0x00 简介

在我们日常使用nginx中总是会发生各种各样的问题而这个时候日志的作用就派上用场了！

在nginx里面我们可以很灵活的为每个虚拟主机配置日志；并且还可以根据记录内容的不同，将nginx的日志分为：**访问日志** 与 **错误日志。**

通过 **访问路径** 我们可以知晓用户的ip地址，网站的哪些部分最受欢迎，用户的浏览时间，用户的来源地址等等。

利用好这两个日志我们甚至可以在网站给黑时，通过它来分析出 骇客 入侵我们系统的步骤！

此功能由**ngx\_http\_log\_module** 模块负责。

水分挤完了，开始干货把！ヽ(✿ﾟ▽ﾟ)ノ

0x01 访问日志主要参数

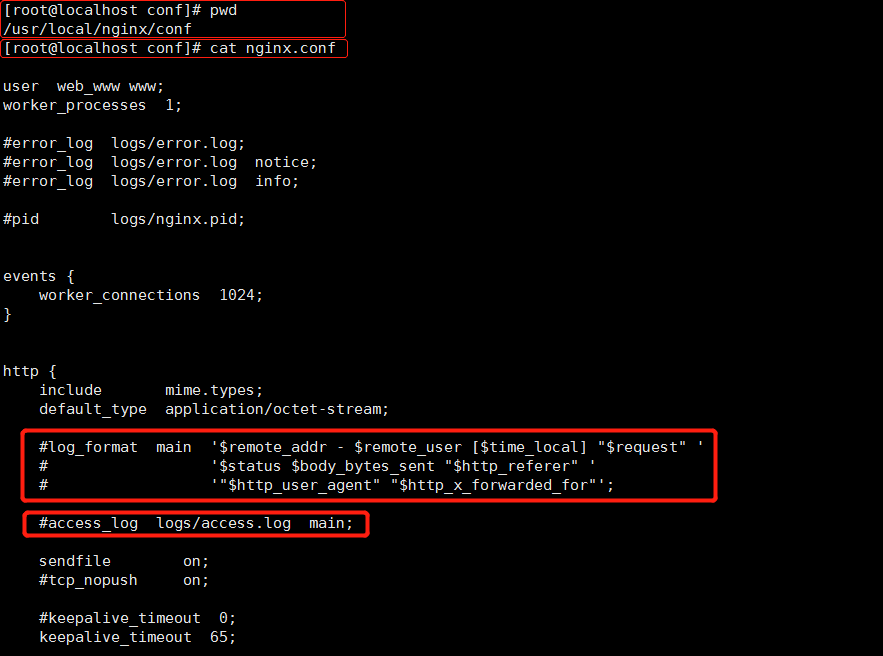
nginx 日志相关指令主要有以下几个

log\_format #用来定义记录日志的格式（可以定义多种日志格式，取不同名字即可）

access\_log #用来指定日至文件的路径及使用的何种日志格式记录日志

error\_log #用于记录客户端在访问Nginx时出现错误的记录，该错误日志格式不支持自定义功能，通过错误日志可以看出系统某个服务的性能瓶颈等。

0x02 查看默认的日志配置



2.1 log\_format格式讲解：

log\_format <格式标签> <日志格式参数>

$remote\_addr #记录访问网站的客户端地址

$remote\_user #远程客户端用户名

$time\_local #记录访问时间与时区

$request #用户的http请求起始行信息

$status #http状态码，记录请求返回的状态码，例如：200、301、404等

$body\_bytes\_sent #服务器发送给客户端的响应body字节数

$http\_referer #记录此次请求是从哪个连接访问过来的，可以根据该参数进行防盗链设置。

$http\_user\_agent #记录客户端访问信息，例如：浏览器、手机客户端等

$http\_x\_forwarded\_for #当前端有代理服务器时，设置web节点记录客户端地址的配置，此参数生效的前提是代理服务器也要进行相关的x\_forwarded\_for设置

2.2 access\_log格式讲解：

access\_log <保存的具体路径> <格式标签>;

2.3 error\_log格式讲解：

error\_log <保存的具体路径> <错误保存的级别>

常用的等级有：debug，info，notice，warn，error，crit；日志记录详情依次递减，

debug记录内容最详细，crit记录内容最简洁；

0x03 日志分割：

3.1 手动分割日志

使用mv命令将需要备份的日志移动到一个新的目录文件中，通常情况下，备份的文件使用日期命名，然后在重新生成一个空的日志文件存储新的记录。

3.2 自动分割日志

第一步抒写shell脚本:

我的脚本存放目录: /usr/local/nginx/logs/autolog.sh

```

# 相关代码

#!/bin/bash

# nginx日志路径 /usr/local/nginx/logs

LOGS\_PATH=/usr/local/nginx/logs

TODAY=$(date -d 'today' +%Y-%m-%d-%H-%M)

#echo $TODAY

# 移动日志并改名

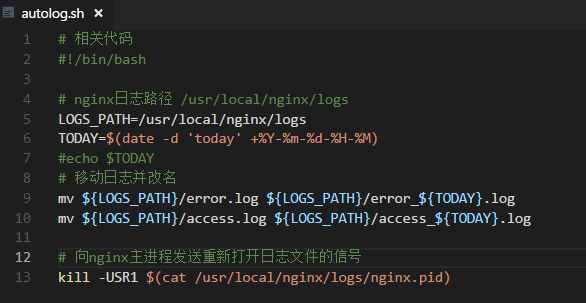
mv ${LOGS\_PATH}/error.log ${LOGS\_PATH}/error\_${TODAY}.log

mv ${LOGS\_PATH}/access.log ${LOGS\_PATH}/access\_${TODAY}.log

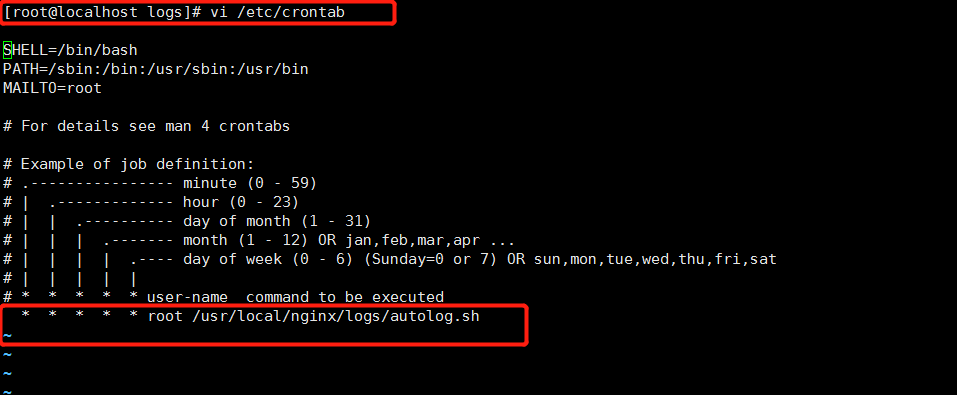
# 向nginx主进程发送重新打开日志文件的信号

kill -USR1 $(cat /usr/local/nginx/logs/nginx.pid)

```



第二步加入定时脚本：



然后等待几分钟就可以看到结果了