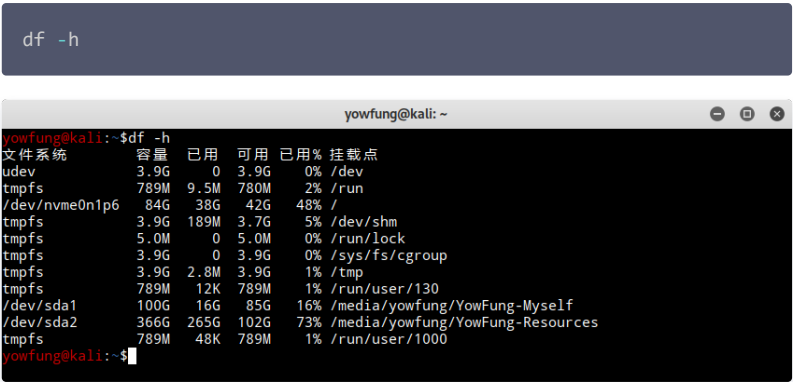
# 运行 htop
htop
```





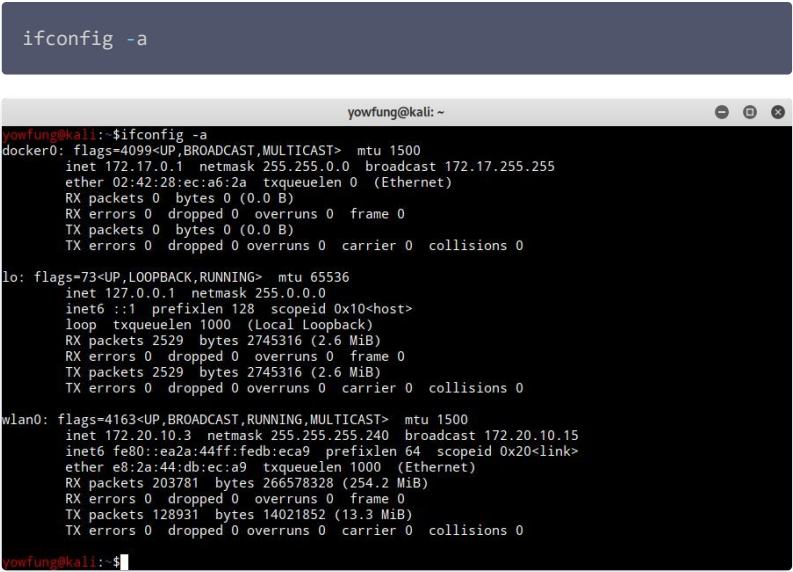
htop 软件可以查看当前的 CPU 和内存使用情况，以及当前正在运行的进程，其界面也是很炫酷美观的。

查看磁盘使用情况：



该命令可以查看所有挂载分区的容量和使用情况。

查看网卡信息：



查看 PCI 总线信息：



```
yowfung@kali:~$ lspci
00:00.0 Host bridge: Intel Corporation Xeon E3-1200 v6/7th Gen Core Processor Host Bridge/DRAM Registers (rev 08)
00:02.0 VGA compatible controller: Intel Corporation UHD Graphics 620 (rev 07)
00:04.0 Signal processing controller: Intel Corporation Xeon E3-1200 v5/E3-1500 v5/6th Gen Core Processor Thermal Subsystem (rev 0B)
00:14.0 USB controller: Intel Corporation Sunrise Point-LP USB 3.0 xHCI Controller (rev 21)
00:14.2 Signal processing controller: Intel Corporation Sunrise Point-LP Thermal subsystem (rev 21)
00:15.0 Signal processing controller: Intel Corporation Sunrise Point-LP Serial I/O I2C Controller #0 (rev 21)
00:16.0 Communication controller: Intel Corporation Sunrise Point-LP CSME HECI #1 (rev 21)
00:17.0 SATA controller: Intel Corporation Sunrise Point-LP SATA Controller [AHCI mode] (rev 21)
00:1c.0 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #1 (rev f1)
00:1c.4 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #5 (rev f1)
00:1c.5 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #6 (rev f1)
00:1d.0 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #9 (rev f1)
00:1f.0 ISA bridge: Intel Corporation Sunrise Point LPC Controller/eSPI Controller (rev 21)
00:1f.2 Memory controller: Intel Corporation Sunrise Point-LP PMC (rev 21)
00:1f.3 Audio device: Intel Corporation Sunrise Point-LP HD Audio (rev 21)
00:1f.4 SMBus: Intel Corporation Sunrise Point-LP SMBus (rev 21)
01:00.0 3D controller: NVIDIA Corporation GP108M [GeForce MX150] (rev a1)
02:00.0 SD Host controller: O2 Micro, Inc. SD/MMC Card Reader Controller (rev 01)
23:00.0 Network controller: Qualcomm Atheros QCA9377 802.11ac Wireless Network Adapter (rev 31)
94:00.0 Non-Volatile memory controller: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PW961
```

磁盘读写测试

磁盘 I/O (Input/Output) 读写速度是磁盘性能的一个重要指标。测试主机磁盘 IO 性能可以用以下两个命令。

