Seguí lo que comentaste, un poco por lógica e intuición pero me encuentro un problema.

- Deployé esto cambiando el valor de la entrada '*hosts*' por el dominio para el que quiero generar el certificado.

https://github.com/BySidecar/devops/tree/dev/helm/bysidecar/letsencrypt

- Accedo al clúster por SSH e instalo '*certbot*'

- Ejecuto certbot en el cluster

sudo certbot certonly --nginx –debug

- Obtengo un error que dice lo siguiente:

The key authorization file from the server did not match this challenge: [cadena de texto]

```
admin@ip-172-20-46-232:~$ sudo certbot certonly --nginx --debug
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator nginx, Installer nginx
No names were found in your configuration files. Please enter in your domain
name(s) (comma and/or space separated)  (Enter 'c' to cancel): www.ofertasvirgin.es
Obtaining a new certificate
Performing the following challenges:
http-01 challenge for www.ofertasvirgin.es
Waiting for verification...
Cleaning up challenges
Exiting abnormally:
Traceback (most recent call last):
  File "/usr/bin/certbot", line 11, in <module>
    load_entry_point('certbot==0.28.0', 'console_scripts', 'certbot')()
  File "/usr/lib/python3/dist-packages/certbot/main.py", line 1340, in main
    return config.func(config, plugins)
  File "/usr/lib/python3/dist-packages/certbot/main.py", line 1225, in certonly
    lineage = _get_and_save_cert(le_client, config, domains, certname, lineage)
  File "/usr/lib/python3/dist-packages/certbot/main.py", line 121, in _get_and_save_cert
    lineage = le_client.obtain_and_enroll_certificate(domains, certname)
  File "/usr/lib/python3/dist-packages/certbot/client.py", line 392, in obtain_and_enroll_certificate
    cert, chain, key, _ = self.obtain_certificate(domains)
  File "/usr/lib/python3/dist-packages/certbot/client.py", line 335, in obtain_certificate
    orderr = self._get_order_and_authorizations(csr.data, self.config.allow_subset_of_names)
  File "/usr/lib/python3/dist-packages/certbot/client.py", line 371, in _get_order_and_authorizations
    authzr = self.auth_handler.handle_authorizations(orderr, best_effort)
  File "/usr/lib/python3/dist-packages/certbot/auth_handler.py", line 82, in handle_authorizations
    self._respond(aauthzrs, resp, best_effort)
  File "/usr/lib/python3/dist-packages/certbot/auth_handler.py", line 161, in _respond
    self._poll_challenges(aauthzrs, chall_update, best_effort)
  File "/usr/lib/python3/dist-packages/certbot/auth_handler.py", line 232, in _poll_challenges
    raise errors.FailedChallenges(all_failed_achalls)
certbot.errors.FailedChallenges: Failed authorization procedure. www.ofertasvirgin.es (http-01): urn:ietf
:params:acme:error:unauthorized :: The client lacks sufficient authorization :: The key authorization fil
e from the server did not match this challenge "IMIOUxyteL87_D7nrXqvQeZwc5mixTyqX8_2-KZhpf8.ivcsnCeAJmWb3
QN2cXPWgxs2g5U6orIeoNY59CBVwSY" != "9-noudz5OBr6tFD9k9JLeVSY3mZ61PCUeSlgtJtjV5M.swyaSWd9ezmz1WuCzMoidKLiW
0-shvl9HhvwVc_Q5vA"
Please see the logfiles in /var/log/letsencrypt for more details.

IMPORTANT NOTES:
 - The following errors were reported by the server:

   Domain: www.ofertasvirgin.es
   Type:   unauthorized
   Detail: The key authorization file from the server did not match
   this challenge
   "IMIOUxyteL87_D7nrXqvQeZwc5mixTyqX8_2-KZhpf8.ivcsnCeAJmWb3QN2cXPWgxs2g5U6orIeoNY59CBVwSY"
   !=
   "9-noudz5OBr6tFD9k9JLeVSY3mZ61PCUeSlgtJtjV5M.swyaSWd9ezmz1WuCzMoidKLiW0-shvl9HhvwVc_Q5vA"

   To fix these errors, please make sure that your domain name was
   entered correctly and the DNS A/AAAA record(s) for that domain
   contain(s) the right IP address.
```

- Entiendo que esta cadena tiene que coincidir con el valor definido en el fichero ‘values.yaml’ [cuador verde 1].

Mi pregunta es, **¿cómo sabes este valor a priori? Es decir, es un valor que genera certbot y cambia en cada ejecución**. Probé a cambiar la ruta del path con:

./.well-known/acme-challenge/

Para que acepte cualquier request que llegue con este path, pero no obtuve resultados.

```
ingress:
  enabled: true
  annotations: {
    ingress.kubernetes.io/force-ssl-redirect: "false"
  }
  paths:
  - /.well-known/acme-challenge/klvQmY9DMN3Sn-oJ0FfyFDwTIup7Y-n9gTcGSya_vl8
  hosts:
    - api.bysidecar.com

vhost: |-
  server {
    listen 80;
    server_name _;

    # This entry is intended to be used to renew the api.bysidecar.com certificate
    location / {
      return 200 'klvQmY9DMN3Sn-oJ0FfyFDwTIup7Y-n9gTcGSya_vl8.swyaSWd9ezmz1WuCzMoidKLiW0-shvl9HhvwVc_Q5vA';
    }
  }
```

La misma pregunta se daría en el caso de querer renovar un certificado, **¿cómo se obtiene la cadena que acomapaña al ‘return 200’ ?** [cuadro verde 2]