**Generate Wildcard SSL certificate using Let’s Encrypt/Certbot**

**Step 1: Setup Pre-requisites**

If you already have a droplet or a system then make sure your system have Python 2.7 or 3 and git installed on it. As I am starting on fresh Ubuntu droplet, we have to setup the above pre-requisites.

apt-get update

apt-get install python-minimal

python --version

apt-get install git-core

git --version

**Step 2: Setup Certbot**

After setting up the pre-requisites, now will setup the Certbot via github.

cd /opt

git clone https://github.com/certbot/certbot.git

cd certbot && ./certbot-auto

After googling, I came to know, the error triggered due to improper locale variables. Set the locale variables and re-run.

export LC\_ALL="en\_US.UTF-8"

export LC\_CTYPE="en\_US.UTF-8"

You can also install the Certbot via apt installer.

apt-get install letsencrypt

**Step 3: Generate The Wildcard SSL Certificate**

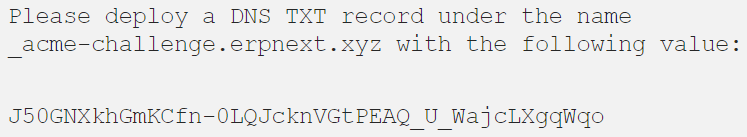
Now with the help of Certbot will generate wildcard certificate for our test domain erpnext.xyz



**Step 4: Authenticate The Domain’s Ownership**

For wildcard certificates, the only challenge method Let’s Encrypt accepts is the DNS challenge, which we can invoke via the preferred-challenges=dns flag.

After executing the above command, the Certbot will share a text record to add to your DNS.



Record Name: \_acme-challenge

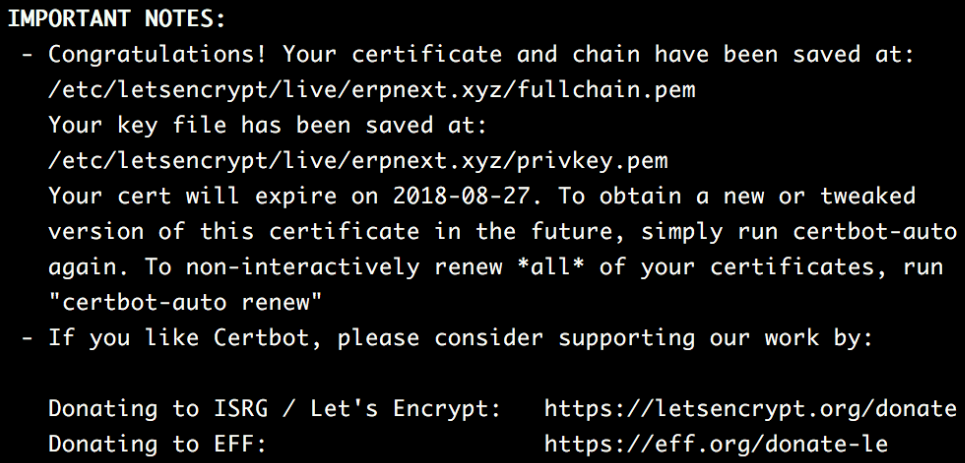
Record Value: J50GNXkhGmKCfn-0LQJcknVGtPEAQ\_U\_WajcLXgqWqo

Create TXT record via DNS console and setup key and value



**Step 5: Get The Certificate**

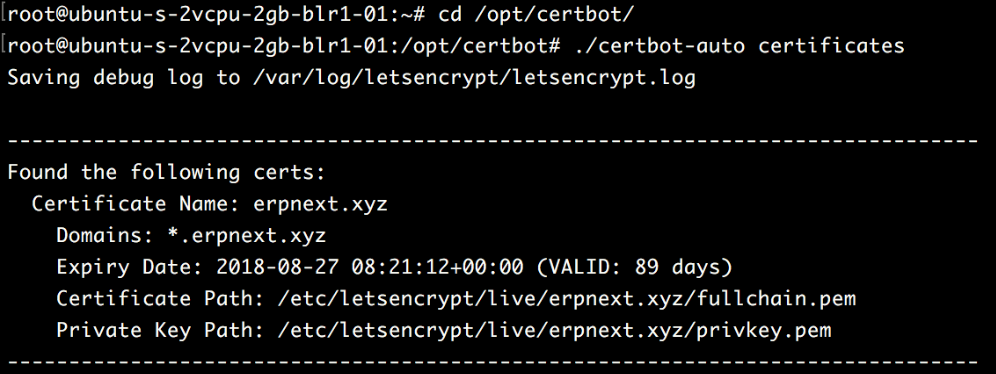
Once you authenticate the domain ownership; by cleaning up dns challenges, Certbot generates the ssl certificate and required keys.



Step 6: Cross Verify The Certificate

To cross verify certificate’s validity via command line run

./certbot-auto certificates



https://medium.com/@saurabh6790/generate-wildcard-ssl-certificate-using-lets-encrypt-certbot-273e432794d7

