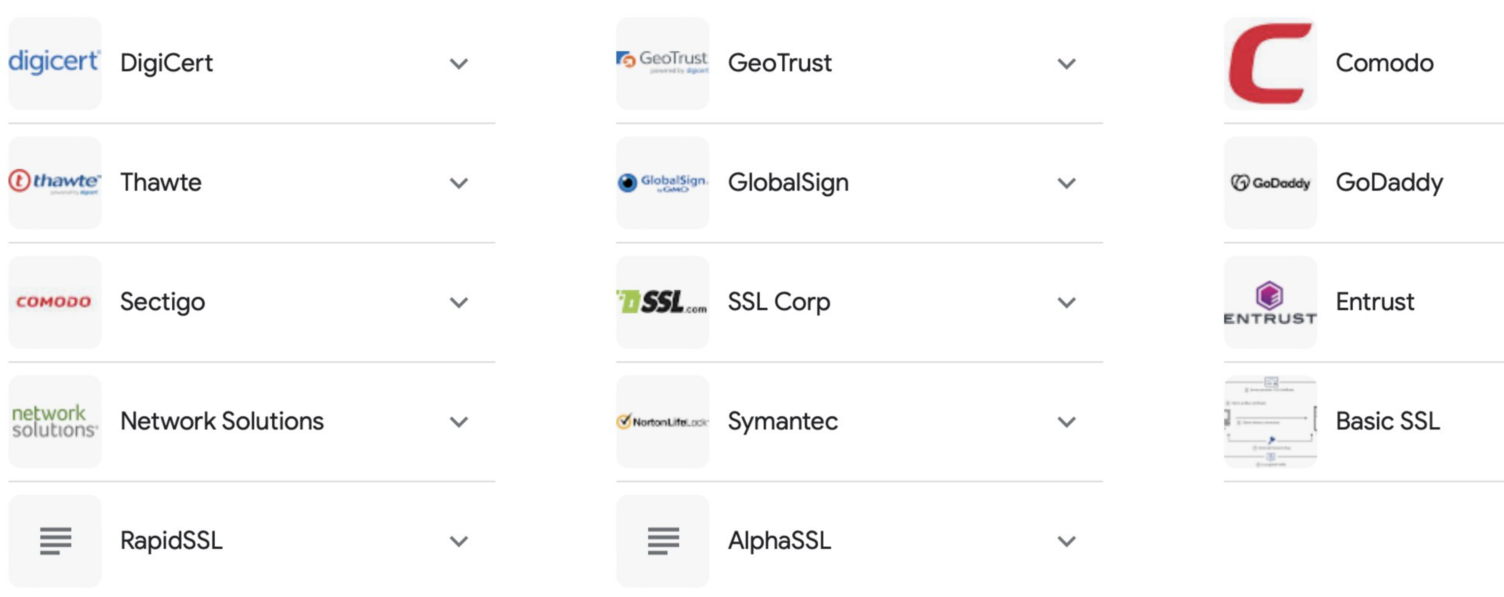
K8s Cert Manager

- SSL/TLS
  - https
- LetsEncrypt
- Installation
- Cluster Issuer
- Ingress bind

K8s Cert Manager

- SSL/TLS
  - https
    - CA



DigiCert

Thawte

Sectigo

Network Solutions

RapidSSL

GeoTrust

GlobalSign

SSL Corp

Symantec

AlphaSSL

Comodo

GoDaddy

Entrust

Basic SSL

C

K8s Cert Manager

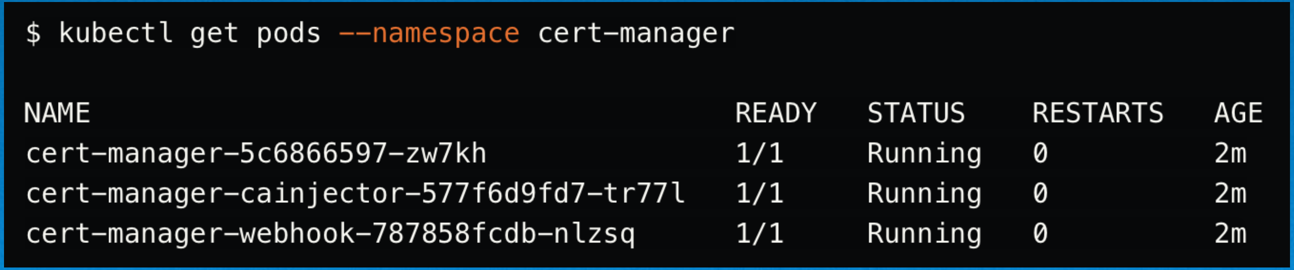
- LetsEncrypt
  - Completely Free
  - need a FQDN
  - http challenge
- Certbot
  - linux tool set
    - get certs
      - for 90 days
    - renew

