软算法支持 SSL 卸载

部署使用说明(标准 nginx)

北京江南天安科技有限公司

2020 年 5 月

版本历史

文档版本	更新时间	修订人	文档更新说明
V1.00	2020-5-13	马晓艳	新建文档。
V1.01	2021-01-21	闫世超	软算法修订版

目录

1. SSL 卸载业务场景

通过使用 TASSL，实现 SSL 卸载；全面支持国密算法证书和国密 SSL 协议，符合监管合规要求。

SSL 卸载业务的典型部署方案：

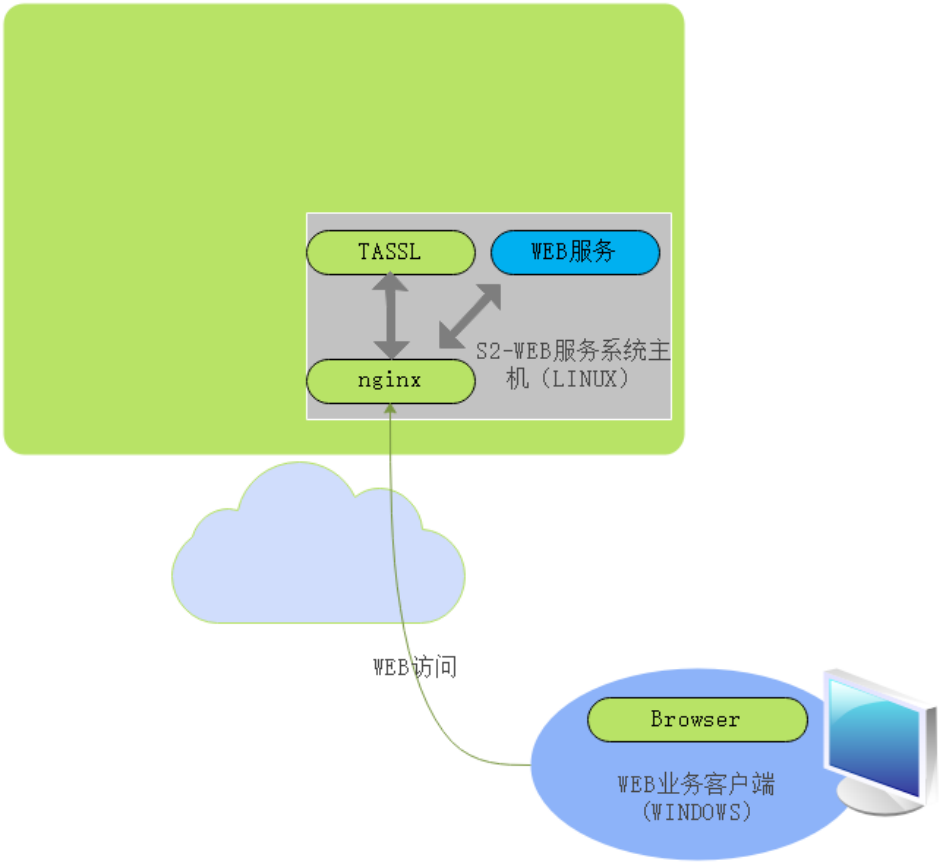


图 1 典型部署图

2. 所需资源

2.1. 硬件资源

如图 1 所示，用户所需的硬件资源环境如下：

类型	要求	说明
S2	Linux 64 位	用于部署用户的业务系统，及 TASSL+nginx

防火墙策略		S2 对外提供 TLS 端口服务；

2.2. 软件资源

1. TASSL

   https://github.com/jntass/TASSL-1.1.1k

2. nginx

   支持使用标准 nginx 配合 TASSL 构建基于国密 SSL 的 web server/反向代理，nginx 版本不应过低，本文档基于 nginx-1.14.2 测试

   https://nginx.org/en/download.html

3. 主要步骤

3.1. TASSL 安装与测试

3.1.1 编译与安装

1、 配置

```
./config --prefix=/opt/local
```

2、 编译及安装

```
make && make install
```

3、 添加 TASSL 链接库路径到 ldconfig 中

```
echo "/opt/local/lib" >> /etc/ld.so.conf && ldconfig
```

