# 官方不支持webbench支持post方式 别人改写的webbench

webbench支持post: [github地址](https://github.com/yi5971/webbench-1.5) https://github.com/yi5971/webbench-1.5

webbech是个不错的压力测试工具，而且非常小，所以就拿过来试用下。

遗憾从[官网](http://home.tiscali.cz/~cz210552/webbench.html)找到的源码，编译安装后不支持post方式。自己动手修改下了源码，以支持post方式 放置于外头的压缩包

官方webbench

1. **官网**  
   <http://home.tiscali.cz/~cz210552/webbench.html>
2. **下载**  
   wget <http://home.tiscali.cz/~cz210552/distfiles/webbench-1.5.tar.gz>

**或者：**<http://www.ha97.com/code/webbench-1.5.tar.gz>

1. **安装【1】未尝试**

tar -xzvf webbench-1.5.tar.gz

cd webbench-1.5

mkdir /usr/local/man

make install clean

**安装【2】自己测试可用**

wget <http://blog.s135.com/soft/linux/webbench/webbench-1.5.tar.gz>  
tar zxvf webbench-1.5.tar.gz  
cd webbench-1.5

make && make install

**我们在make的时候会报错：**

[root@vm1 webbench-1.5]# make ctags \*.c /bin/sh: ctags: command not found  
make: [tags] 错误 127 (忽略)

 意思是说缺少ctags命令,我没有装了,运行以下命令即可

yum install ctags

**make的时候会报错：**

webbench.c:21:10: fatal error: rpc/types.h: No such file or directory

把webbench.c 这个文件下面的# include <rpc/types.h>删除就行了 这是写webbench的时候版本问题引发的 现在已经没有这个东西了

**make install时报错，如下**

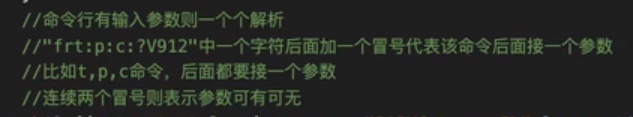
install: cannot create regular file '/usr/local/man/man1': No such file or directory make: \*\*\* [install] Error 1

解决：先创建man目录

mkdir /usr/local/man

再make install

**4、参数**  
只支持get请求，可设定head参数  
常用参数：  
-c 并发用户数  
-t 压测时长

1.   
    "webbench [选项参数]... URL\n"

" -f|--force 不等待服务器响应\n"

" -r|--reload 重新请求加载（无缓存）\n"

" -t|--time <sec> 运行时间，单位：秒，默认为30秒\n"

" -p|--proxy <server:port> 使用代理服务器发送请求\n"

" -c|--clients <n> 创建多少个客户端，默认为1个\n"

" -9|--http09 使用http0.9协议来构造请求\n"

" -1|--http10 使用http1.0协议来构造请求\n"

" -2|--http11 使用http1.1协议来构造请求\n"

" --get 使用GET请求方法\n"

" --head 使用HEAD请求方法\n"

" --options 使用OPTIONS请求方法\n"

" --trace 使用TRACE请求方法\n"

" -?|-h|--help 显示帮助信息\n"

" -V|--version 显示版本信息\n" );

1. *//相关参数选项的默认值*
2. int method = METHOD\_GET;
3. int clients = 1;
4. int force = 0; *//默认需要等待服务器相应*
5. int force\_reload = 0; *//默认不重新发送请求*
6. int proxyport = 80; *//默认访问80端口，http国际惯例*
7. char\* proxyhost = NULL; *//默认无代理服务器，因此初值为空*
8. int benchtime = 30; *//默认模拟请求时间*
9. **注意：***//定位url中主机名开始的位置*
10. *//比如 http://www.xxx.com/*
11. *//在主机名开始的位置找是否有'/'，若没有则非法*

**5、结果分析**

webbench -t 60 -c 100 http://www.baidu.com/

* 1

Speed 为每分钟多个个请求  
Requests 成功多少个请求，失败多少个请求

Webbench - Simple Web Benchmark 1.5

Copyright (c) Radim Kolar 1997-2004, GPL Open Source Software.

Benchmarking: GET http://www.baidu.com/

100 clients, running 60 sec.