```
# 方案一
dd if=/dev/zero of=test bs=4k count=256000 oflag=dsync

# 方案二
dd if=/dev/zero of=test bs=4k count=256000 conv=fdatasync
```

```
yowfung@kali:~$ dd if=/dev/zero of=test bs=4k count=256000 oflag=dsync
记录了 256000+0 的读入
记录了 256000+0 的写出
1048576000 bytes (1.0 GB, 1000 MiB) copied, 1733.06 s, 605 kB/s
yowfung@kali:~$ dd if=/dev/zero of=test bs=4k count=256000 conv=fdatasync
记录了 256000+0 的读入
记录了 256000+0 的写出
1048576000 bytes (1.0 GB, 1000 MiB) copied, 1.17986 s, 889 MB/s
yowfung@kali:~$
```

两个都是往硬盘中写入 1 Gbytes 的数据，只是第一个的速度慢的要命。

使用 `dsync`，`dd` 会从 `/dev/zero` 中，每次读取 4Kbytes 数据，然后直接写入到硬盘当中，重复此步骤，直到共读取并且写入了 1 Gbytes 的数据。

使用 `fdatasync`，`dd` 会从 `/dev/zero` 中一次性读取 1 Gbytes 的数据，写入到磁盘的缓存中，然后再从磁盘缓存中读取，一次性写入到硬盘当中。

网络通信测试

测试网络连通情况：

```
ping -c 4 {HOST}
```

```
yowfung@kali:~$ sudo ping -c 4 www.baidu.com
PING www.a.shifen.com (183.232.231.172) 56(84) bytes of data:
64 bytes from 183.232.231.172 (183.232.231.172): icmp_seq=1 ttl=55 time=53.4 ms
64 bytes from 183.232.231.172 (183.232.231.172): icmp_seq=2 ttl=55 time=43.7 ms
64 bytes from 183.232.231.172 (183.232.231.172): icmp_seq=3 ttl=55 time=83.1 ms
64 bytes from 183.232.231.172 (183.232.231.172): icmp_seq=4 ttl=55 time=44.7 ms

--- www.a.shifen.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 689ms
rtt min/avg/max/mdev = 43.733/56.252/83.142/15.978 ms
yowfung@kali:~$
```

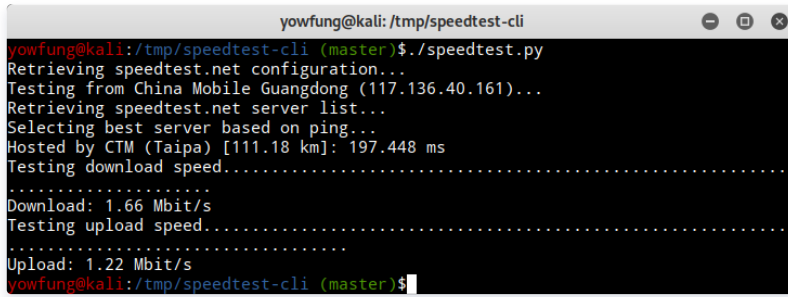
其中 `{HOST}` 为目标主机，可以是域名或 IP 地址。有显示响应时间则表示网络连通。

测试网速：