注：由于很多系统会自带 openssl 库，为避免程序链接错误的 openssl 动态库产生的异常，需添加 TASSL 库路径到 ld.so.conf 中，也可通过环境变量等方式

4、 检查链接库是否正确

```
ldd /opt/local/bin/openssl
```

```
[root@hsmyf_10 ~]# ldd /opt/local/bin/openssl
        linux-vdso.so.1 =>  (0x00007fffd3dff000)
        libssl.so.1.1 => /opt/local/lib/libssl.so.1.1 (0x00007f6d9b85e000)
        libcrypto.so.1.1 => /opt/local/lib/libcrypto.so.1.1 (0x00007f6d9b331000)
        libdl.so.2 => /lib64/libdl.so.2 (0x00007f6d9b12c000)
        libpthread.so.0 => /lib64/libpthread.so.0 (0x00007f6d9af0f000)
        libc.so.6 => /lib64/libc.so.6 (0x00007f6d9ab7b000)
        /lib64/ld-linux-x86-64.so.2 (0x00007f6d9bb1c000)
[root@hsmyf_10 ~]#
```

5、 检查 TASSL 版本

/opt/local/bin/openssl version -a

```
[root@hsmyf_10 ~]# /opt/local/bin/openssl version -a
OpenSSL 1.1.1k  Tassl 2.0.2 12 Nov 2021
built on: Thu Nov 25 09:44:17 2021 UTC
platform: linux-x86_64
options:  bn(64,64) rc4(16x,int) des(int) idea(int) blowfish(ptr)
compiler: gcc -fPIC -pthread -m64 -Wa,--noexecstack -Wall -O3 -DOPENSSL_USE_NODELETE -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUID_OBJ -DOP
ENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DSHA512_ASM -DKECCAK1600_A
SM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOLY1305_ASM -DNDEBUG
OPENSSLDIR: "/opt/local/ssl"
ENGINESDIR: "/opt/local/lib/engines-1.1"
Seeding source: os-specific
[root@hsmyf_10 ~]#
```

3.1.2 签发国密证书

按照实际业务需要产生 SM2 的证书申请文件并签发证书，测试阶段可以采用自签发证书，正式运行阶段建议从运营 CA 签发合格的服务器证书。

a)

cd /opt/local/tassl_demo/cert/

b)

./gen_sm2_cert.sh

注：执行脚本后当前目录下生成 certs 目录，CA.crt/CA.key 为根证书/密钥，SS.crt/SS.key 为服务端签名证书/密钥，SE.crt/SE.key 为服务端加密证书/密钥，CS.crt/CS.key 为客户端签名证书/密钥，CE.crt/CE.key 为客户端加密证书/密钥

3.1.3 s_server/s_client 测试国密 SSL

注:如使用 nginx 进行国密 SSL 卸载,可跳过此节。此节为使用 openssl 命令行 s_server/s_client 测试国密 SSL

a)

cd /opt/local/bin/

b) 启动服务端

./openssl s_server -cert ../tassl_demo/cert/certs/SS.crt -key ../tassl_demo/cert/certs/SS.key -cert_enc ../tassl_demo/cert/certs/SE.crt -key_enc ../tassl_demo/cert/certs/SE.key -accept 4433

c) 启动客户端

./openssl s_client -verify 10 -CAfile ../tassl_demo/cert/certs/CA.crt -connect 127.0.0.1:4433 -gmtls1_1

3.1.4 SSL demo 测试国密 SSL

注:如使用 nginx 进行国密 SSL 卸载，可跳过此节。此 demo 提供 SSL 服务端和客户端的最小示例工程，如需自行调用 TASSL 实现国密 SSL 功能，参考此 demo

a)

cd /opt/local/tassl_demo/ssl

b)