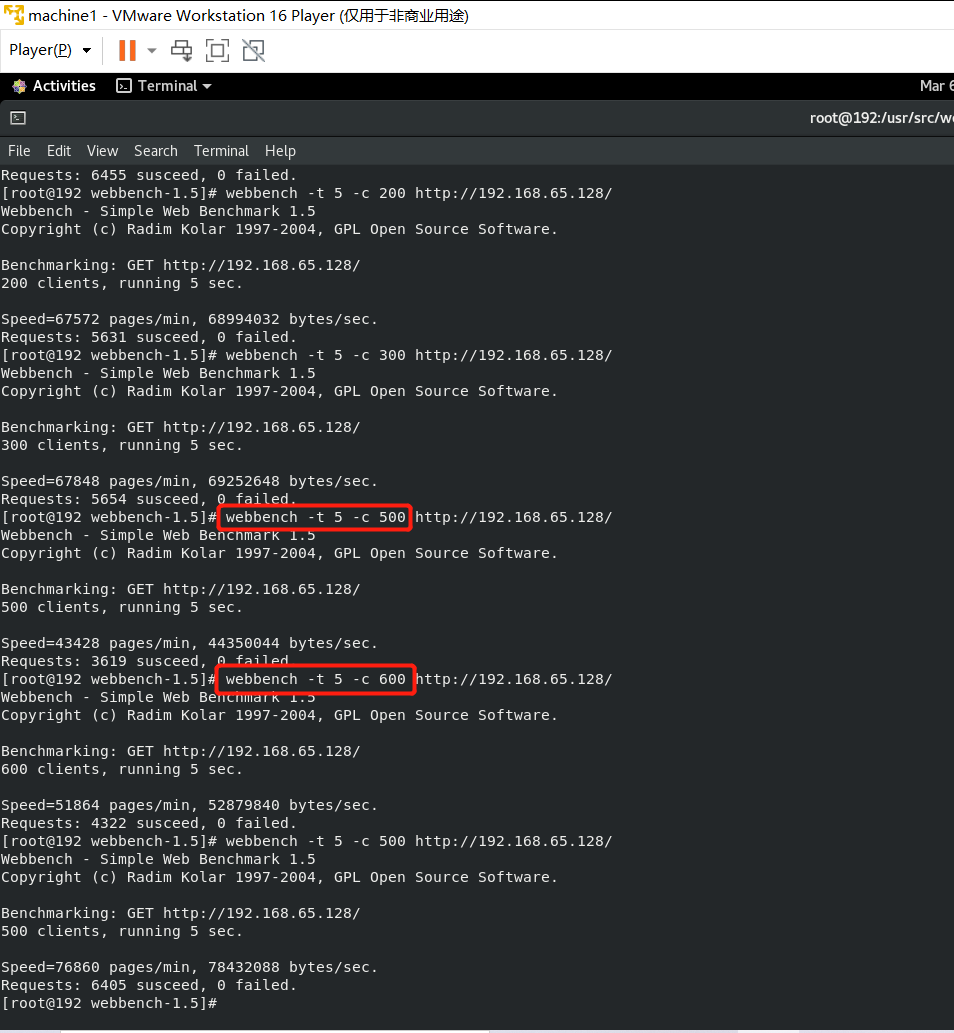
Speed=2643 pages/min, 5045450 bytes/sec.

Requests: 2641 susceed, 2 failed.

