acme.sh - Génération de certificat gratuit

Présentation

"acme.sh" est un outil qui permet de générer des certificats à partir d'autorité gratuite. Il gére les autorités suivantes :

- ZeroSSL.com CA(default)
- Letsencrypt.org CA
- BuyPass.com CA
- SSL.com CA
- Google.com Public CA
- Pebble strict Mode
- Toutes autorités qui est compatible avec la RFC8555

Dépot GIT : https://github.com/acmesh-official/acme.sh

Installation de acme.sh

```
curl https://get.acme.sh | sh -s email=my@example.com
```

Génération d'un certificat

> Remarque : Utiliser de préférence la CA **Let's Encrypt**, par expérience l'API de ZeroSSL est parfois indisponible.

Génération d'un certification en précisant le dossier "webroot"

```
/root/.acme.sh/acme.sh --issue -d example.com -d www.example.com -w /home/wwwroot/example.com
--server letsencrypt
```

Génération d'un certificat en mode server HTTP standalone

```
/root/.acme.sh/acme.sh --issue -d example.com -d www.example.com --standalone --httpport 8443
--server letsencrypt
```

Installation du certificat

Installation du certificat sur HaProxy

```
/root/.acme.sh/acme.sh --install-cert -d example --reloadcmd "cat \$CERT_KEY_PATH
\$CERT_FULLCHAIN_PATH >/etc/pki/mycert.pem &&  systemctl reload haproxy"
```

Installation du certificat sur Apache httpd

```
/root/.acme.sh/acme.sh --install-cert -d example.com --cert-file /etc/ssl/certs/mycert.pem  --
key-file /etc/ssl/private/mykey.key  --fullchain-file /etc/ssl/certs/fullchain.pem --reloadcmd
"systemctl reload apache2"
```

Installation du certificat sur nginx

```
/root/.acme.sh/acme.sh --install-cert -d example.com --key-file
/etc/ssl/private/example.com.key  --fullchain-file /etc/ssl/certs/example.com.pem --reloadcmd
"service nginx force-reload"
```

Configuration des serveurs web pour accepter le challenge "acme"

Configuration du server "apache httpd"

TODO

Configuration du server "haproxy"

```
frontend extranet-ffsa-noprod
   bind 193.41.223.33:443 ssl crt /etc/pki/moncertificat.pem no-sslv3 no-tlsv10 no-tlsv11 no-
tls-tickets ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-
AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-
CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-
SHA384:TLS_AES_128_GCM_SHA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256
   bind 193.41.223.33:80

...
...
...
   #Let's encrypt
   acl begin_acme_challenge  path_beg -i  /.well-known/acme-challenge/
...
...
...
   use_backend acme_standalone_http if begin_acme_challenge
...
```

```
...
...
backend acme_standalone_http
    log global
    mode http
    server srv_letsencrypt 127.0.0.1:8443
```

Configuration du server "nginx"

TODO

Références

Site officiel de acme.sh : https://github.com/acmesh-official/acme.sh

---

Revision #7
Created 2024-01-30 20:24:33 UTC by NoNo
Updated 2024-02-08 16:14:44 UTC by NoNo