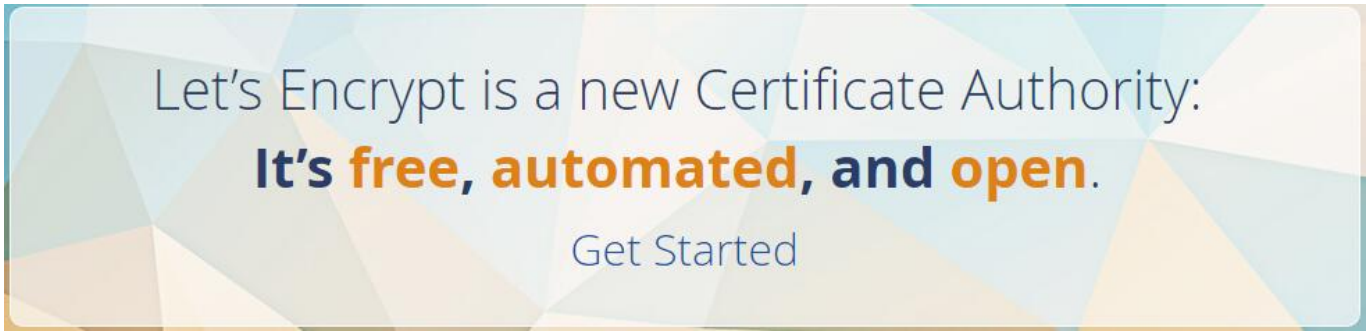
如何申请免费好用的HTTPS证书Let's Encrypt

Let's Encrypt是一款免费、自动化、开放的证书签发服务(Let's Encrypt is a new Certificate Authority: It's free, automated, and open)，它由非营利的网际网路安全研究组织ISRG（Internet Security Research Group，互联网安全研究小组）提供营运服务，并得到EFF、Mozilla、Cisco、Akamai、IdenTrust与密西根大学研究人员的支持，发展十分迅猛。Let's Encrypt目的在于推动全球所有的网站都使用HTTPS加密传输。Let's Encrypt已于2015年12月3日进入公测（Public Beta），所有网站都可以免费获取Let's Encrypt的证书。



如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

本文将介绍如何为个人的网站申请Let's Encrypt的HTTPS证书。在申请证书的时候，本博客使用到的是acme-tiny工具，可以参见https://github.com/diafygi/acme-tiny。当然你也可以使用Let's Encrypt官方提供的certbot-auto工具来申请证书。

创建Let's Encrypt账号

先在服务器的创建一个文件夹，用于存放这里生产的所有文件，本博客使用到的目录是/data/web/ssl/，我们进入这个文件夹，然后使用下面命令创建一个RSA 私钥用于 Let's Encrypt 识别你的身份：

```
iteblog$ cd /data/web/ssl/
iteblog$ openssl genrsa 4096 > account.key
```

acme-tiny工具还可以使用之前生成的Let's Encrypt key来创建RSA 私钥，这里我就不介绍了。

为你的域名创建CSR 文件

Let's Encrypt的证书签发过程使用的就是 ACME
协议，而ACME协议要求用户提交一个CSR文件给它，创建域名RSA 私钥和上面创建账号的RSA
私钥用到的命令一样，如下：

```
iteblog$ openssl genrsa 4096 > domain.key
```

但是我们需要注意的是，一定不要使用上面创建好的account.key来当做domain.key！

有了RSA 私钥之后，我们就可以创建 CSR
文件了，一般域名都有www和不带www的，我们在创建 CSR 文件的时候全部写进去：

```
iteblog$ openssl req -new -sha256 -key domain.key -subj "/" -reqexts SAN -config <(cat /etc/ssl
/openssl.cnf <(printf "[SAN]₩nsubjectAltName=DNS:iteblog.com,DNS:www.iteblog.com")) > do
main.csr
```

把上面的www.iteblog.com和iteblog.com替换成你的域名即可。如果提示找不到
/etc/ssl/openssl.cnf 文件，那么看看/usr/local/openssl/ssl/openssl.cnf文件是否存在；如果还是
不存在，那你运行find / -name
"openssl.
cnf"命令看是否可
以找到；如果还是找不到，那你自己
Google（Google翻墙用《Tunnello：免费的浏览器翻墙插件》）找下原因的。

配置验证服务

在为域名签发证书的时候Let's Encrypt需要确定你对这个域名拥有所有权， Let's
Encrypt的方法是在你的服务器上生成一个随机验证文件，再通过创建 CSR
时指定的域名访问，如果可以访问则表明你对这个域名有控制权。我们可以如下操作：

1、选中一个目录用于创建存放验证文件的目录，本博客使用的是/date/web/iteblog/www/challe
nges/，当然，这个目录你可以根据你自己实际情况设置：

```
iteblog$ mkdir -p /date/web/iteblog/www/challenges/
```