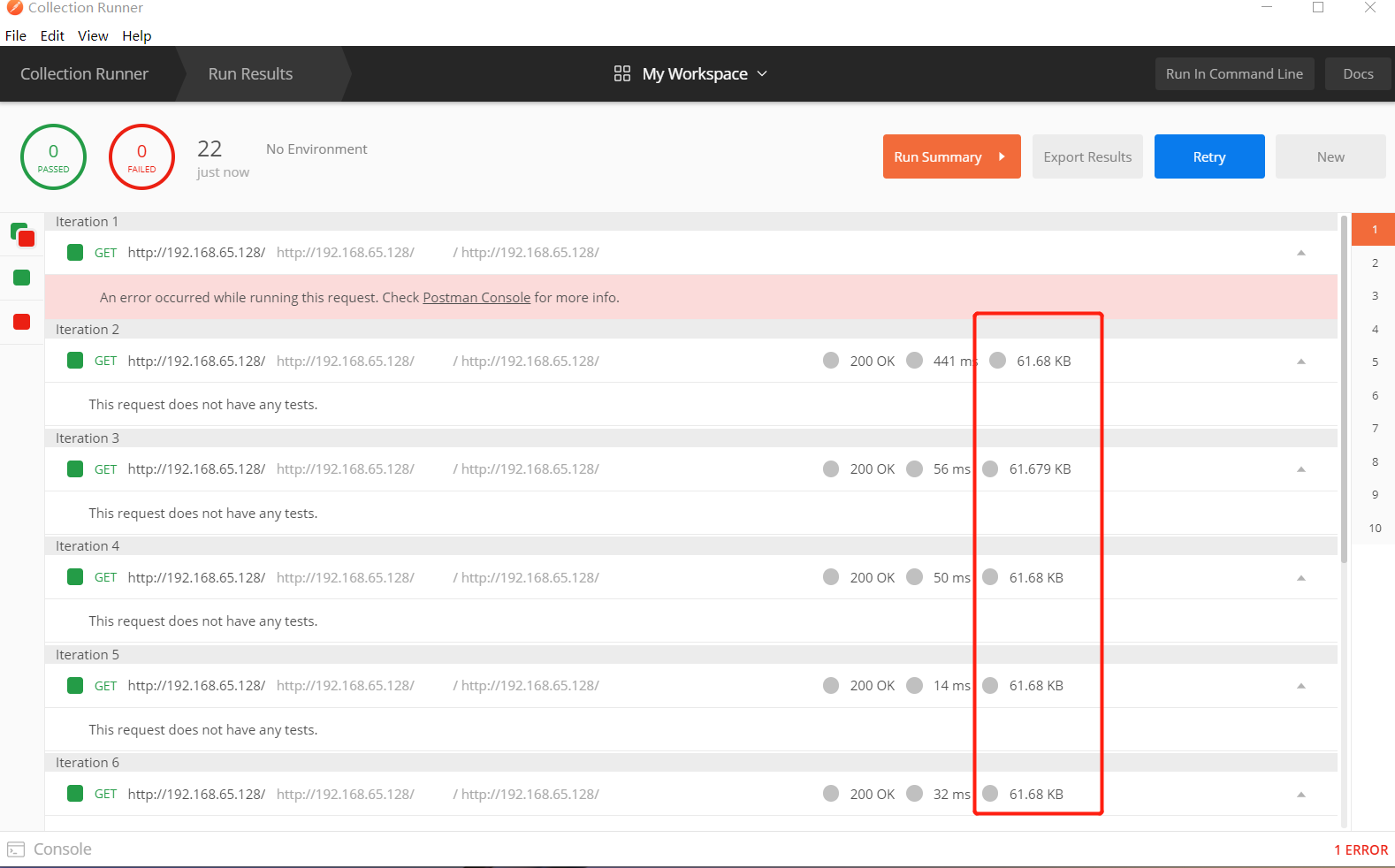
**WebBench使用**

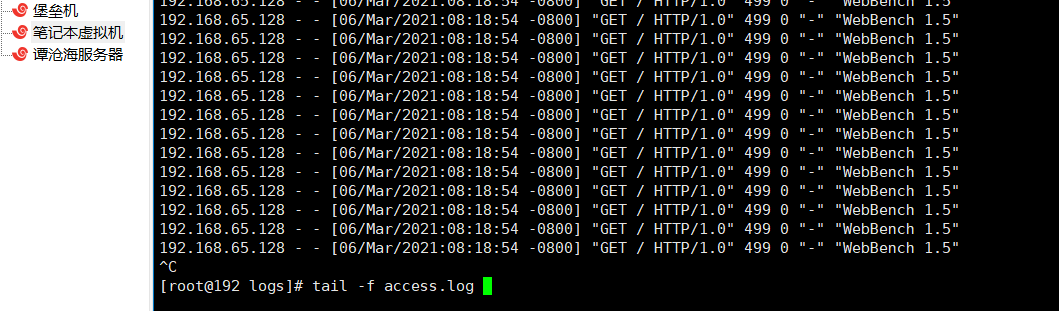
1. webbench -c 100 -t 60 http://test.domain.com/phpinfo.php

webbench -c 并发数 -t 运行测试时间(秒) URL

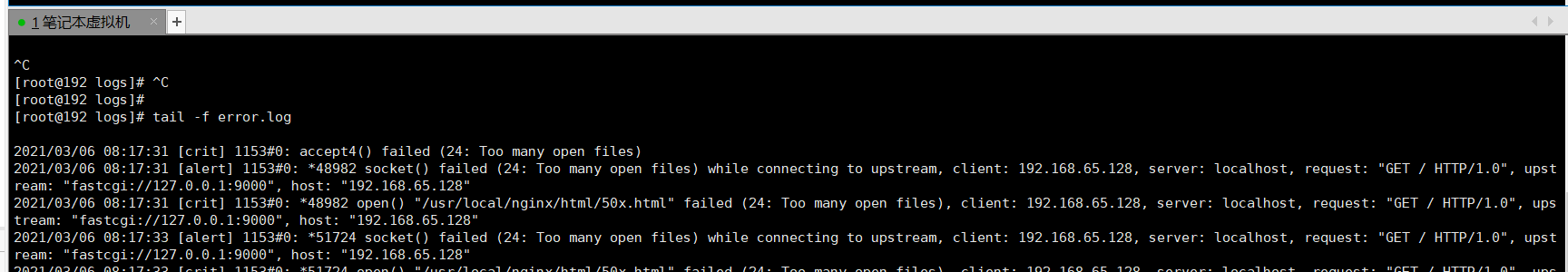


每次访问所占内存空间



通过测试，在A服务器执行webbench命令后，B服务器的CPU飙升至90%，同时B服务器操作卡顿，B服务器上的网站也打不开了。通过观察nignx日志，access.log发现大量的并发请求：  


通过error.log来观测当达到多少的时候会报错



如何防止短时间的大量并发导致服务器宕机。

### 4.总结：

1）**webbench 做压力测试时，该软件自身也会消耗CPU和内存资源，为了测试准确，请将 webbench 安装在别的服务器上。**

2）**压力测试工作应该放到产品上线之前，而不是上线以后**

3）**测试时并发应当由小逐渐加大，比如并发100时观察一下网站负载是多少、打开页面是否流畅，并发200时又是多少、网站打开缓慢时并发是多少、网站打不开时并发又是多少；**

4）**更详细的进行某个页面测试，如电商网站可以着重测试购物车、推广页面等，因为这些页面占整个网站访问量比重较大**。