

Nginx Openresty Build for Windows

关于如何正确在windows编译

简介

- 360企业安全
- 提供私有云安全管理平台产品
- 超过10000企业用户部署了私有云安全服务器
- Nginx Openresty for Windows（NOW）是私有云安全服务器的基础组件
- 服务器每天处理终端的安全策略下发和日志上传等
- 单服务器支撑最大安全终端数 35000 以上

NOW特性

- Openresty支持使用 Lua 代码做高效开发
- 利用 Nginx 异步事件模型，享受其强大并发能力
- 在 Windows 系统下进行移植，方便部署与使用，并与 MS 共同成长

与官网 Nginx 的对比

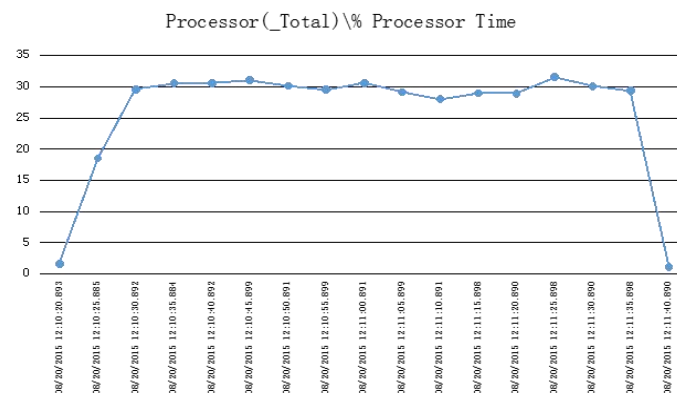
- 支持多进程的进程架构
- 支持共享内存的使用
- 支持更大的并发连接数（C10k以上）
- 支持 64bit 程序编译

性能对比（测试环境）

- OS: Windows Server 2008 R2 Standard X64
- RAM: 4GB
- CPU: Intel Core i5-3570 @3.40GHZ 物理4核心
- 测试参数: `ab -kc 200 -n 5000000 172.22.131.45/connect`
- `/connect` URI返回字符串“true”给请求终端

性能对比（官网Nginx）

- CPU

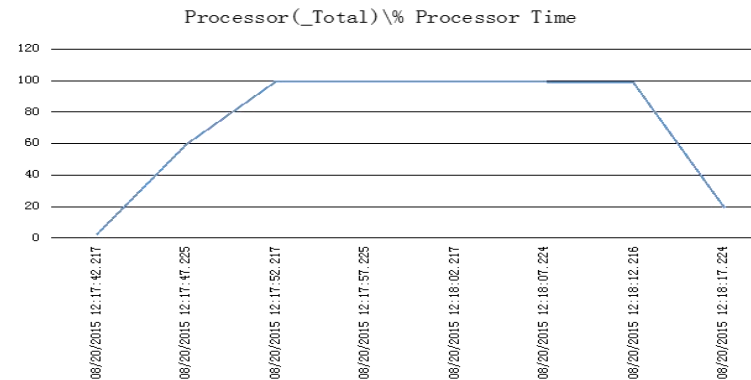


- RPS

```
Concurrency Level:      200
Time taken for tests:    70.718 seconds
Complete requests:      5000000
Failed requests:        0
Write errors:           0
Keep-Alive requests:    4950056
Total transferred:      744750429 bytes
HTML transferred:       20000004 bytes
Requests per second:    70703.33 [#/sec] (mean)
Time per request:       2.829 [ms] (mean)
Time per request:       0.014 [ms] (mean, across all concurrent requests)
Transfer rate:          10284.44 [Kbytes/sec] received
```

性能对比（NOW 32BIT）

•CPU

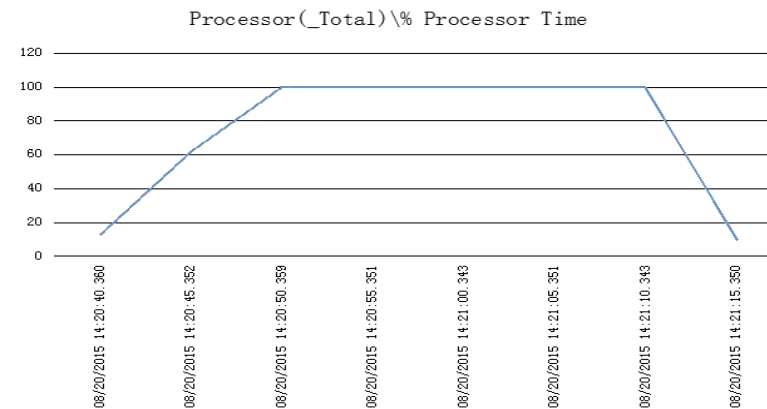


•RPS

```
Concurrency Level:      200
Time taken for tests:    26.117 seconds
Complete requests:      5000000
Failed requests:         0
Write errors:            0
Keep-Alive requests:    4950101
Total transferred:      749759500 bytes
HTML transferred:       25000300 bytes
Requests per second:    191446.02 [#/sec] (mean)
Time per request:       1.045 [ms] (mean)
Time per request:       0.005 [ms] (mean, across all concurrent requests)
Transfer rate:          28034.86 [Kbytes/sec] received
```

