5月29日网站nginx运维故障复盘

Published on Jun 2, 2022 UTC by Wang Ye <https://wangye.org>

29日下午正好有时间例行网站应用代码升级维护，正好发现nginx HTTP3模块进行了更新，一直关注HTTP3的发展，网站一开始使用的是Cloudflare开发的HTTP3扩展，等nginx官方有了HTTP3支持后也是第一时间升级了，由于本站架构运行在Docker之上，所以这次升级也就改一下nginx代码库commit id，然后重新编译Docker镜像，一直都是这么升级，结果今天翻车了。

由于编译一次Docker镜像需要耗费较长时间，所以我会习惯性把一些该升级的组件进行一次升级：例如更新频率比较高的BoringSSL。这次还发现了ngx_brotli这个很长时间没有更新的扩展也有了更新记录，顺带也升级到最新commit。 完成编译后便直接将这个最新版本的nginx镜像推送至registry，然后服务器同步拉取（pull）完成更新容器更新，由于没有测试直接更新容器，结果炸了。受问题镜像同步的影响，共用nginx镜像的威言威语的网站也进入了不可用状态，好在他服务器上有前一个镜像的备份，遂执行了了回滚，约10分钟后威言威语网站恢复，但我自己的网站在更新的时候清除了镜像备份，所以只能重新上传镜像备份，由于网站服务器在境外，这个过程耗时较长，所以我的网站中断了长达2个多小时，算是较为严重的运维事故。

开始以为是更新的commit存在问题，后来发现不是，经过摸索得到具体的问题出在安全模块ModSecurity上，自构建的Docker镜像没有使用特定commit，导致每次安装的ModSecurity都是最新版本，`SpiderLabs/ModSecurity`在commit id `00483e4`更新上使用线程安全版本取代了单实例：

这导致在启用了此版本及后续版本的ModSecurity的nginx接收请求时进程崩溃，具体表现是出现以下错误：


```
[alert] 9#9: worker process 111 exited on signal 11
```

然后客户端接收到的是empty response，如果不启用ModSecurity扩展，则无此类问题。

解决的办法也很简单，直接固定ModSecurity的commit为前一个版本`606f572`即可，当然对于Docker编译来说通常使用下面的编译指令：

```
# 以下是代码片段，仅供参考
ENV MODSEC_COMMIT    606f5721c2b52e030f1028ae4e62f359a86443ce
RUN git clone -b v3/master https://github.com/SpiderLabs/ModSecurity \
  && git reset --hard $MODSEC_COMMIT \
  && git submodule init \
  && git submodule update \
  && ./build.sh \
  && ./configure --with-yajl --with-ssdeep --with-lmdb --with-geoip --
enable-examples=no\
  && make -j$(getconf _NPROCESSORS_ONLN)\
  && make -j$(getconf _NPROCESSORS_ONLN) install
```

其中需要注意的是增加了`git reset --hard commit_id`指令，该指令将强制使用特定`commit_id`的版本，这样问题可以算是临时解决了，至于为啥ModSecurity改Thread safe导致nginx进程崩溃还需要继续研究，目前先临时这么处理。

总结来说，以后对于这类镜像的编译和部署还是要先行测试，不能偷懒图省事了，因为隐秘的角落总是会有奇奇怪怪的Bug在伺机而动。

External References (Links)

威言威语
https://www.weisay.com/