2、在你的Nginx配置文件里面加入以下内容：

```
server {
    listen 80;
    server_name iteblog.com www.iteblog.com;

    location /.well-known/acme-challenge/ {
        alias /date/web/iteblog/www/challenges/;
        try_files $uri =404;
    }

    ...the rest of your config
}
```

设置完上面的配置之后需要重启nginx，那么如何验证上面的设置是否起作用？我们只要在 /date/web/iteblog/www/challenges目录下随便创建一个文件比如iteblog.txt，然后我们通过/.well-known/acme-challenge/iteblog.txt URL访问那个文件，如果能够访问的话说明你配置正确；否则说明你配置错误！这步一定要配置正确，否则下面步骤将无法进行下去。

获取网站证书

上面步骤配置好之后，现在我们使用acme_tiny.py来获取网站证书，我们先下载acme_tiny.py脚本到/data/web/ssl/目录：

```
iteblog$ wget https://raw.githubusercontent.com/diafygi/acme-tiny/master/acme_tiny.py
```

然后执行以下命令获取网站证书：

```
iteblog$ python acme_tiny.py --account-key ./account.key --csr ./domain.csr --acme-
dir /date/web/iteblog/www/challenges/ > ./signed.crt
```

如果一切正常，当前目录下就会生成一个 signed.crt，这就是申请好的证书文件。但是按照上面步骤运行不出意外的话我们肯定会遇到以下的异常：

```
iteblog$ python acme_tiny.py --account-key ./account.key --csr ./domain.csr --acme-
dir /date/web/iteblog/www/challenges/ > ./signed.crt
Parsing account key...
Parsing CSR...
```

```
Registering account...
Already registered!
Verifying www.iteblog.com...
Traceback (most recent call last):
  File "acme_tiny.py", line 198, in <module>
    main(sys.argv[1:])
  File "acme_tiny.py", line 194, in main
    signed_crt = get_crt(args.account_key, args.csr, args.acme_dir, log=LOGGER, CA=args.ca)
  File "acme_tiny.py", line 123, in get_crt
    wellknown_path, wellknown_url))
ValueError: Wrote file to /date/web/iteblogssl/www/challenges/LQJxTnLALp0ef3kB3-63tT6J0QE
qNrKa35QtXoBBFXU, but couldn't download /.well-known/acme-
challenge/LQJxTnLALp0ef3kB3-63tT6J0QEqNrKa35QtXoBBFXU
```

网上有人说这是因为你的域名DNS服务器是在国内导致的，这个是错误的！因为我把我域名DNS迁移到国外，等DNS服务器全部生效之后，还是遇到上面的异常，后来我Google了一番，找了很多英文资料，后面终于找到了原因！是因为acme_tiny.py脚本的问题导致的，我们可以编辑acme_tiny.py脚本，注释掉下面的全部内容：

```
 try:
    resp = urlopen(wellknown_url)
    resp_data = resp.read().decode('utf8').strip()
    assert resp_data == keyauthorization
except (IOError, AssertionError):
    os.remove(wellknown_path)
    raise ValueError("Wrote file to {0}, but couldn't download {1}".format(
        wellknown_path, wellknown_url))
```

然后保存再次获取网站证书：

```
iteblog$ python acme_tiny.py --account-key ./account.key --csr ./domain.csr --acme-
dir /date/web/iteblogssl/www/challenges/ > ./signed.crt
Parsing account key...
Parsing CSR...
Registering account...
Already registered!
Verifying www.iteblog.com...
www.iteblog.com verified!
Verifying iteblog.com...
iteblog.com verified!
```

```
Signing certificate...
Certificate signed!
```

如果你看到如上的内容，那么恭喜你，你的网站证书已经成功获取了！这时候我们的/data/web/ssl/目录下应该有如下几个文件了

```
[iteblog@iteblog.com ssl] $ ll
total 28
-rw-r--r-- 1 iteblog iteblog 3243 Aug  5 09:21 account.key
-rw-r--r-- 1 iteblog iteblog 9159 Aug  5 09:33 acme_tiny.py
-rw-r--r-- 1 iteblog iteblog 1635 Aug  5 09:22 domain.csr
-rw-r--r-- 1 iteblog iteblog 3243 Aug  5 09:21 domain.key
-rw-r--r-- 1 iteblog iteblog 2163 Aug  5 09:34 signed.crt
```

到目前为止我们已经成功地申请到Let's Encrypt的HTTPS证书了，对于Nginx用户，我们还需要把Let's Encrypt的中间证书加入到刚刚生成的signed.crt文件中，具体操作如下：

```
iteblog$ wget -O - https://letsencrypt.org/certs/lets-encrypt-x3-cross-signed.pem > intermediate.pem
iteblog$ cat signed.crt intermediate.pem > chained.pem
```

上面申请完证书之后，我们到《在Nginx中使用Let's Encrypt免费证书配置HTTPS》文章里面查看如何在nginx配置HTTPS。