Let's Encrypt Domain SSLs!

0 - WHAT YOU NEED FROM THE CLIENT FIRST

-Check to see that the hostname is properly resolving to the server.

-Check to see if they have a different email address they want you to use.

-Check to see if Let's Encrypt is installed. This oneliner will help:

```
if [ -a /root/letsencryptscript.sh ]; then if [ -a /root/installssl.sh ]; then echo
"LetsEncrypt is already installed, but without the new cPanel connecting script for
the hostname. Please perform step 3 of this guide, check /root/letsencryptscript.sh
for the correct lines to be sure that it will work going forwards ( you have to
replace the old cPanel script lines with .pl in them with the new script: /bin/sh
/root/installssl.sh [domain name] ), and then skip to step 5."; else echo "LetsEncrypt
is already installed, please skip to step 5 of this guide."; fi; else echo "Please
install LetsEncrypt from Step 1."; fi
```

1 - Installing Let's Encrypt itself

> **AAAAAAAAAAAAAAAAAa**
> Please note that this will install EPEL and install/upgrade Python to 2.7! Also please note that it does not, as far as I know, work with CloudLinux as of yet (I'll have to research this if anybody running CloudLinux wants this set up).

This only works for CentOS 6/7 (probably cloudlinux too). No CentOS5.

For CentOS 6:

```
rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
rpm -ivh https://rhel6.iuscommunity.org/ius-release.rpm
yum -y install python27 python27-devel python27-pip python27-setuptools
python27-virtualenv --enablerepo=ius
cd /root
git clone https://github.com/letsencrypt/letsencrypt
cd /root/letsencrypt
sed -i "s|--python python2|--python python2.7|" letsencrypt-auto
./letsencrypt-auto --verbose
```

For CentOS 7:

CentOS 7

```
cd /root
git clone https://github.com/letsencrypt/letsencrypt
cd /root/letsencrypt
./letsencrypt-auto --verbose
```

2 - Set up the script to generate a LetsEncrypt SSL automatically in the future

You need to know the email you want to use, and then supply the domain to this one-liner and it will set the other variables:

```
EMAIL="notarealemail@notarealemail.com"; echo -e "Domain?"; read DOMAIN;
CPUSER=`/scripts/whoowns $DOMAIN`; echo "EMAIL=$EMAIL DOMAIN=$DOMAIN CPUSER=$CPUSER";
```

And then you can run this to create the script that we will run to create the certificates. **Please note that this doesn't work for subdomains as of yet, you want to doublecheck the path in that case until this is edited**

/root/letsencryptscript.sh

```
cat > /root/letsencryptscript.sh <<EOF
#simple letsencrypt script to auto-reinstall new certificates.
#Run letsencrypt. This version of python is mentioned in the script itself so we're
using it.
/root/.local/share/letsencrypt/bin/python2.7
/root/.local/share/letsencrypt/bin/letsencrypt --text --agree-tos --email $EMAIL
certonly --webroot --webroot-path /home/$CPUSER/public_html/ --renew-by-default -d
$DOMAIN -d www.$DOMAIN
#Run the cPanel script to install the newly refreshed cert, again with the version of
perl mentioned in the script.
/bin/sh /root/installssl.sh $DOMAIN
#Restart cPanel to make the new certs take effect there.
service cpanel restart
EOF
```

3 - Set up the script to install the certificates into cPanel.

This will download it:

installssl.sh

```
wget files.wiredtree.com/misc/installssl.sh -O /root/installssl.sh
```

4 - Check everything and run the script

Check to make sure this is working - you should be able to manually run the cronjob from the command line, which will create the certificates and install them within cPanel:

```
sh /root/letsencryptscript.sh
```

Assuming that all goes well you can add the cronjob so that this reruns every two months. This one-liner will do it and will show you the current crontab, so you can delete any possible extra lines from it (e.g. from a previous version of this script which used the incorrect value '*/60):

```
if [[ -z `grep "0 0 01 */2 * /bin/sh /root/letsencryptscript.sh" /var/spool/cron/root`
]]; then crontab -l | { cat; echo "0 0 01 */2 * /bin/sh /root/letsencryptscript.sh"; }
| crontab - ; fi ; crontab -l | grep letsencrypt
```

5 - Install Additional Domains

If you already have LetsEncrypt installed, you can add domains to the script by opening the script and copying the current lines, then adding them. Basically, here are all the lines you need:

```
/root/.local/share/letsencrypt/bin/python2.7
/root/.local/share/letsencrypt/bin/letsencrypt --text --agree-tos --email
notanemail@notanemail.com certonly --webroot --webroot-path
/home/somelinu/public_html/ -d somelinux.us -d www.somelinux.us
/bin/sh /root/installssl.sh somelinux.us
```

A short breakdown: The first line creates the cert, and the other line installs it onto the cPanel services. If you want to change this you can edit the lines to change the email, webroot, and domain. Those three things are the only variables that change between sites that you're securing. If you want something to make the line for you:

```
EMAIL="notarealemail@notarealemail.com"; echo -e "Domain?"; read DOMAIN; echo -e
"Email?"; read EMAIL; if [[ -z $EMAIL ]]; then
EMAIL="notarealemail@notarealemail.com"; fi; CPUSER=`/scripts/whoowns $DOMAIN`; echo
"EMAIL=$EMAIL DOMAIN=$DOMAIN CPUSER=$CPUSER";
```

Assuming that shows you the right variables you can then run this:

```
echo "/root/.local/share/letsencrypt/bin/python2.7
/root/.local/share/letsencrypt/bin/letsencrypt --text --agree-tos --email $EMAIL
certonly --webroot --webroot-path /usr/local/apache/htdocs --renew-by-default -d
$DOMAIN -d www.$DOMAIN";
echo "/bin/sh /root/installssl.sh $DOMAIN";
```

And copy and paste those two lines into the /root/letsencryptscript.sh file. Those two lines will recreate the certificate, and then reinstall it within cPanel.

You're done!

You should be fine from here.  If you have any small issues or big issues just let me know and I can edit this to make it clearer or better!