./mk.sh

c) 启动服务端

./sslsvr -p 4433 -sc ../cert/certs/SS.crt -sk ../cert/certs/SS.key -ec ../cert/certs/SE.crt -ek ../cert/certs/SE.key

```
[root@hsmyf_10 ssl]# ./sslsvr -p 4433 -sc ../cert/certs/SS.crt -sk ../cert/certs/SS.key -ec ../cert/certs/SE.crt -ek ../cert/certs/SE.
key
Start With
                Listening Port 4433
                Sign Cert ../cert/certs/SS.crt
                Sign Key ../cert/certs/SS.key
                Enc Cert ../cert/certs/SE.crt
                Enc Key ../cert/certs/SE.key
                CA Cert null
                Engine null
                Verify Peer False
```

其中：

-p：指定监听端口

-sc：指定 3.1.2 签发国密证书生成的签名证书

-sk：指定 3.1.2 签发国密证书生成的签名密钥

-ec：指定 3.1.2 签发国密证书生成的加密证书

-ek：指定 3.1.2 签发国密证书生成的加密密钥

d) 启动客户端

./sslcli -s 127.0.0.1:4433 --gmssl --verify -ca ../cert/certs/CA.crt

```
[root@hsmyf_10 ssl]# ./sslcli -s 127.0.0.1:4433 --gmssl --verify -ca ../cert/certs/CA.crt
Start With
                Server Address 127.0.0.1:4433
                Sign Cert null
                Sign Key null
                Enc Cert null
                Enc Key null
                CA Cert ../cert/certs/CA.crt
                Engine null
                Verify Peer True
SSL connection using GMTLSv1.1, ECC-SM4-SM3
对端证书信息:
证书: /C=CN/ST=BJ/L=HaiDian/O=Beijing JNTA Technology LTD./OU=BSRC of TASS/CN=server sign (SM2)
颁发者: /C=CN/ST=BJ/L=HaiDian/O=Beijing JNTA Technology LTD./OU=SORB of TASS/CN=Test CA (SM2)
recv 36 bytes : this message is from the SSL server!
[root@hsmyf_10 ssl]#
```

其中：

-s：指定服务器地址

--gmssl：使用国密 TLSv1.1

--verify：打开证书验证选项

-ca：指定 CA 证书

3.2. nginx 安装与测试

以下内容以 web server 方式介绍如何使用 nginx 搭建服务器并使用浏览器进行测试。

3.2.1 编译与安装

1、 配置

./configure --without-http_uwsgi_module --with-http_ssl_module --with-stream --with-stream_ssl_module --prefix=/opt/local/nginx

```
checking for OpenSSL library in /usr/local/ ... not found
checking for OpenSSL library in /usr/pkg/ ... not found
checking for OpenSSL library in /opt/local/ ... found
checking for zlib library ... found
creating objs/Makefile

Configuration summary
  + using system PCRE library
  + using system OpenSSL library
  + using system zlib library

  nginx path prefix: "/opt/local/nginx"
  nginx binary file: "/opt/local/nginx/sbin/nginx"
  nginx modules path: "/opt/local/nginx/modules"
  nginx configuration prefix: "/opt/local/nginx/conf"
  nginx configuration file: "/opt/local/nginx/conf/nginx.conf"
  nginx pid file: "/opt/local/nginx/logs/nginx.pid"
  nginx error log file: "/opt/local/nginx/logs/error.log"
  nginx http access log file: "/opt/local/nginx/logs/access.log"
  nginx http client request body temporary files: "client_body_temp"
  nginx http proxy temporary files: "proxy_temp"
  nginx http fastcgi temporary files: "fastcgi_temp"
  nginx http scgi temporary files: "scgi_temp"

[root@hsmyf_10 nginx-1.14.2]#
```

