Let’sencrypt

standolone

```
# service nginx stop
# certbot certonly --dry-run -d tst-amo.net.ua -d www.tst-amo.net.ua -d mail.tst-amo.net.ua -d cloud.tst-amo.net.ua
```

Программа должна отчитаться об успешной работе:

```
IMPORTANT NOTES:
- The dry run was successful.
```

Запускаем вчистовую:

```
# certbot certonly -d tst-amo.net.ua -d www.tst-amo.net.ua -d mail.tst-amo.net.ua -d cloud.tst-amo.net.ua
```

После запускаем и проверяем:

```
# service nginx start
# service nginx status
```

webroot

Этот вариант не предполагает остановку сервиса nginx.

Чтобы не писать каждый раз длинную строку из опций, а еще лучше – не вспоминать о них, запишем основные настройки в файл конфигурации, который certbot ожидает найти в /etc/letsencrypt/cli.ini:

```
authenticator = webroot
webroot-path = /home/www/
post-hook = service nginx reload
text = True
```

Последняя директива нужна чтобы избавить нас от прелестей и красивостей ncurses.

Также нам нужно мягко перезагрузить nginx (без перерыва в

обслуживании) при успешном обновлении сертификатов. Ничего не мешает в этот же момент перезапускать и другие сервисы вроде Postfix, использующие полученные сертификаты. *Команды указываются через точку с запятой*.

Ожидается что certbot будет создавать необходимые для проверки прав на домен файлы в подкаталогах ниже по иерархии к указанному. Вроде таких:

/home/www/.well-known/acme-challenge/example.html

Эти файлы должны будут быть доступны из сети на целевом домене по крайней мере по HTTP:

http://tst-amo.net.ua/.well-known/acme-challenge/example.html

Для следующих проверок создадим какой-то такой файл:

```
mkdir -p /home/www/.well-known/acme-challenge
chown -R nginx:www-data /home/www/.well-known/
echo Success > /home/www/.well-known/acme-challenge/example.html
```

Регистрацию нужно сделать только один раз:

```
certbot register --email postmaster@tst-amo.net.ua
```

В общем случае для получения сертификата необходимо во всех блоках server добавить следующий блок до других блоков location:

```
location /.well-known {
    root /home/www/;
}
```

Понятно, что вписывать для каждого сайта такой блок явно — это моветон, потому создадим файл /etc/nginx/acme.conf с содержанием блока выше.

```
# cat /etc/nginx/acme.conf
location /.well-known {
    root /home/www/;
}
```

Затем для каждого домена и поддомена, для которых нужно получить сертификаты, в блоке server перед всеми блоками location укажем:

```
include acme.conf;
```

Хосты-редиректоры (например, с голого домена на www) можно пропустить. ACME сервер обязан учитывать стандартную переадресацию.

Перезагрузим nginx и проверим что наш тестовый файл виден:

```
# service nginx reload
# curl -L http://tst-amo.net.ua/.well-known/acme-challenge/example.html
Success
```

После проверки лучше удалить тестовый файл – certbot любит удалять за собой всё лишнее, а такой файл будет мешать и вызывать сообщение об ошибке (Unable to clean up challenge directory).

```
rm /home/www/.well-known/acme-challenge/example.html
```

Получаем сертификаты.

Если нужно добавить поддомен или домен в сертификат

Если вы вдруг забыли указать поддомен www, или вам нужно добавить другой домен или поддомен в сертификат (которых может быть до 100 в одном сертификате), то это легко сделать после получения сертификата. Просто запустите команду еще раз, добавив требуемое имя:

```
# certbot certonly -d tst-amo.net.ua -d www.tst-amo.net.ua -d mail.tst-amo.net.ua -d cloud.tst-amo.net.ua
```

Вам будет безальтернативно предложено добавить этот домен в сертификат. Если хочется избежать вопросов, то можно сразу указать одобряющий такое поведение ключ:

```
# certbot certonly --expand -d example.com -d www.example.com
-d shop.example.com
```

Продление

Для автоматизации продления сертификатов добавим в crontab от root одну лишь строчку (sudo crontab -e):

```
33 */12 * * * certbot renew --quiet --allow-subset-of-names
```

Согласно рекомендаций Let’s Encrypt следует пытаться обновить сертификаты два раза в день. Делать это нужно в случайным образом выбранную минуту того часа, а значит вам нужно заменить 33 в этой строке на другое число в диапазоне между 0 и 59. Либо вы можете поступить так как это делается в /etc/cron.d/certbot.

В этой команде ключ *–allow-subset-of-names* нужен чтобы Certbot пытался получить сертификаты для частичного набора доменов.

https://habr.com/post/318952/
https://www.digitalocean.com/community/questions/letsencrypt-failed-authorization-process
https://certbot.eff.org/lets-encrypt/ubuntuxenial-nginx.html