



Learn Git and GitHub without any code!

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

Read the guide

bluebossa63 / tkg-extensions Private

- Code
- Issues
- Pull requests
- Actions
- Projects
- Security
- Insights
- Settings

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

Comparing changes

base: main ← compare: cert-manager-with-ca

✓ **Able to merge.** These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

- 2 commits
- 25 files changed
- 0 comments
- 1 contributor

Commits on Jan 26, 2021

ignore

1ba9a2a



change to clusterissuer

Showing 25 changed files with 584 additions and 240 deletions.

Unified

Split

```

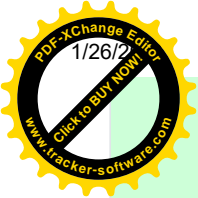
 7 .gitignore
@@ -0,0 +1,7 @@
1 + history-07.txt
2 + history-06.txt
3 + history-05.txt
4 + history-04.txt
5 + history-03.txt
6 + registry/harbor/history-02.txt
7 + ingress/examples/https-ingress/check-history.txt

```

```

57 ingress/contour/02-certs-ca-issued.yaml
@@ -0,0 +1,57 @@
1 + apiVersion: cert-manager.io/v1alpha2
2 + kind: Certificate
3 + metadata:
4 +   name: contour-cert
5 +   namespace: tanzu-system-ingress
6 + spec:
7 +   secretName: contourcert
8 +   duration: 8760h
9 +   renewBefore: 360h
10 +   organization:
11 +     - Project Contour
12 +   commonName: contour
13 +   isCA: false
14 +   keySize: 2048
15 +   keyAlgorithm: rsa
16 +   keyEncoding: pkcs1
17 +   usages:
18 +     - server auth
19 +   dnsNames:
20 +     - contour
21 +     - contour.tanzu-system-ingress
22 +     - contour.tanzu-system-ingress.svc
23 +     - contour.tanzu-system-ingress.svc.cluster.local
24 +   ipAddresses: []
25 +   issuerRef:
26 +     name: niceeasy-ca
27 +     kind: ClusterIssuer
28 +     group: cert-manager.io
29 + ---

```



```

30 + apiVersion: cert-manager.io/v1alpha2
31 + kind: Certificate
32 + metadata:
33 +   name: envoy-cert
34 +   namespace: tanzu-system-ingress
35 + spec:
36 +   secretName: envoycert
37 +   duration: 8760h
38 +   renewBefore: 360h
39 +   organization:
40 +     - Project Contour
41 +   commonName: envoy
42 +   isCA: false
43 +   keySize: 2048
44 +   keyAlgorithm: rsa
45 +   keyEncoding: pkcs1
46 +   usages:
47 +     - client auth
48 +   dnsNames:
49 +     - envoy
50 +     - envoy.tanzu-system-ingress
51 +     - envoy.tanzu-system-ingress.svc
52 +     - envoy.tanzu-system-ingress.svc.cluster.local
53 +   ipAddresses: []
54 +   issuerRef:
55 +     name: niceeasy-ca
56 +     kind: ClusterIssuer
57 +     group: cert-manager.io

```

30 ingress/contour/02-certs-selfsigned.yaml

Load diff

This file was deleted.

2 ingress/contour/overlays/change-namespace.yaml

38	38	subjects:
39	39	#@overlay/match by=kind.serviceaccount
40	40	- kind: ServiceAccount
41		- namespace: #@ values.contour.namespace
	41	+ namespace: #@ values.contour.namespace

1 ingress/examples/common/02-deployments.yaml



```

17     app: hello
18     tier: web
19   spec:
20 +   serviceAccountName: sample
21   containers:
22     - name: hello-app
23       image: gcr.io/google-samples/hello-app:1.0

```

26 ■■■■■ ingress/examples/https-ingress/https-certificate.yaml

```

...   ...   @@ -0,0 +1,26 @@
1 + apiVersion: cert-manager.io/v1alpha2
2 + kind: Certificate
3 + metadata:
4 +   name: contour-cert
5 +   namespace: test-ingress
6 + spec:
7 +   secretName: https-secret
8 +   duration: 8760h
9 +   renewBefore: 360h
10 +   organization:
11 +   - Project Contour
12 +   commonName: samples
13 +   isCA: false
14 +   keySize: 2048
15 +   keyAlgorithm: rsa
16 +   keyEncoding: pkcs1
17 +   usages:
18 +   - server auth
19 +   dnsNames:
20 +   - foo.bar.com
21 +   - foo.ne.local
22 +   ipAddresses: []
23 +   issuerRef:
24 +     name: niceneasy-ca
25 +     kind: ClusterIssuer
26 +     group: cert-manager.io 

```

9 ■■■■■ ingress/examples/https-ingress/https-secret.yaml

[Load diff](#)

This file was deleted.



27 ■■■■■ monitoring/grafana/certificates.yaml

```
...  ...  @@ -0,0 +1,27 @@
1  + apiVersion: cert-manager.io/v1alpha2
2  + kind: Certificate
3  + metadata:
4  +   name: grafana-tls-cert
5  +   namespace: tanzu-system-monitoring
6  + spec:
7  +   secretName: grafana-tls
8  +   duration: 87600h
9  +   renewBefore: 360h
10 +   organization:
11 +   - Project Grafana
12 +   commonName: grafana
13 +   isCA: false
14 +   keySize: 2048
15 +   keyAlgorithm: rsa
16 +   keyEncoding: pkcs1
17 +   usages:
18 +   - server auth
19 +   - client auth
20 +   dnsNames:
21 +   - grafana.tanzu.ne.local
22 +   - notary.grafana.tanzu.ne.local
23 +   ipAddresses: []
24 +   issuerRef:
25 +     name: niceneasy-ca
26 +     kind: ClusterIssuer
27 +     group: cert-manager.io
```

52 ■■■■■ monitoring/grafana/ingress-secret.yaml

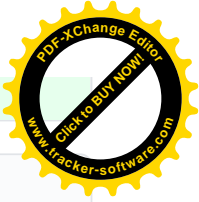
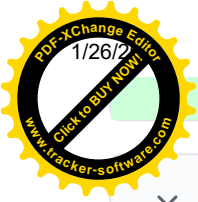
[Load diff](#)
This file was deleted.