```
# 下载测试脚本
git clone https://github.com/sivel/speedtest-cli.git

# 进入下载目录
```

```
cd speedtest-cli/  
  
# 运行网速测试  
./speedtest.py
```



A terminal window titled 'yowfung@kali: /tmp/speedtest-cli' showing the execution of 'speedtest.py'. The output includes: 'Retrieving speedtest.net configuration...', 'Testing from China Mobile Guangdong (117.136.40.161)...', 'Retrieving speedtest.net server list...', 'Selecting best server based on ping...', 'Hosted by CTM (Taipa) [111.18 km]: 197.448 ms', 'Testing download speed...', 'Download: 1.66 Mbit/s', 'Testing upload speed...', 'Upload: 1.22 Mbit/s'. The prompt returns to 'yowfung@kali: /tmp/speedtest-cli (master)\$'.

运行这个脚本后，可以明显看得出下载和上传的网速。需要注意的是，这里的单位是 **Mbit/s**，而不是我们平常所说网速的多少 **MB/s**，他们之间的换算关系为 $8\text{Mbit/s} = 1\text{MB/s}$ 。

测试 TCP 吞吐量和 UDP 丢包率：

iperf 是一款基于 TCP/IP 和 UDP/IP 的网络性能测试工具，它可以用来测量网络带宽和网络质量，还可以提供网络延迟抖动、数据包丢失率、最大传输单元等统计信息。网络管理员可以根据这些信息了解并判断网络性能问题，从而定位网络瓶颈，解决网络故障。

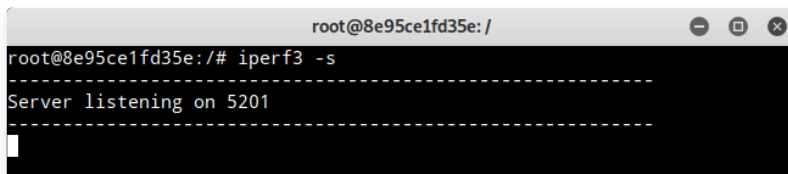
以下的测试过程需要由两台主机来配合完成，如果没有两台物理主机的话，也可以在虚拟机或 docker 容器中进行。

在下面的测试中，我们假设客户端主机为 **172.20.10.3**，服务器主机为 **172.17.0.2**，测试前分别在客户端主机和服务器主机中安装 iperf。

```
# 安装 iperf 软件  
sudo apt install iperf3
```

在服务器主机中输入以下命令开启 iperf 服务，此时服务器会启动监听 5201 端口。

```
iperf3 -s
```



A terminal window titled 'root@8e95ce1fd35e: /' showing the execution of 'iperf3 -s'. The output is 'Server listening on 5201'.

测试 TCP 吞吐量：在客户端主机中输入以下命令。

```
iperf3 -c {ServerHost}
```



```
yowfung@kali: ~  
yowfung@kali:~$iperf3 -c 172.17.0.2  
Connecting to host 172.17.0.2, port 5201  
[ 5] local 172.17.0.1 port 55780 connected to 172.17.0.2 port 5201  
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd  
[ 5] 0.00-1.00 sec  7.81 GBytes 67.0 Gbits/sec  0    423 KBytes  
[ 5] 1.00-2.00 sec  7.74 GBytes 66.5 Gbits/sec  0    489 KBytes  
[ 5] 2.00-3.00 sec  8.14 GBytes 69.9 Gbits/sec  0    489 KBytes  
[ 5] 3.00-4.00 sec  7.73 GBytes 66.4 Gbits/sec  0    489 KBytes  
[ 5] 4.00-5.00 sec  7.27 GBytes 62.4 Gbits/sec  0    515 KBytes  
[ 5] 5.00-6.00 sec  6.41 GBytes 55.1 Gbits/sec  0    578 KBytes  
[ 5] 6.00-7.00 sec  8.12 GBytes 69.8 Gbits/sec  0    642 KBytes  
[ 5] 7.00-8.00 sec  7.92 GBytes 68.0 Gbits/sec  0    642 KBytes  
[ 5] 8.00-9.00 sec  7.96 GBytes 68.4 Gbits/sec  0    642 KBytes  
[ 5] 9.00-10.00 sec 7.12 GBytes 61.2 Gbits/sec  0    642 KBytes  
-----  
[ ID] Interval      Transfer    Bitrate      Retr  
[ 5] 0.00-10.00 sec 76.2 GBytes 65.5 Gbits/sec  0  
[ 5] 0.00-10.00 sec 76.2 GBytes 65.5 Gbits/sec  
iperf Done.  
yowfung@kali:~$
```

其中 {ServerHost} 为服务器主机的 IP，这里为上面假设的 172.17.0.2。

为了确定网卡的最大吞吐量，iperf 将尝试从客户端尽可能快地向服务端发送数据请求，并且会输出发送的数据量和网卡平均带宽值。

从图中输出信息可以看出，iperf 默认测试 10 秒钟，共传送了 76.2 GB 的数据量，网卡的带宽平均速率为 65.5 Gbits/s，即 8.19 GB/s。由于我在测试时，服务器主机在本地电脑的 docker 容器中，故才会有这么惊人的传输速率，而在实际应用时，服务器主机应该选择实际被测试的目标，一般情况下不会有这么高的传输速率。

为了模拟更真实的测试，你可以添加 iperf 命令的参数，自定义传送的数据量、测试时长、输出频率以及线程数等。

