安装nginx+ngx_lua支持WAF防护功能

最近实在是被网上的攻击者搞烦了,防住了cc攻击又给你来垃圾评论,垃圾评论还没有彻底解决又给你来sql注入和cookie攻击,为了让自己轻松点,发现网上有个好的lua规则,好吧,那就用nginx+nginx_lua来抵挡下吧.

nginx lua模块淘宝开发的nginx第三方模块,它能将lua语言嵌入到nginx配置中,从而使用lua就极大增强了nginx的能力.nginx以高并发而知名,lua脚本轻便,两者的搭配堪称完美.

系统:centos 5.9

需要的软件:LuaJIT-2.0.0.tar.gz

nginx-1.4.4.tar.gz

v0.2.19(ngx_devel_kit)

v0.9.5rc2(nginx_lua模块)

1.安装前准备

```
yum -y install gcc gcc-c++  ncurses-devel libxml2-devel  \
openssl-devel curl-devel libjpeg-devel libpng-devel \
autoconf pcre-devel libtool-libs freetype-devel gd zlib-devel \
zip unzip wget crontabs iptables file bison cmake patch \
mlocate flex diffutils automake make \
readline-devel  glibc-devel glibc-static glib2-devel \
bzip2-devel gettext-devel libcap-devel logrotate ntp libmcrypt-devel \
GeoIP*
```

下载nginx

```
wget http://nginx.org/download/nginx-1.4.4.tar.gz
```

下载LuaJIT 2.0

```
wget http://luajit.org/download/LuaJIT-2.0.0.tar.gz
```

下载ngx_devel_kit并解压

```
wget --no-check-certificate https://github.com/simpl/ngx_devel_kit/archive/v0.2.19.tar.gz
tar zxf v0.2.19
```

下载nginx_lua_module并解压

```
wget --no-check-certificate https://github.com/chaoslawful/lua-nginx-module/archive/v0.9.5rc2.tar.gz
tar zxf v0.9.5rc2
```

下载ngx_lua_waf并解压

```
wget --no-check-certificate https://github.com/loveshell/ngx_lua_waf/archive/master.zip
unzip master
```

2.安装LuaJIT 2.0
tar zxf LuaJIT-2.0.0.tar.gz && cd LuaJIT-2.0.0
make && make install

注:lib和include是直接放在/usr/local/lib和usr/local/include

再来设置环境变量(这是给后面nginx编译的时候使用的):
vi /etc/profile
export LUAJIT_LIB=/usr/local/lib
export LUAJIT_INC=/usr/local/include/luajit-2.0
export LD_LIBRARY_PATH=/usr/local/lib/:$LD_LIBRARY_PATH

保存后执行:
source /etc/profile

3.安装nginx和nginx_lua
useradd -M -r -s /sbin/nologin www

tar zxf nginx-1.4.4.tar.gz && cd nginx-1.4.4

./configure --user=www --group=www --add-module=../ngx_devel_kit-0.2.19 \
--add-module=../lua-nginx-module-0.9.5rc2 --prefix=/usr/share/nginx --sbin-path=/usr/sbin/nginx \
--conf-path=/etc/nginx/nginx.conf --error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log --http-client-body-temp-path=/var/lib/nginx/tmp/client_body \
--http-proxy-temp-path=/var/lib/nginx/tmp/proxy --http-fastcgi-temp-path=/var/lib/nginx/tmp/fastcgi \
--pid-path=/var/run/nginx.pid --lock-path=/var/lock/subsys/nginx --with-http_secure_link_module \
--with-http_random_index_module --with-http_ssl_module --with-http_realip_module \
--with-http_addition_module --with-http_sub_module --with-http_dav_module --with-http_flv_module \
--with-http_gzip_static_module --with-http_stub_status_module --with-http_perl_module \
--with-http_geoip_module --with-mail --with-mail_ssl_modul

make -j2
make install

完了之后,还要执行一步:
ln -s /usr/local/lib/libluajit-5.1.so.2 /lib64/libluajit-5.1.so.2

不执行这步的,你使用nginx -V或者启动nginx会报下面的错误:
error while loading shared libraries: libluajit-5.1.so.2: cannot open shared object file: No such file or directory

然后创建下面3个文件夹:
mkdir -p /var/lib/nginx/tmp/client_body
mkdir -p /data/logs/hack/
chown -R www:www /data/logs/hack/
chmod -R 755 /data/logs/hack/
mkdir -p /etc/nginx/conf.d

把nginx-1.4.4启动脚本这篇文章里的启动脚本添加到:
/etc/init.d/nginx
chmod +x /etc/init.d/nginx
chkconfig --add nginx
chkconfig nginx on

4.配置ngx_lua_waf
mv ngx_lua_waf-master /etc/nginx/conf.d/waf

vi /etc/nginx/conf.d/waf/config.lua
修改RulePath = "/usr/local/nginx/conf/waf/wafconf/"为:
RulePath = "/etc/nginx/conf.d/waf/wafconf/"

修改logdir = "/usr/local/nginx/logs/hack/"为:
logdir = "/data/logs/hack/"

其他的根据你自己的需要进行修改.

config.lua配置文件说明:
RulePath = "/usr/local/nginx/conf/waf/wafconf/"
--规则存放目录
attacklog = "off"
--是否开启攻击信息记录，需要配置logdir
logdir = "/usr/local/nginx/logs/hack/"
--log存储目录，该目录需要用户自己新建，切需要nginx用户的可写权限

UrlDeny="on"
--是否拦截url访问
Redirect="on"
--是否拦截后重定向
CookieMatch = "on"
--是否拦截cookie攻击
postMatch = "on"
--是否拦截post攻击
whiteModule = "on"
--是否开启URL白名单
ipWhitelist={"127.0.0.1"}
--ip白名单，多个ip用逗号分隔
ipBlocklist={"1.0.0.1"}
--ip黑名单，多个ip用逗号分隔
CCDeny="on"
--是否开启拦截cc攻击(需要nginx.conf的http段增加lua_shared_dict limit 10m;)
CCrate = "100/60"
--设置cc攻击频率，单位为秒.
--默认1分钟同一个IP只能请求同一个地址100次
html=[[Please go away~~]]
--警告内容,可在中括号内自定义
备注:不要乱动双引号，区分大小写

5.修改nginx配置
vi /etc/nginx/nginx.conf
在nginx.conf里的http配置里添加:
lua_need_request_body on;
lua_package_path "/etc/nginx/conf.d/waf/?.lua";
lua_shared_dict limit 10m;
init_by_lua_file /etc/nginx/conf.d/waf/init.lua;
access_by_lua_file /etc/nginx/conf.d/waf/waf.lua;

保存后,使用/etc/init.d/nginx configtest检查下是否有错,然后启动nginx.

6.测试
在/var/www/vhosts/下创建个test.php文件,内容为test
使用curl来访问,当然前提是nginx做好了虚拟主机,这里就不介绍怎么做虚拟主机了.
curl http://localhost/test.php?id=../etc/passwd
返回的内容:
test

curl https://blog.slogra.com/test.php?id=../etc/passwd
返回的内容:
Please go away~~
说明规则是生效了的.

好了,剩下的还要再测试下才能上到服务器上.

https://blog.slogra.com/post-497.html