2、 编译及安装

make && make install

```
cp conf/nginx.conf '/opt/local/nginx/conf/nginx.conf.default'
test -d '/opt/local/nginx/logs' \
                || mkdir -p '/opt/local/nginx/logs'
test -d '/opt/local/nginx/logs' \
                || mkdir -p '/opt/local/nginx/logs'
test -d '/opt/local/nginx/html' \
                || cp -R html '/opt/local/nginx'
test -d '/opt/local/nginx/logs' \
                || mkdir -p '/opt/local/nginx/logs'
make[1]: Leaving directory `/root/nginx-1.14.2'
[root@hsmyf_10 nginx-1.14.2]#
```

3、 检查 nginx 依赖的 openssl 库

ldd /opt/local/nginx/sbin/nginx

```
[root@hsmyf_10 nginx-1.14.2]# ldd /opt/local/nginx/sbin/nginx
        linux-vdso.so.1 =>  (0x00007fff2afff000)
        libdl.so.2 => /lib64/libdl.so.2 (0x00007f6912b29000)
        librt.so.1 => /lib64/librt.so.1 (0x00007f6912921000)
        libpthread.so.0 => /lib64/libpthread.so.0 (0x00007f6912703000)
        libcrypt.so.1 => /lib64/libcrypt.so.1 (0x00007f69124cc000)
        libpcre.so.0 => /lib64/libpcre.so.0 (0x00007f691229f000)
        libssl.so.1.1 => /opt/local/lib/libssl.so.1.1 (0x00007f6912007000)
        libcrypto.so.1.1 => /opt/local/lib/libcrypto.so.1.1 (0x00007f6911b39000)
        libz.so.1 => /lib64/libz.so.1 (0x00007f6911923000)
        libc.so.6 => /lib64/libc.so.6 (0x00007f691158e000)
        /lib64/ld-linux-x86-64.so.2 (0x00007f6912d34000)
        libfreebl3.so => /lib64/libfreebl3.so (0x00007f691138b000)
[root@hsmyf_10 nginx-1.14.2]#
```

4、 检查 nginx 版本

/opt/local/nginx/sbin/nginx -V

```
[root@hsmyf_10 nginx-1.14.2]# /opt/local/nginx/sbin/nginx -V
nginx version: nginx/1.14.2
built by gcc 4.4.7 20120313 (Red Hat 4.4.7-17) (GCC)
built with OpenSSL 1.1.1k  Tassl 2.0.2 12 Nov 2021
TLS SNI support enabled
configure arguments: --without-http_uwsgi_module --with-http_ssl_module --with-stream --with-stream_ssl_module --prefix=/opt/local/ngi
nx
[root@hsmyf_10 nginx-1.14.2]#
```

3.2.2 修改并测试配置文件

1、 使用国密双证书

a)

cd /opt/local/nginx/conf

b) 修改 nginx.conf 为以下内容

开发机 - root@hsmyf_10:~/nginx/conf - Xshell 6 (Fre

```
1
2  user  root;
3  worker_processes  1;
4
5  events {
6      worker_connections  1024;
7  }
8
9  http {
10     include       mime.types;
11     default_type  application/octet-stream;
12
13     server {
14         listen       8020 ssl;
15         server_name  localhost;
16
17         #not use engine
18         ssl_certificate          /opt/local/tassl_demo/cert/certs/SS.crt;
19         ssl_certificate_key      /opt/local/tassl_demo/cert/certs/SS.key;
20         ssl_certificate          /opt/local/tassl_demo/cert/certs/SE.crt;
21         ssl_certificate_key      /opt/local/tassl_demo/cert/certs/SE.key;
22
23         location / {
24             root   html;
25             index  index.html index.htm;
26         }
27
28         error_page   500 502 503 504  /50x.html;
29         location = /50x.html {
30             root   html;
31         }
32     }
33 }
34
```

其中：
ssl_certificate：为 3.1.2 签发国密证书生成的签名证书
ssl_certificate_key：为 3.1.2 签发国密证书生成的签名密钥
ssl_certificate：为 3.1.2 签发国密证书生成的加密证书
ssl_certificate_key：为 3.1.2 签发国密证书生成的加密密钥

c)