```
iperf3 -c {ServerHost} -t {Time} -i {Interval} -n {Number} -P {Parallel}
```

```
yowfung@kali: ~  
yowfung@kali:~$iperf3 -c 172.17.0.2 -i 5 -n 107374182400 -P 2  
Connecting to host 172.17.0.2, port 5201  
[ 5] local 172.17.0.1 port 55866 connected to 172.17.0.2 port 5201  
[ 7] local 172.17.0.1 port 55868 connected to 172.17.0.2 port 5201  
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd  
[ 5] 0.00-5.00 sec  22.1 GBytes 37.9 Gbits/sec  0    351 KBytes  
[ 7] 0.00-5.00 sec  22.1 GBytes 37.9 Gbits/sec  97   313 KBytes  
[SUM] 0.00-5.00 sec  44.2 GBytes 75.9 Gbits/sec  97  
-----  
[ 5] 5.00-10.00 sec  21.4 GBytes 36.7 Gbits/sec  0    1.08 MBytes  
[ 7] 5.00-10.00 sec  21.4 GBytes 36.7 Gbits/sec  0    389 KBytes  
[SUM] 5.00-10.00 sec  42.8 GBytes 73.5 Gbits/sec  0  
-----  
[ 5] 10.00-12.51 sec  6.54 GBytes 22.4 Gbits/sec  0    1.08 MBytes  
[ 7] 10.00-12.51 sec  6.54 GBytes 22.4 Gbits/sec  0    406 KBytes  
[SUM] 10.00-12.51 sec  13.1 GBytes 44.7 Gbits/sec  0  
-----  
[ ID] Interval      Transfer    Bitrate      Retr  
[ 5] 0.00-12.51 sec  50.0 GBytes 34.3 Gbits/sec  0  
[ 5] 0.00-12.51 sec  50.0 GBytes 34.3 Gbits/sec  
[ 7] 0.00-12.51 sec  50.0 GBytes 34.3 Gbits/sec  97  
[ 7] 0.00-12.51 sec  50.0 GBytes 34.3 Gbits/sec  97  
[SUM] 0.00-12.51 sec  100 GBytes 68.7 Gbits/sec  97  
[SUM] 0.00-12.51 sec  100 GBytes 68.6 Gbits/sec  
iperf Done.  
yowfung@kali:~$
```

其中 {ServerHost} 为服务器主机的 IP，{Time} 为测试的时长，{Interval} 为每隔几秒输出一个测试结果，{Number} 为要传送的数据量，{Parallel} 为采用多少线程进行传输。这里值得注意的是，这里 -t 参数和 -n 参数不能同时设置。

如上图所示，这里设置了 107374182400 字节的数据量，即 100 GB，每隔 5 秒输出一次测试结果，并且设置为双线程传输。可以看到最终的平均传输速率为 68.6 Gbits/s，并且每一条虚线分隔的，都是一次传输的记录，每次都有两条（双线程）记录和一条总的平均记录。

这里要提醒一下，如果你是用来测试你的付费云主机，请注意你的带宽流量费用，非必要不要像我这样一次性传送这么大的数据量。

下面来测试 UDP 丢包和延迟。

说明：由于 UDP 协议是一个非面向连接的轻量级传输协议，并且不提供可靠的数据传输服务，因此对 UDP 应用的关注点不是传输数据有多快，而是它的丢包率和延时指标。

