let's encrypt免费ssl证书部署

本文 PDF 版本：let's encrypt免费ssl证书部署.pdf

0.前言

单域名证书使用 certbot 工具生成：https://certbot.eff.org/
通配符证书使用 acme.sh 脚本生成：https://github.com/Neilpang/acme.sh

1.单域名证书部署

1.1.安装依赖

```
# cat /etc/redhat-release  # 查看系统版本
CentOS Linux release 7.1.1503 (Core)
# 打开 https://certbot.eff.org/  选择对应版本。这里是nginx + centos7
# yum -y install yum-utils  # 按照依赖
# yum-config-manager --enable rhui-REGION-rhel-server-extras rhui-REGION
-rhel-server-optional
# yum install python2-certbot-nginx # 安装插件
```

1.2.生成证书：

方法一： 配置 nginx 插件，并生成证书。这两参数知道配置路径及nginx命令路径

```
# certbot --nginx-server-root /usr/local/nginx/conf --nginx-ctl /usr/loc
al/nginx/sbin/nginx
```

方法二：指定域名网站目录，生成证书

```
# certbot certonly --webroot -w /data/www/chopper/frontend/public
```

方法三：服务器没有跑web服务，也是可以生成的。这种方法工具会启动自带的web服务，如果80被占用了，命令会失败

```
# certbot certonly --standalone
```

不管哪个方法，刚开始的时候，都会要填一个邮箱接收通知，以及同意协议。

1.3.添加自动更新ssl服务

通过方法一，也就是安装 web 服务插件，只需要执行 `certbot renew` 即可，这个命令，会检测证书是否过期，并通过 web 服务插件重载配置。
这里用的步骤2的方法二，所以呢要加 nginx 的重载配置命令，更新证书脚本：

```
#!/usr/bin/env bash

certbot renew
/usr/local/nginx/sbin/nginx -s reload
```

添加定时任务：

```
# 每天检查域名ssl证书是否过期，默认是到期前30天更新
0 1 * * * /data/sh/renew_ssl.sh >>/data/logs/crontab/renew_ssl.log 2>&1
```

说明：

```
# ls /etc/letsencrypt/
accounts archive csr keys live options-ssl-nginx.conf renewal renewal-ho
oks ssl-dhparams.pem
```

生成的证书存在了 archive 目录，live目录里面文件软链接到 archive。renewal 目录，存放了可以更新的域名，并且每个域名还可以配置更新规则。

如下是其中一个域名的更新配置。默认是到期前30天进行更新证书。定时任务可以每天、或者每周执行一次。

```
# cat /etc/letsencrypt/renewal/www.xsy.me.conf
# renew_before_expiry = 30 days
version = 0.25.1
archive_dir = /etc/letsencrypt/archive/www.xsy.me
cert = /etc/letsencrypt/live/www.xsy.me/cert.pem
privkey = /etc/letsencrypt/live/www.xsy.me/privkey.pem
chain = /etc/letsencrypt/live/www.xsy.me/chain.pem
fullchain = /etc/letsencrypt/live/www.xsy.me/fullchain.pem

# Options used in the renewal process
[renewalparams]
authenticator = webroot
```

```
installer = None
account = 2cc116157725dacc214b9ae588752786
webroot_path = /data/www/chopper/frontend/public,
[[webroot_map]]
www.xsy.me = /data/www/chopper/frontend/public
```

2.通配符证书部署

这里的通配符，是指统配二级域名，所有二级域名都用一份证书。三级或以上的是统配不了的。
这里以dnspod 为例，如果域名在阿里云或者狗爹上解析，基本也是一样的。
三个步骤：

- 1、申请dnspod api token
- 2、下载安装 acme.sh 脚本
- 3、通过 acme.sh 脚本生成证书

下面是具体操作记录。

2.1.获取api操作需要的token

如果使用狗爹的 api，则需要 key 和 secret，在 https://developer.godaddy.com/keys/ 申请。
如果是阿里云，登录账号，移动鼠标到头像 > 访问控制 > 人员管理 > 用户 > 新建用户 > 点击用户 > 创建新的AccessKey（下载csv） > 权限管理 > 添加权限 AliyunDNSFullAccess

这里使用的是 dnspod api，申请的是 DP_Id 和 DP_Key：在 dnspod 账号的 用户中心 》安全设置 下申请。

```
export DP_Id="45824"
export DP_Key="394f4762a1232cf5aad403c2d5c0dabb"
```