K8s Cert Manager

- Install steps for k8s

  1. Install Cert-manager onto your cluster

```
kubectl apply -f https://github.com/cert-manager/cert-manager/releases/download/v1.1.1/cert-manager.yaml
```



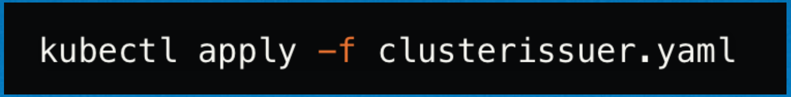
```
$ kubectl get pods --namespace cert-manager

NAME                                       READY   STATUS    RESTARTS   AGE
cert-manager-5c6866597-zw7kh               1/1     Running   0          2m
cert-manager-cainjector-577f6d9fd7-tr77l   1/1     Running   0          2m
cert-manager-webhook-787858fcdb-nlzsq      1/1     Running   0          2m
```

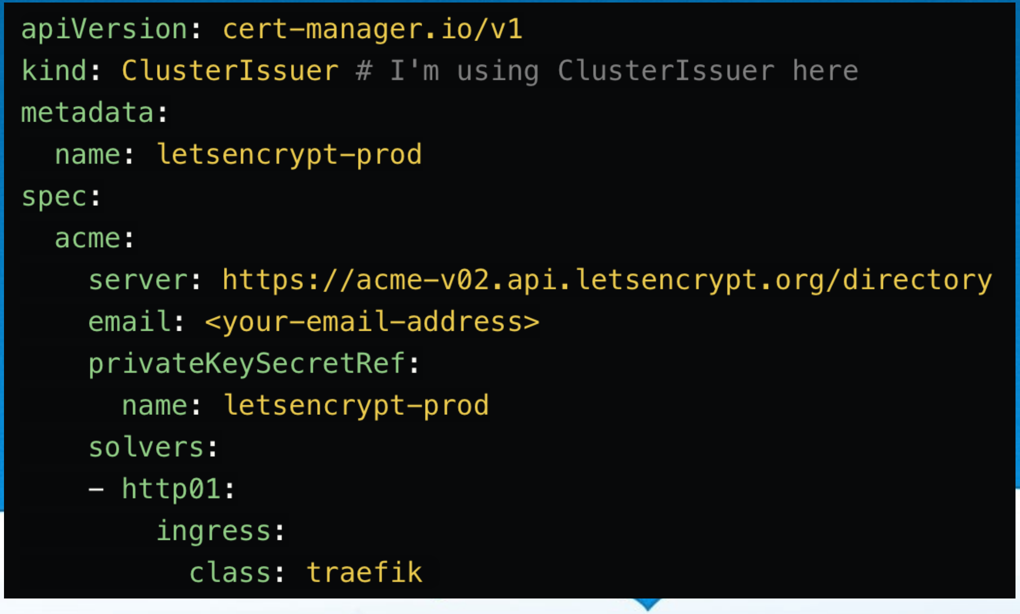
K8s Cert Manager

- Install steps for k8s

  2. Add LetsEncrypt as an Issuer (or ClusterIssuer)



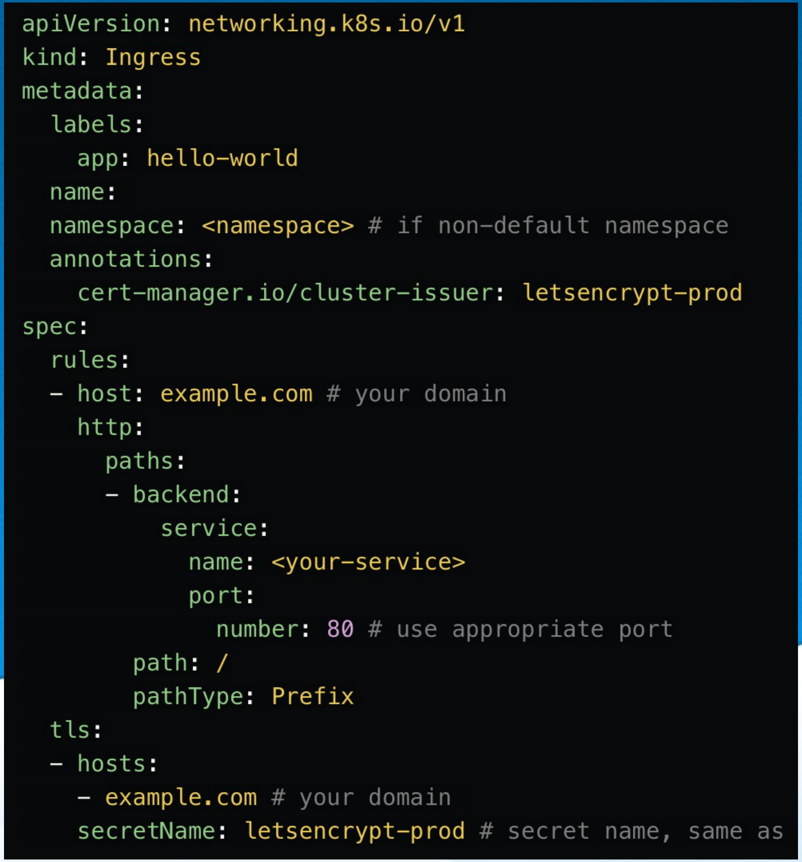
```
kubectl apply -f clusterissuer.yaml
```