```
iperf3 -c {ServerHost} -u -i {Interval} -b {Bandwidth}
```

```
yowfung@kali: ~  
yowfung@kali:~$ iperf3 -c 172.17.0.2 -u -i 1 -b 200M  
Connecting to host 172.17.0.2, port 5201  
[ 5] local 172.17.0.1 port 56501 connected to 172.17.0.2 port 5201  
[ ID] Interval      Transfer    Bitrate    Total Datagrams  
[ 5] 0.00-1.00 sec  23.8 MBytes 200 Mbits/sec 17251  
[ 5] 1.00-2.00 sec  23.8 MBytes 200 Mbits/sec 17268  
[ 5] 2.00-3.00 sec  23.8 MBytes 200 Mbits/sec 17266  
[ 5] 3.00-4.00 sec  23.8 MBytes 200 Mbits/sec 17264  
[ 5] 4.00-5.00 sec  23.8 MBytes 200 Mbits/sec 17266  
[ 5] 5.00-6.00 sec  23.8 MBytes 200 Mbits/sec 17268  
[ 5] 6.00-7.00 sec  23.8 MBytes 200 Mbits/sec 17262  
[ 5] 7.00-8.00 sec  23.8 MBytes 200 Mbits/sec 17266  
[ 5] 8.00-9.00 sec  23.8 MBytes 200 Mbits/sec 17265  
[ 5] 9.00-10.00 sec 23.8 MBytes 200 Mbits/sec 17264  
-----  
[ ID] Interval      Transfer    Bitrate    Jitter    Lost/Total Datagrams  
[ 5] 0.00-10.00 sec 238 MBytes 200 Mbits/sec 0.000 ms  0/172640 (0%) sender  
[ 5] 0.00-10.00 sec 238 MBytes 200 Mbits/sec 0.006 ms  0/172640 (0%) receiver  
iperf Done.  
yowfung@kali:~$
```

其中 {ServerHost} 为服务器主机的 IP，{Interval} 为每隔多少秒输出一次测试结果，{Bandwidth} 为需要传输的带宽。进行 UDP 测试的主要参数为 `-u`。如上图，设置为每隔 1 秒输出测试结果，带宽为 200M。注意虚线下面的信息，`Jitter` 表示抖动时间（或称为传输延迟），`Lost/Total` 表示丢失的数据报和总的数据报数量，后面的 0% 是平均丢包的比率，`Datagrams` 表示总共传输数据报的数量。

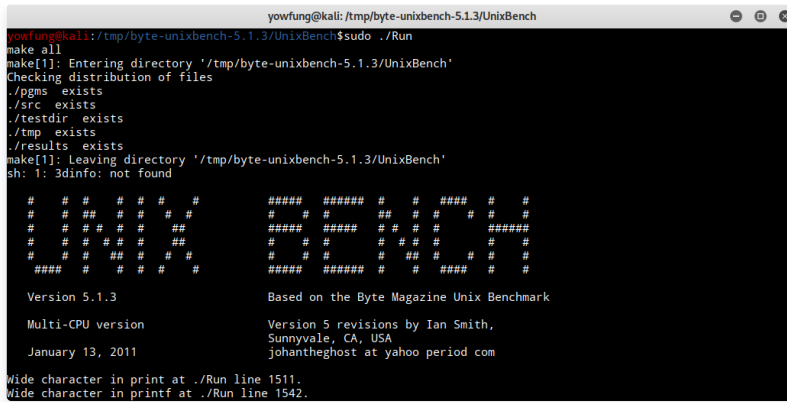
这里需要说明的是，由于我是使用本地的 docker 容器进行测试，所以得到的结果比较理想（0%），而实际上 UDP 的丢包率很少会出现这样理想的情况的。

综合性能跑分

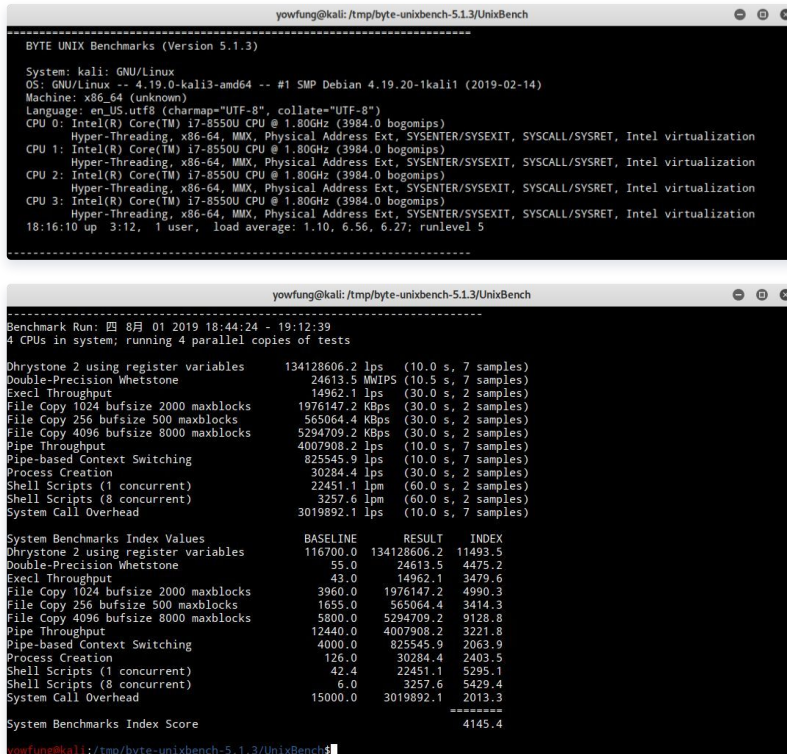
UnixBench 是一款开源的测试 Unix 系统基本性能的工具，是比较通用的测试性能的工具，Unixbench 的主要测试项目有：系统调用、读写、进程、图形化测试、2D、3D、管道、运算、C库等系统基准性能提供测试数据。

操作步骤：