/opt/local/nginx/sbin/nginx -t

```
[root@hsmyf_10 conf]# /opt/local/nginx/sbin/nginx -t
nginx: the configuration file /opt/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /opt/local/nginx/conf/nginx.conf test is successful
[root@hsmyf_10 conf]#
```

3.2.3 启动 nginx

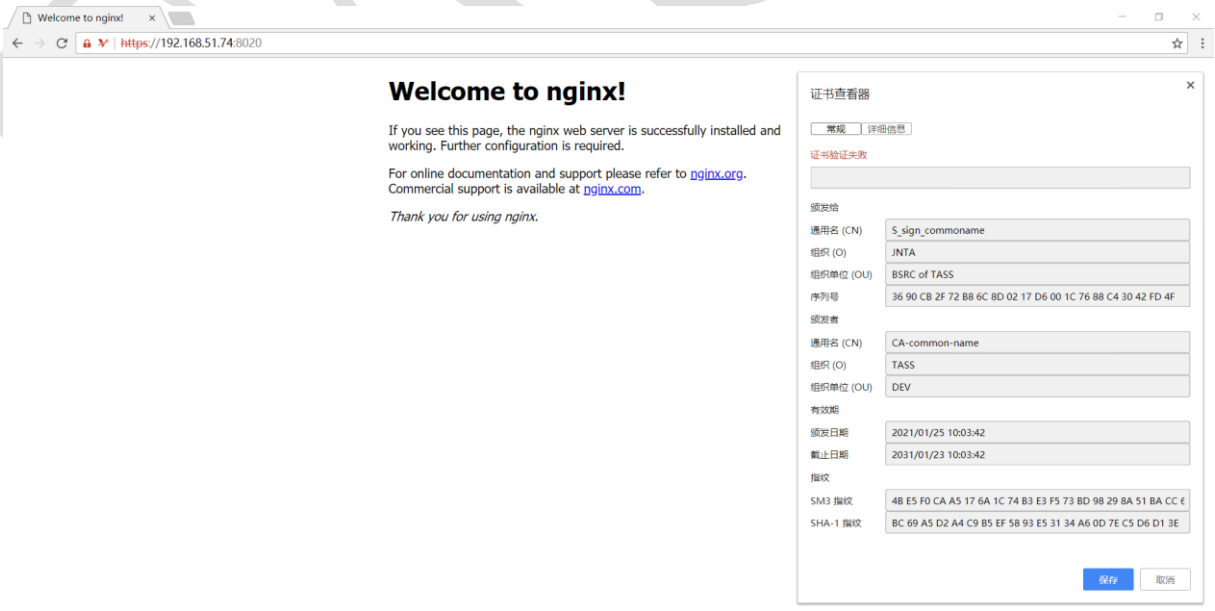
a)

