用 Tongsuo(铜锁) + Tengine 配合实现 SM2/RSA 双证书部署方案

Tengine 是一款开源、高性能、可扩展的 Web 服务器和反向代理服务器，它在 Nginx 的基础上，针对大访问量网站的需求，添加了很多高级功能和特性。。

Tengine 已对铜锁/Tongsuo（原名 BabaSSL）进行适配，并增加了对 NTLS（National TLS，也称国密 TLS）相关能力的支持。对于有使用国密算法进行安全通信协议需求的用户，可以直接使用 Tengine+Tongsuo 的组合。

铜锁/Tongsuo 是一个提供现代密码学算法和安全通信协议的开源基础密码库。

安装所使用的命令包

在服务器执行以下命令安装需要使用的软件命令包。

```
yum  -y  install  wget

yum  -y  install  gcc    gcc-c++

yum  -y  install  make

yum  -y  install  pcre    pcre-devel

yum  -y  install  gzip

yum  -y  install  zlib-devel

yum  -y  install  perl
```

下载安装 Tongsuo 和 Tengine

在服务器上执行以下命令，下载 Tongsuo 和 Tengine。

下载 Tongsuo 8.2.1 并解压。

```
wget  -c  https://github.com/BabaSSL/BabaSSL/archive/refs/tags/8.2.1.tar.gz

tar  -zxvf  Tongsuo-8.2.1.tar.gz     #解压 BabaSSL 压缩包
```

下载 Tengine 2.4.1 并解压。

```
wget  -c  https://tengine.taobao.org/download/tengine-2.4.1.tar.gz

tar  -zxvf  tengine-2.4.1.tar.gz
```

在 Tengine 解压目录下，执行以下命令，编译安装 Tengine 并关联 Tongsuo。
复制命令后，请参考下述说明修改对应的目录：

--prefix=/usr/local/tengine：指定 Tengine 安装目录为/usr/local/tengine，您可以指定其他目录。
--add-module=modules/ngx_tongsuo_ntls：加载 Tongsuo NTLS。Tengine 2.4.1 之前的版本需将 ngx_tongsuo_ntls 替换为 ngx_openssl_ntls。
--with-openssl=../Tongsuo-8.2.1：OpenSSL 替换为 Tongsuo 以实现 SM2。../Tongsuo-8.2.1 需替换为实际 Tongsuo 安装路径。

```
./configure  --prefix=/usr/local/tengine    \
```

```
--add-module=modules/ngx_tongsuo_ntls    \
--with-openssl=../Tongsuo-8.2.1      \
--with-openssl-opt="enable-ntls"    \
--with-http_ssl_module      \
--with-stream  \
--with-stream_ssl_module  \
--with-stream_sni
make  -j
make  install
```

在 Tengine 服务器安装证书

执行以下命令在 Tengine 服务器配置文件目录中创建证书存放目录。

```
cd  /usr/local/tengine/conf  #步骤二中指定的 Tengine 安装目录，请您可以根据实际配置调整。
mkdir  cert  #创建证书目录，命名为 cert。
```

将证书相关文件上传到 cert 目录（示例中为/usr/local/tengine/conf/cert）。

编辑 Tengine 配置文件 nginx.conf（示例中文件位于/usr/local/tengine/conf），修改与证书相关的配置。配置内容如下：

```
server  {
        listen  443  ssl;
        server_name  yourdomain.com;  #需修改为对应的网站域名

        #  启用 NTLS。Tengine 针对 BabaSSL 中的 NTLS 功能进行了适配，本文使用 BabaSSL 作为 Tengine 的底层密码库来实现通信加密的能力。
        enable_ntls  on;

        #  配置国密算法签名证书
        ssl_sign_certificate  cert/usercert.cet;  #需修改为实际证书文件路径
        ssl_sign_certificate_key  cert/usercert.key;  #需修改为实际私钥文件路径

        #  配置国密算法加密证书
        ssl_enc_certificate  cert/enccert.cer;  #需修改为实际证书文件路径
        ssl_enc_certificate_key  cert/enccert.key;  #需修改为实际私钥文件路径
```

```
        #  配置 RSA 算法证书
        ssl_certificate  cert/server_rsa.pem;  #需修改为实际证书文件路径
        ssl_certificate_key  cert/server_rsa.key;  #需修改为实际私钥文件路径

        ssl_session_cache  shared:SSL:1m;
        ssl_session_timeout  5m;

        #  配置加密套件#  ssl_ciphers  HIGH:  !aNULL:  !MD5;
        ssl_ciphers  ECC-SM2-SM4-CBC-SM3:ECC-SM2-SM4-GCM-SM3:ECDHE-SM2-SM4-CBC-SM3:ECDHE-SM2-SM4-GCM-SM
3:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-SHA:AES128-GCM-SHA256:AES128-SHA256:AES128-SHA:ECDHE-RSA
-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:AES256-GCM-SHA384:AES256-SHA256:AES256-
SHA:ECDHE-RSA-AES128-SHA256:!aNULL:!eNULL:!RC4:!EXPORT:!DES:!3DES:!MD5:!DSS:!PKS;
        ssl_protocols  TLSv1  TLSv1.1  TLSv1.2  TLSv1.3;
        ssl_prefer_server_ciphers  on;

        location  /  {
                root  html;
                index  index.html  index.htm;
        }
}
```

执行如下命令，重启 Tengine 服务

```
cd  /usr/local/tengine/sbin  #进入 Tengine 服务的可执行目录。
./nginx  -s  reload    #重新载入配置文件。
```

验证安装是否成功

证书安装完成后，您可通过访问证书的绑定域名验证该证书是否安装成功。

```
https://yourdomain.com      #需要将 yourdomain 替换成证书绑定的域名。
```

国密算法证书验证需要使用国密浏览器测试。

RSA 算法证书使用 Google、Firefox、Edge 等浏览器测试。

通过浏览器访问网站时，服务器会根据浏览器对国密算法的支持情况自动切换使用相应的加密方式：

如果浏览器支持国密算法，则服务器会选择国密 SM2 算法证书进行 HTTPS 加密通信。

若使用的浏览器不支持国密算法（如大多数国际主流浏览器），则服务器将使用 RSA 算法证书进行 HTTPS 加密通信。