```
# 进入系统临时目录  
cd /tmp  
  
# 下载 UnixBench 源码包  
wget https://github.com/kdlucas/byte-unixbench/archive/v5.1.3.tar.gz  
  
# 解压缩  
tar -zxvf v5.1.3.tar.gz  
  
# 进入软件目录  
cd byte-unixbench-5.1.3/UnixBench/  
  
# 运行跑分测试  
sudo ./Run
```



测试时间比较久，等待测试完成后，可以看到如下这样的界面：



看最后一行的 **System Benchmarks Index Score**，是跑分测试的综合总分。

列表中 **BASELINE** 为基准线, **RESULT** 为跑分结果, 而 **INDEX** 则为测试系统的测试结果与一个基线系统测试结果的比值, 即 $INDEX = RESULT / BASELINE \times 10$ 。

这样得出的值比原始测试结果更有参考价值。

再来看看每一项测试结果所表示的含义吧。

项目	说明
Dhrystone 2 using register variables	此项用于测试 string handling。因为没有浮点操作，所以深受软件和硬件设计、编译和链接、代码优化、对内存的cache、等待状态、整数数据类型的影响。
Double-Precision Whetstone	这一项测试浮点数操作的速度和效率。覆盖面很广的一系列 C 函数：sin, cos, sqrt, exp, log 被用于整数和浮点数的数学运算、数组访问、条件分支和程序调用。



项目	说明
Execl Throughput	此测试考察每秒钟可以执行的 execl 系统调用的次数。
File Copy	测试从一个文件向另外一个文件传输数据的速率。每次测试使用不同大小的缓冲区。
Pipe Throughput	测试一秒钟内一个进程可以向一个管道写 512 字节数据然后再读回的次数。
Pipe-bases Context Switching	测试两个进程每秒钟通过一个管道交换一个不断增长的整数的次数。这个测试程序首先创建一个子进程，再和这个子进程进行双向的管道传输。
Process Creation	测试每秒钟一个进程可以创建子进程然后收回子进程的次数（子进程一定立即退出）。Process Creation 的关注点是新进程进程控制块（process control block）的创建和内存分配。
Shell Scripts	测试一秒钟内一个进程可以并发地开始一个 shell 脚本的 n（一般取值为1、2、4、8）个拷贝的次数。
System Call Overhead	测试进入和离开操作系统内核的代价，即一次系统调用的代价。它利用一个反复地调用 getpid 函数的小程序达到此目的。
Graphical Tests	测试非常粗的 2D 和 3D 图形性能，尤其是 3D 测试非常有限。测试结果和硬件，系统合适的驱动关系很大。

必要说明：

我在写这篇推文的时候，也参考了网上很多大咖写的博客和书籍，这里所涉及的大部分方案均不是我首先提出的，我只是将各位大咖的资料做了一个整合与加工，如果对本篇推文提及的一些技术有不懂的，可以移步查阅下面的参考来源，也可以在公众号给我留言。

参考文章：

- 《dd命令的conv=fsync,oflag=sync/dsync》 | 学步园
- 《基于Linux系统的性能测试》 | CSDN
- 《Unixbench 测试工具分析》 | 简书
- 《Linux网络性能评估工具iperf、CHARIOT测试网络吞吐量》 | konglingbin

本文分享自微信公众号 - 悠风的采坑日记 (yowfung)，作者：小悠风