可以把上面export 加到 `.acme.sh/dnsapi/dns_dp.sh` 脚本里面，或者加到/etc/profile里面。

2.2.安装脚本

安装依赖工具

```
yum -y install curl cron socat
```

安装acme.sh脚本，遇到以下报错

```
[root@txvps sh]# curl https://get.acme.sh | sh
......
[2018年 09月 25日 星期二 17:18:34 CST] Downloading https://github.com/Neilpang/acme.sh/archive/master.tar.gz
[2018年 09月 25日 星期二 17:18:34 CST] Please refer to https://curl.haxx.se/libcurl/c/libcurl-errors.html for error code: 35
[2018年 09月 25日 星期二 17:18:34 CST] Download error.
```

解决办法：`yum install nss`再次安装:

```
[root@txvps sh]# curl https://get.acme.sh | sh
  % Total % Received % Xferd Average Speed Time Time Time Current
                                 Dload Upload Total Spent Left Speed
100 705 100 705 0 0 704 0 0:00:01 0:00:01 --:--:-- 705
  % Total % Received % Xferd Average Speed Time Time Time Current
                                 Dload Upload Total Spent Left Speed
100 164k 100 164k 0 0 455k 0 --:--:-- --:--:-- --:--:-- 456k
[2018年 09月 25日 星期二 17:23:23 CST] Installing from online archive.
[2018年 09月 25日 星期二 17:23:23 CST] Downloading https://github.com/Neilpang/acme.sh/archive/master.tar.gz
[2018年 09月 25日 星期二 17:23:26 CST] Extracting master.tar.gz
[2018年 09月 25日 星期二 17:23:26 CST] Installing to /root/.acme.sh
[2018年 09月 25日 星期二 17:23:26 CST] Installed to /root/.acme.sh/acme.sh
[2018年 09月 25日 星期二 17:23:26 CST] Installing alias to '/root/.bashrc'
[2018年 09月 25日 星期二 17:23:26 CST] OK, Close and reopen your terminal to start using acme.sh
[2018年 09月 25日 星期二 17:23:26 CST] Installing alias to '/root/.cshrc'
[2018年 09月 25日 星期二 17:23:26 CST] Installing alias to '/root/.tcshrc'
[2018年 09月 25日 星期二 17:23:26 CST] Installing cron job
[2018年 09月 25日 星期二 17:23:26 CST] Good, bash is found, so change the shebang to use bash as preferred.
[2018年 09月 25日 星期二 17:23:26 CST] OK
[2018年 09月 25日 星期二 17:23:26 CST] Install success!
```

安装脚本成功后：

- 在当前用户家目录下多了一个`.acme.sh`目录。后续生成的证书也会存放到这个目录下，以域名为子目录名称存放。
- 添加了定时任务，自动更新证书
- 添加了别名

2.3.生成证书

```
# 使用dnspod api
[root@txvps ~]# .acme.sh/acme.sh --issue --dns dns_dp -d xsy.me -d *.xsy
.me
[2018年 09月 25日 星期二 17:32:01 CST] Registering account
[2018年 09月 25日 星期二 17:32:02 CST] Registered
[2018年 09月 25日 星期二 17:32:02 CST] ACCOUNT_THUMBPRINT='_cnIv56YeqD9WRg
HyeElsqYl8hfkZFVMSLHYtqseUZk'
[2018年 09月 25日 星期二 17:32:02 CST] Creating domain key
[2018年 09月 25日 星期二 17:32:02 CST] The domain key is here: /root/.acme
.sh/xsy.me/xsy.me.key
[2018年 09月 25日 星期二 17:32:02 CST] Multi domain='DNS:xsy.me,DNS:*.xsy.
me'
[2018年 09月 25日 星期二 17:32:02 CST] Getting domain auth token for each
domain
[2018年 09月 25日 星期二 17:32:03 CST] Getting webroot for domain='xsy.me'
[2018年 09月 25日 星期二 17:32:03 CST] Getting webroot for domain='*.xsy.m
e'
[2018年 09月 25日 星期二 17:32:03 CST] Found domain api file: /root/.acme.
sh/dnsapi/dns_dp.sh
[2018年 09月 25日 星期二 17:32:04 CST] Adding record
[2018年 09月 25日 星期二 17:32:04 CST] Found domain api file: /root/.acme.
sh/dnsapi/dns_dp.sh
[2018年 09月 25日 星期二 17:32:04 CST] Adding record
[2018年 09月 25日 星期二 17:32:04 CST] Sleep 120 seconds for the txt recor
ds to take effect
[2018年 09月 25日 星期二 17:34:06 CST] Verifying:xsy.me
[2018年 09月 25日 星期二 17:34:08 CST] Success
[2018年 09月 25日 星期二 17:34:08 CST] Verifying:*.xsy.me
[2018年 09月 25日 星期二 17:34:11 CST] Success
[2018年 09月 25日 星期二 17:34:11 CST] Removing DNS records.
[2018年 09月 25日 星期二 17:34:13 CST] Verify finished, start to sign.
[2018年 09月 25日 星期二 17:34:14 CST] Cert success.
-----BEGIN CERTIFICATE-----
......
-----END CERTIFICATE-----
[2018年 09月 25日 星期二 17:34:14 CST] Your cert is in /root/.acme.sh/xsy.
me/xsy.me.cer
[2018年 09月 25日 星期二 17:34:14 CST] Your cert key is in /root/.acme.sh/
xsy.me/xsy.me.key
[2018年 09月 25日 星期二 17:34:14 CST] The intermediate CA cert is in /roo
t/.acme.sh/xsy.me/ca.cer
[2018年 09月 25日 星期二 17:34:14 CST] And the full chain certs is there:
/root/.acme.sh/xsy.me/fullchain.cer
```