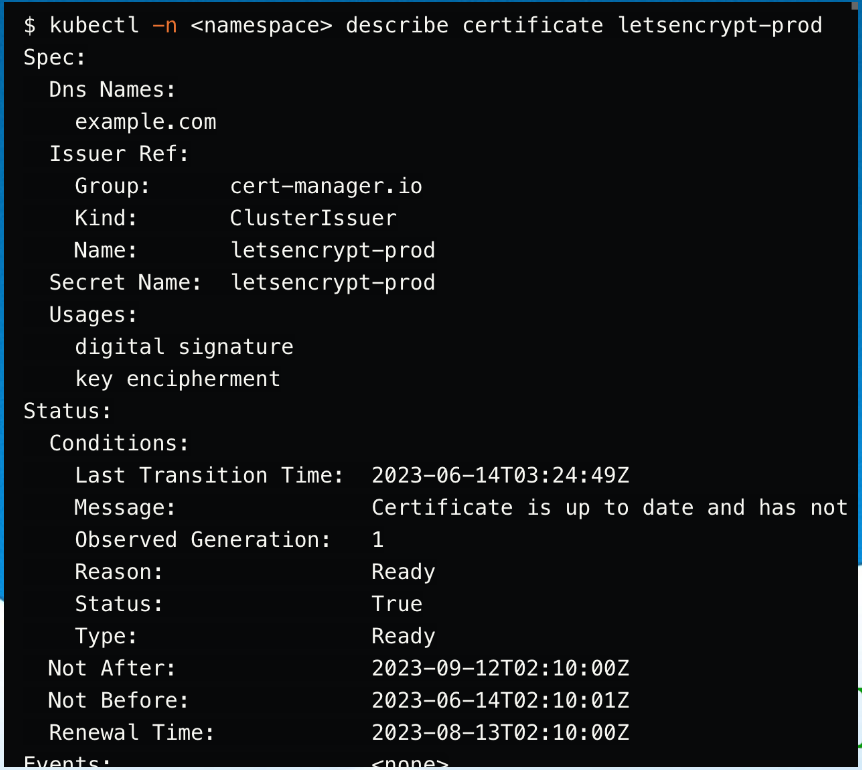
```yaml
apiVersion: cert-manager.io/v1
kind: ClusterIssuer # I'm using ClusterIssuer here
metadata:
  name: letsencrypt-prod
spec:
  acme:
    server: https://acme-v02.api.letsencrypt.org/directory
    email: <your-email-address>
    privateKeySecretRef:
      name: letsencrypt-prod
    solvers:
    - http01:
        ingress:
          class: traefik
```

K8s Cert Manager

3. Update ingress to use certificate



```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  labels:
    app: hello-world
  name:
  namespace: <namespace> # if non-default namespace
  annotations:
    cert-manager.io/cluster-issuer: letsencrypt-prod
spec:
  rules:
  - host: example.com # your domain
    http:
      paths:
      - backend:
          service:
            name: <your-service>
            port:
              number: 80 # use appropriate port
        path: /
        pathType: Prefix
  tls:
  - hosts:
    - example.com # your domain
    secretName: letsencrypt-prod # secret name, same as
```



```
$ kubectl -n <namespace> describe certificate letsencrypt-prod
Spec:
  Dns Names:
    example.com
  Issuer Ref:
    Group:      cert-manager.io
    Kind:       ClusterIssuer
    Name:       letsencrypt-prod
  Secret Name:  letsencrypt-prod
  Usages:
    digital signature
    key encipherment
Status:
  Conditions:
    Last Transition Time:  2023-06-14T03:24:49Z
    Message:               Certificate is up to date and has not
    Observed Generation:   1
    Reason:                Ready
    Status:                True
    Type:                  Ready
  Not After:               2023-09-12T02:10:00Z
  Not Before:              2023-06-14T02:10:01Z
  Renewal Time:            2023-08-13T02:10:00Z
Events:                    <none>
```