原文出处及转载信息见文内详细说明，如有侵权，请联系 yunjia_community@tencent.com 删除。
原始发表时间：2019-08-02
本文参与[腾讯云自媒体分享计划](#)，欢迎正在阅读的你也加入，一起分享。

测试服务 WeTest

UDP

TCP/IP

Unix

举报

点赞 3

分享

0 条评论

我来说两句

登录 后参与评论

相关文章

笔记 | Linux 帮助命令

悠风

【粤嵌实训】Python小游戏开发之“代码大战”

自从 PHP 大张旗鼓宣称其为世界上最好的编程语言后，世界各路编程语言群起讨伐，战火一直蔓延到21世纪中叶。战...



悠风

「学习笔记」C语言（二）：指针篇

这是我大一暑假时系统学习谭浩强第三版《C程序设计》时写下的笔记。也正是因为那一次耗时整整一个星期系统性...



悠风

移动测试避坑指南（第一篇）：从流程到技术的知识概要

京东技术

在快速迭代的项目中减少测试返工

在互联网产品中，产品的迭代速度越来越快，项目中的测试同学面临着前期需求摇摆不定，中间各种开发进度死锁，而发布时间却无法推迟。项目的前期阶段似乎总是...

飞天小子

浅酌 iOS 11 兼容性

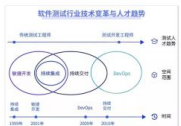
苹果在 WWDC 2017大会，公布了最新的 iOS 11，系统新特性肯定是让不少果粉充满期待。在网上已能看到不少关于...

IT特性	iOS 11版本兼容性运营运行	
备注	功能开发：设置、相机、格式	
UIP	新增系统级原生UI功能支持APP	
UI	系统UI更新	测试验证
UI/UX	APP程序功能与系统UI兼容性验证，UI/UX兼容性验证，UI/UX兼容性验证，UI/UX兼容性验证	UI/UX兼容性验证，UI/UX兼容性验证，UI/UX兼容性验证，UI/UX兼容性验证

WeTest质量开放平台团队

Python 测试开发从入门到高手成长之路

在“质量第一，效率为王”的移动互联网和大数据时代，互联网IT 技术团队为了应对产品快速迭代要求，就必须具备持续...

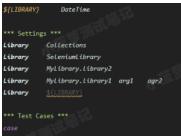


测试小兵

Robot Framework (5) - 使用测试库

https://www.cnblogs.com/poloyy/category/1770899.html

小菠萝测试笔记



维护Selenium测试自动化的最佳实践

自动化测试框架和基础组件需要及时、良好的维护。如果团队无法跟上与维护相关的需求，那么以后可能会付出更大代价，最终带来自动化项目的深陷泥潭。这里有一些减...

FunTester

功能测试与非功能测试

它测试了被测软件的行为。根据客户的需求，称为软件规范或需求规范的文档将用作测试应用程序的指南。

用户7466307

[更多文章 >](#)

社区

专栏文章
阅读清单
互动问答
技术沙龙
技术快讯
团队主页
开发者手册
智能钛AI

活动

原创分享计划
自媒体分享计划
邀请作者入驻
自荐上首页
在线直播
生态合作计划

资源

腾讯云大学
技术周刊
社区标签
开发者实验室

关于

视频介绍
社区规范
免责声明
联系我们

云+社区



扫码关注云+社区
领取腾讯云代金券

热门产品

域名注册	云服务器	区块链服务	消息队列	网络加速	云数据库	域名解析
云存储	视频直播					

热门推荐

人脸识别	腾讯会议	企业云	CDN 加速	视频通话	图像分析	MySQL 数据库
SSL 证书	语音识别					

更多推荐

数据安全	负载均衡	短信	文字识别	云点播	商标注册	小程序开发
网站监控	数据迁移					