How to use Let's Encrypt

W. Adam Koszek

Koszek ORG

wojciech@koszek.com

2022-10-05T17:47:21Z

Intro. I start from easy SSL introduction. Then explain LetsEncrypt/ACME shortly. Then show you how to get a certificate.

SSL/TLS introduction

Certificates are an underlying element of the SSL/TLS technology. They are an ID of the computers in the Internet, and because of it, people can understand who they connect to. They provide you encryption, and all the popular protocols, because those can be tunneled in the SSL stream.

LetsEncrypt is project and an organization behind free SSL certificates. It's a baby of Electronic Frontier Foundation and Mozilla Foundation. Its goal is popularizing encryption and safe exchange of information. Before you'd have to pay to get this electronic ID–SSL certificate. With LetsEncrypt it's free.

Here you learn how LetsEncrypt works, how to generate a secure SSL certificate and how to renew and maintain it. I use `dehydrated`, which is a simple LetsEncrypt client. It doesn't require any external dependencies, which is important for the server. After reading this article, you'll be able to make yourself an SSL certificate for your service.

How it works

Assume you're John Smith and you want to get a certificate for "domain.com", install it on your server, and have your website users see "green lock" next to the address bar.

Normally there are adhoc ways to prove you own "domain.com". An example would be: if you got "domain.com" from GoDaddy, it means you have an account there, maybe account manager called you over the phone. And they e-mailed you, and maybe you replied. In general, they know you own a domain, so they can also sell you an SSL certificate stating: “Yes, GoDaddy confirms user John Smith owns domain.com - we verified it”.

LetsEncrypt crafted a special protocol for this. It’s called ACME. If you’ve ever configured Google Analytics, Mailchimp, custom Medium domain, SPF of DKIM you might be familiar with how it work:

You must modify something on your side of the connection to prove you really control the domain. By default you’re asked to add a file to your website. You can also modify your DNS entry to a specific value. These changes let LetsEncrypt bot understand that you’re the administrator of this domain.

Example of a verification session:

- **server**: “Hello domain.com owner; add”iownit.txt” in your website files and put ‘allright’ there”
- **client**: “Here you go, I’ve done what you’ve asked me for. `iownit.txt` is where you wanted”
- **server**: “All right. Let’s try. All right! … I see it. Here’s your certificate”

Similar for DNS. Server will say “Add iownit.domain.com subdomain and point it to 123.123.123.123”, the client will have to conform, and server will poll for this domain.

ACME works in a similar way. It’s a protocol for doing these secret exchanges for you. Right now you more or less know how it works, you must pick a LetsEncrypt client.

LetsEncrypt client: acme.sh

Default LetsEncrypt client from the official project sources installed a lot of Python plugins. Additionally each time I run it, it attempted to update itself. This means more Python plugins getting updated. I found this a little bit out of the control.

`acme.sh` is way better. Written in Bash, it comes as a single file. It can bootstrap itself for you.

How to use the LetsEncrypt client

```
acme.sh --install
```

How to renew LetsEncrypt certificate?

How to use the certificate?

how often to renew how many domains