性能对比（NOW 64BIT）

•CPU



•RPS

```
Concurrency Level:      200
Time taken for tests:    26.253 seconds
Complete requests:      5000000
Failed requests:        0
Write errors:           0
Keep-Alive requests:    4950091
Total transferred:      749753455 bytes
HTML transferred:      25000100 bytes
Requests per second:    190452.06 [#/sec] (mean)
Time per request:       1.050 [ms] (mean)
Time per request:       0.005 [ms] (mean, across all concurrent requests)
Transfer rate:          27889.08 [Kbytes/sec] received
```


代码下载

- OpenResty
 - a fast web app server by extending nginx
 - ver 1.7.10.2
 - www.openresty.org
- Nginx
 - 高性能的HTTP和反向代理服务器
 - ver 1.8.0
 - dev 1.9.3
 - 需要通过 hg 下载到 windows 平台代码, url: <http://hg.nginx.org/nginx>
 - www.nginx.org

依赖库

- Zlib
 - 数据压缩用的函数库
 - ver 1.28
 - www.zlib.net
- PCRE
 - Perl正则库
 - ver 8.31
 - www.pcre.org
- OpenSSL
 - 强大的安全套接字层密码库
 - ver 1.0.1o
 - www.openssl.org

编译PCRE

- Openresty build configure
- 得到 makefile.msc
- Visual Studio x86 Command Prompt
- nmake -f makefile.msc
- 得到 pcre.lib

Visual Studio x86 Command Prompt

- 环境提供如下程序的调用：
- cl 代替 gcc
- link 代替 gcc
- nmake 代替 make
- mt (windows特有)



编译OpenSSL

- Visual Studio x86 Command Prompt
- perl Configure VC-WIN32 --prefix=c:\openssl
- ms\do_nasm, ms\do_ms
- nmake -f ms\ntdll.mak, 这个是 dll
- nmake -f ms\nt.mak, 这个是 lib
- 动态库静态库都编译:
 - 静态库用于nginx链接
 - 动态库用于其他动态库依赖。

编译Zlib

- Visual Studio x86 Command Prompt
- `nmake -f win32\Makefile.msc`
- `zlib.lib zdll.lib zlib.dll`
- 动态库静态库都编译：
 - 静态库用于nginx链接
 - 动态库用于其他动态库依赖。

编译Luajit

- Visual Studio x86 Command Prompt
- src\msvcbuild.bat static 回车
- 得到 lua51.lib
- 静态库用于nginx 编译 ngx_lua 的依赖。

修改configure

- 去掉 configure 里边 my @modules = 不需要的模块
- 一般保留：
 - devel_kit
 - echo
 - set-misc
 - ngx_postgres
 - ngx_lua
 - rds-json

编译前的配置

- `./configure --with-cc=cl --with-luajit --with-debug --with-pcre=/d/lib/pcre-8.31 --with-zlib=/d/lib/zlib-1.2.7 --with-openssl=/d/lib/openssl-1.0.1g/openssl --with-cc-opt="-D FD_SETSIZE=4096" --with-select_module`
 - `--with-cc=cl` 编译器
 - `--with-luajit` 核心，luajit 模块添加进来
 - `--with-pcre=` 依赖库 pcre 的路径
 - `--with-openssl=` 依赖库 openssl 的路径
 - `--with-zlib=` 依赖库 zlib 的路径
 - `--with-cc-opt="-D FD_SETSIZE=4096"` 把fd_set设置成4096了，默认是64
 - `--with-select_module` 编译 select 模块

修改Makefile

- CFLAGS = -O2 -W4 -nologo -MT -Zi -D FD_SETSIZE=4096 -DNDK_SET_VAR -DNDK_UPSTREAM_LIST -DNO_SYS_TYPES_H
 - -MT 静态链接运行期库
 - -Zi 编译时带debug信息
- \$(LINK) objs/nginx.res -link -verbose:lib -debug -base:0x0400000 -DYNAMICBASE:NO
 - -debug 这个选项可以输出pdb，方便调试
 - -base:0x0400000 修改基址
 - -DYNAMICBASE:NO 关闭映像地址随机化

编译Nginx

- Visual Studio x86 Command Prompt
- 在 openresty\nginx 目录下运行 nmake
- 输出在 openresty\nginx\nginx.exe

编译完成后处理

- 得到 nginx.exe
- Visual Studio x86 Command Prompt
- `mt.exe -manifest crt.xml -outputresource:nginx.exe;1`
- 用于保证运行期库路径正确选择当前目录下的运行期动态库。

Crt.xml 节选

- <dependency>
- <dependentAssembly>
- <assemblyIdentity type="win32" name="Microsoft.VC90.CRT" version="9.0.21022.8" processorArchitecture="x86"></assemblyIdentity>
- </dependentAssembly>
- </dependency>

单元测试

- ActivePerl 下载并安装
- 官方回归测试集 mdounin.ru/hg/nginx-tests
- Openresty回归测试集 github.com/openresty/test-nginx
- 公司QA同事用python写了一个脚本跑所有测试案例
- unittest.py

环境已经入库

- 不需要重复部署环境了
- 代码库: github.com/LomoX-Offical/nginx-openresty-windows.git
- Openresty 1.7.10.2
- 在 nginx 目录下运行 nmake 即可编译出来了

谢谢

张家柱 (zqzjz1982@gmail.com)