使用openssl创建自签名的证书链

CodingCode 关注
2021.08.06 01:06:49 字数 352 阅读 100

> 第一步：生成自签名的根证书

```
$ openssl req -x509 \
    -newkey rsa \
    -outform PEM -out tls-rootca.pem \
    -keyform PEM -keyout tls-rootca.key.pem \
    -days 35600 \
    -nodes \
    -subj "/C=cn/O=mycomp/OU=mygroup/CN=rootca"
```

结果是生成根证书文件：`tls-rootca.pem`和`tls-rootca.key.pem`

查看根证书的内容：

```
$ openssl x509 -text -noout -in tls-rootca.pem
    ...
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=cn, O=mycomp, OU=mygroup, CN=rootca
        Validity
            Not Before: Aug  5 15:44:47 2021 GMT
            Not After : Jan 24 15:44:47 2119 GMT
        Subject: C=cn, O=mycomp, OU=mygroup, CN=rootca
        Subject Public Key Info:
            ...
            X509v3 Basic Constraints:
                CA:TRUE
    ...
```

> 第二步，生成中间证书

2.1 生成csr和key文件

```
$ openssl req -newkey rsa:2048 \
  -outform PEM -out tls-intermca.csr \
  -keyform PEM -keyout tls-intermca.key.pem \
  -nodes \
  -extensions v3_ca \
  -config /etc/pki/tls/openssl.cnf \
  -subj "/C=cn/O=mycomp/OU=mygroup/CN=intermca"
```

这一步的结果是生成`tls-intermca.csr`和`tls-intermca.key.pem`

2.2 用rootca对intermca进行签发

```
$ openssl x509 \
  -req -days 365 \
  -in tls-intermca.csr \
  -out tls-intermca.pem \
  -CA tls-rootca.pem \
  -CAkey tls-rootca.key.pem \
  -CAcreateserial \
  -extensions v3_ca \
  -extfile /etc/pki/tls/openssl.cnf
```

这一步的结果是生成`tls-rootca.srl`和`tls-intermca.pem`，其中`tls-rootca.srl`是rootca签发的serial文件，先不用管它；我们关注生成的intermca证书文件`tls-intermca.pem`。

```
$ openssl x509 -text -noout -in tls-intermca.pem
    ...
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=cn, O=mycomp, OU=mygroup, CN=rootca
        Validity
            Not Before: Aug  5 16:23:45 2021 GMT
            Not After : Aug  5 16:23:45 2022 GMT
        Subject: C=cn, O=mycomp, OU=mygroup, CN=intermca
        Subject Public Key Info:
        ...
        X509v3 extensions:
            ...
            X509v3 Basic Constraints:
                CA:TRUE
    ...
```

可以看到intermca证书已经是被rootca证书签过了。

> 第三步，生成叶子证书

3.1 生成csr和key文件

```
$ openssl req -newkey rsa:2048 \
    -outform PEM -out tls-cert.csr \
    -keyform PEM -keyout tls-cert.key.pem \
    -nodes \
    -reqexts SAN \
    -extensions v3_req \
    -config <(cat /etc/pki/tls/openssl.cnf <(printf "\n[SAN]\nsubjectAltName=DNS:server.mycomp.com, DNS:localhost, DN
    -subj "/C=cn/O=mycomp/OU=mygroup/CN=server"
```

这一步的结果是生成tls-cert.csr和tls-cert.key.pem

3.2 用intermca对cert进行签发

```
$ openssl x509 -req -days 365 \
    -in tls-cert.csr \
    -out tls-cert.pem \
    -CA   tls-intermca.pem \
    -CAkey tls-intermca.key.pem \
    -CAcreateserial \
    -extensions SAN \
    -extfile <(cat /etc/pki/tls/openssl.cnf <(printf "\n[SAN]\nsubjectAltName=DNS:server.mycomp.com, DNS:localhost, D
```

这一步的结果是生成tls-intermca.srl和tls-cert.pem，其中tls-intermca.srl是intermca签发的serial文件，也先不用管它；我们关注生成的cert证书文件tls-cert.pem。

```
$ openssl x509 -text -noout -in tls-cert.pem
    ...
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=cn, O=mycomp, OU=mygroup, CN=intermca
        Validity
            Not Before: Aug  5 16:48:40 2021 GMT
            Not After : Aug  5 16:48:40 2022 GMT
        Subject: C=cn, O=mycomp, OU=mygroup, CN=server
        Subject Public Key Info:
        ...
        X509v3 extensions:
            X509v3 Subject Alternative Name:
                DNS:server.mycomp.com, DNS:localhost, DNS:127.0.0.1
        ...
```

这里可以看到tls-cert.pem证书已经是被intermca证书签过了。

> 第四步，来验证证书链

验证intermca：

```
$ openssl verify -verbose -CAfile tls-rootca.pem tls-intermca.pem
tls-intermca.pem: OK
```

验证叶子证书：

CodingCode 关注
总资产22

Golang使用github.com/spf13/cobra处理多级命令
阅读 50

jq使用外部文件作为--arg参数值
阅读 12

如何把commit从一个repos拷贝到另一个repos
阅读 14

推荐阅读

二进制安装-k8s高可用集群03-kubectl命令行工具
阅读 204

Http1.0、Http1.1和Http2.0以及Https
阅读 197

Doker开启远程安全访问(x.x.x:2376)
阅读 224

TLS握手
阅读 103

https 协议交互报文解析
阅读 388

```
$ openssl verify -verbose -CAfile tls-intermca.pem tls-cert.pem
tls-cert.pem: C = cn, O = mycomp, OU = mygroup, CN = intermca
error 2 at 1 depth lookup:unable to get issuer certificate

$ openssl verify -verbose -CAfile tls-rootca.pem tls-cert.pem
tls-cert.pem: C = cn, O = mycomp, OU = mygroup, CN = server
error 20 at 0 depth lookup:unable to get local issuer certificate
```

可见不管是rootca还是intermca都不能单独验证叶子证书，需要合起来验证：

```
$ openssl verify -CAfile tls-rootca.pem -untrusted tls-intermca.pem tls-cert.pem
tls-cert.pem: OK
```

或者：

```
$ openssl verify -verbose -CAfile <(cat tls-intermca.pem tls-rootca.pem) tls-cert.pem
tls-cert.pem: OK
```

还能这样把intermca和cert打成一个bundle，然后用rootca验证：

```
$ cat tls-intermca.pem tls-cert.pem > tls-bundle.pem
$ openssl verify -CAfile tls-rootca.pem tls-bundle.pem
tls-bundle.pem: OK
```


OpenSSL解决ArcGIS软件部署的证书问题(二)--创建根证书和服务器证书
一、前言 成话 受到12306的启发，我们自己从根证书起搭建一条完整的信任链。本章中我们使用OpenSSL生成根证书...


openssl生成证书的原理及使用场景
1. 基本原理 参考：http://www.cnblogs.com/phpinfo/archive/2013/08...


半自动化创建CA和申请证书
1 概述 本文之所以称之为半自动化，是因为证书的申请并非日常工作，只是一段时间才需要申请，同时，在创建证书和办法证...

OpenSSL简介及证书创建API使用
1. OpenSSL 1.1 OpenSSL简介 SSL，Security Socket Layer,是一个安全传...


如何创建自签名的 SSL 证书
先把用到的命令行放上来方便备查：如不需要私钥密码，则删掉 -des3 参数即可 自签名：私有 CA 签名，注：...