上面生成的文件中：

/root/.acme.sh/xsy.me/fullchain.cer
/root/.acme.sh/xsy.me/xsy.me.key
就是我们需要的证书和key了。

2.4.部署安装证书

```
[root@txvps:~]# mkdir /usr/local/nginx/conf/ssl/xsy.me/ -p
[root@txvps:~]# acme.sh --installcert -d xsy.me \
> --key-file /usr/local/nginx/conf/ssl/xsy.me/xsy.me.key \
> --fullchain-file /usr/local/nginx/conf/ssl/xsy.me/fullchain.cer \
> --reloadcmd "/etc/init.d/nginx reload"

[2018年 09月 25日 星期二 17:35:14 CST] Installing key to:/usr/local/nginx/
conf/ssl/xsy.me/xsy.me.key
[2018年 09月 25日 星期二 17:35:14 CST] Installing full chain to:/usr/local
/nginx/conf/ssl/xsy.me/fullchain.cer
[2018年 09月 25日 星期二 17:35:14 CST] Run reload cmd: /etc/init.d/nginx r
eload
Reloading nginx configuration (via systemctl): [ 确定 ]
[2018年 09月 25日 星期二 17:35:14 CST] Reload success
[root@txvps:~]#
```

安装证书后会记录安装信息，后续自动续期了证书，会自动更新安装目录的证书。

2.5.补充说明

```
# tree .acme.sh/ -d
.acme.sh/
|-- xsy.me
|-- ca
|  `-- acme-v02.api.letsencrypt.org
|-- deploy
`-- dnsapi
```

- .acme.sh 是安装脚本后生成的目录。
- dnsapi 保存各种 dns 的 api 操作脚本
- xsy.me 是生成 xsy.me 证书后，生成的目录
    - xsy.me 目录存放了：

```
 tree xsy.me/
xsy.me/
|-- xsy.me.cer
```

```
|-- xsy.me.conf
|-- xsy.me.csr
|-- xsy.me.csr.conf
|-- xsy.me.key
|-- ca.cer
`-- fullchain.cer
0 directories, 7 files
```

其中 xsy.me.conf 保存了该域名的到期信息，定时任务自动更新证书，会用到它。

3.我博客nginx配置

```
[root@txvps ~]# cat /usr/local/nginx/conf/vhost/www.xsy.me.conf
server {
        listen 443;
        server_name xsy.me www.xsy.me;
        ssl on;

        index index.html index.htm;
        root /home/hexo/www.xsy.me/public/;

        ssl_certificate /root/.acme.sh/xsy.me/fullchain.cer;
        ssl_certificate_key /root/.acme.sh/xsy.me/xsy.me.key;
        ssl_session_timeout 5m;
        ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
        ssl_ciphers ECDHE-RSA-AES128-GCM-SHA256:HIGH:!aNULL:!MD5:!RC4:!D
HE;
        ssl_prefer_server_ciphers on;

        location ~ .*\.(gif|jpg|jpeg|png|bmp|swf)$
        {
            expires      30d;
        }
        location ~ .*\.(js|css)?$
        {
            expires      12h;
        }
        location ~ /.well-known {
            allow all;
        }
        location ~ /\.
        {
            deny all;
        }
```

```
        access_log off;
    }
server
    {
        listen 80;
        server_name xsy.me www.xsy.me;
        index index.html index.htm;

        # 如果用的是单域名证书部署，以下4行去掉注释。这个路径是更新证书时候会访问的，
需要在重定向到https之前，先匹配了。我用的通配符，所以注释了。
        #location ^~ /.well-known/acme-challenge/ {
        #        alias         /home/hexo/www.xsy.me/public/challenges/;
        #        try_files     $uri =404;
        #}

        # 重定向到https
        location / {
                rewrite ^/(.*)$ https://www.xsy.me/$1 permanent;
            }
            access_log off;
        }
```