/opt/local/nginx/sbin/nginx

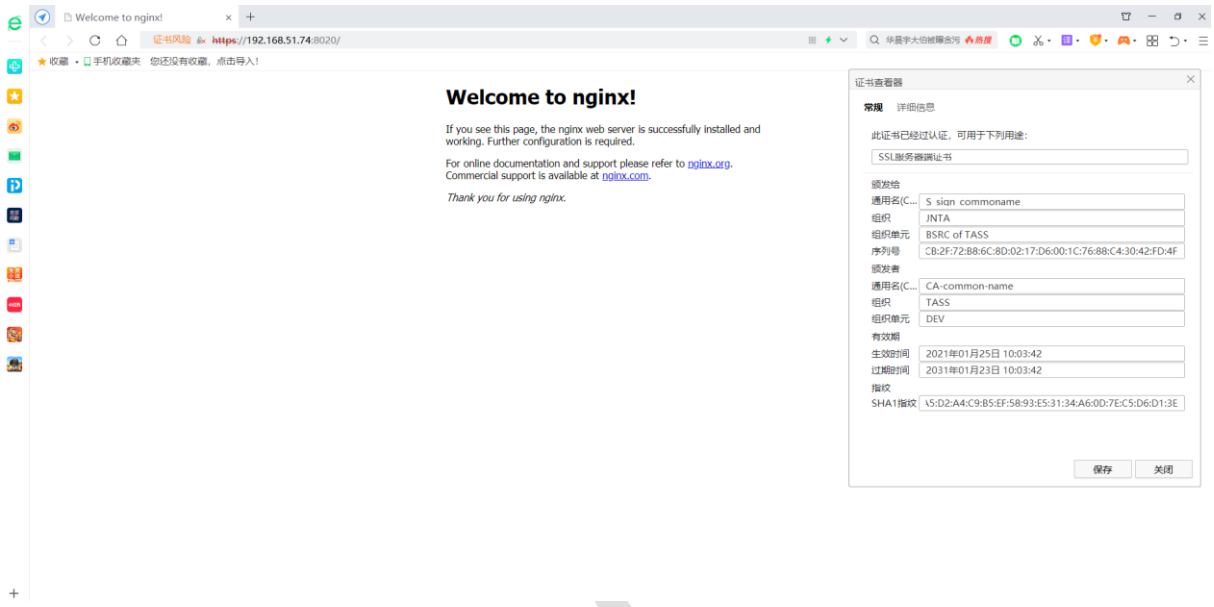
```
[root@hsmyf_10 conf]# /opt/local/nginx/sbin/nginx
[root@hsmyf_10 conf]#
```

3.2.4 测试验证 nginx

a) 密信浏览器



b) 360 浏览器



注：360 浏览器需启用国密 SSL 协议支持

3.3. 常见问题

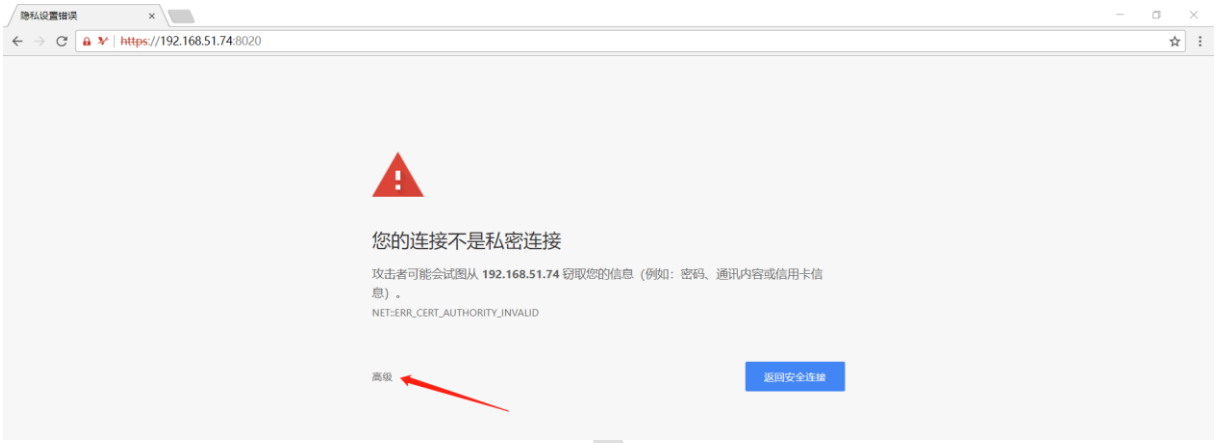
Q：360 浏览器测试国密 SSL，无法访问

A：确认是否”启用国密 SSL 协议支持”

Q：360 浏览器测试国密 SSL，显示服务器证书验证出错

A：切换”急速模式”

Q：浏览器测试验证，显示如下问题



A：证书不受信，选择高级->继续访问

Q：浏览器测试验证，显示证书风险

A：证书不受信；可将证书导入到浏览器可信证书或从受信 CA 签发证书