54 ■■■■■ monitoring/prometheus/certificates.yaml

```
...  ...  @@ -0,0 +1,54 @@
1  + apiVersion: cert-manager.io/v1alpha2
2  + kind: Certificate
3  + metadata:
4  +   name: prometheus-ca
```



```
5 + namespace: tanzu-system-monitoring
6 + spec:
7 +   secretName: prometheus-ca-key-pair
8 +   duration: 87600h
9 +   renewBefore: 360h
10 +   organization:
11 +     - Project Prometheus
12 +   commonName: Prometheus CA
13 +   isCA: true
14 +   keySize: 2048
15 +   keyAlgorithm: rsa
16 +   keyEncoding: pkcs1
17 +   usages:
18 +     - server auth
19 +     - client auth
20 +   dnsNames:
21 +     - prometheusca
22 +   ipAddresses: []
23 +   issuerRef:
24 +     name: niceneasy-ca
25 +     kind: ClusterIssuer
26 +     group: cert-manager.io
27 + ---
28 + apiVersion: cert-manager.io/v1alpha2
29 + kind: Certificate
30 + metadata:
31 +   name: prometheus-tls-cert
32 +   namespace: tanzu-system-monitoring
33 + spec:
34 +   secretName: prometheus-tls
35 +   duration: 87600h
36 +   renewBefore: 360h
37 +   organization:
38 +     - Project Prometheus
39 +   commonName: prometheus
40 +   isCA: false
41 +   keySize: 2048
42 +   keyAlgorithm: rsa
43 +   keyEncoding: pkcs1
44 +   usages:
45 +     - server auth
46 +     - client auth
47 +   dnsNames:
48 +     - prometheus.tanzu.ne.local
49 +     - notary.prometheus.tanzu.ne.local
50 +   ipAddresses: []
51 +   issuerRef:
52 +     name: niceneasy-ca
53 +     kind: ClusterIssuer
```



54 + group: cert-manager.io

52 monitoring/prometheus/ingress-secret.yaml

Load diff

This file was deleted.

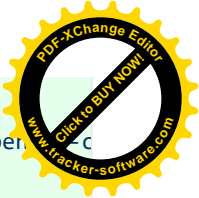
101 push-ca-certs.yaml

@@ -0,0 +1,101 @@

```

1 + apiVersion: v1
2 + data:
3 +   ca.pem: |+
4 +     -----BEGIN CERTIFICATE-----
5 +     MIIEMTCCAxmgAwIBAgIEYATU6DANBgkqhkiG9w0BAQsFADCBPTEqMCgGCSqGSIb3
6 +     DQEJARYbZGFuYWVsZS51bHJpY2hAbm1jZW51YXN5LmNoMQswCQYDVQGEwJDSDEN
7 +     MAsGA1UECAwEQmVybWVjEWMBQGA1UEBwwNTc08aGxldGh1cm51bjETMBEGA1UECgWK
8 +     bm1jZW51YXN5IDEUMBIGA1UECwwLRGV2ZWxvcG11bnQxGDAWBGNVBAMMD2NhLm5p
9 +     Y2VuZWZzeS5jaDAeFw0yMTAxMTgwMDIzMDRaFw0yNDAxMTgwMDIzMDRaMIGlMSow
10 +    KAYJKoZIhvcNAQkBFhtkYW5pZWx1LnVsZm1jaEBuaWw1bmVhc3kuY2gxCzAJBgNV
11 +    BAYTAKNIMQ0wCwYDVQQIDARCRXJlYXN5Y2h0c1u0sETGHUg7LgU17CE6YAXt
12 +    EQYDVQKDApuawN1bmVhc3kgMRQwEgYDVQQLDAtEZXZ1bG9wbWVudDEUMBYGA1UE
13 +    AwwPY2Eubm1jZW51YXN5LmNoMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
14 +    AQEAu6t+hNEwhxyIWyw/PbhYZK4DcsTBIF0jJrAQYK1HDDjiZXVf9G63bFipWyRa
15 +    +wb3KN9pJZ075yiiGr51x/keJNQ5zICEP2Jx2jh0c1u0sETGHUg7LgU17CE6YAXt
16 +    dHNBBEAWGr17XtRWUbae+30HYDXDxbZa0M1jpbE3u/EPVgFRtQe1SyupNJHC6xE
17 +    N19IUrTYQcrjhyRQsEjnl6VAXNg3qhzI1fY5c9s08ur9KM1eA6vniCKbCQP1V2dk
18 +    aqqEng7c1fky9sL0Et8Gn1UFd8ev4IgpTPIRejK2WK9RzM4CB6YORMLy034Wyx8y
19 +    3S2zDyHg2QtGW6qCeQ3DKDy+NwIDAQABo2cwZTAPBgNVHRMECDAGAQH/AgEDMA5G
20 +    A1UdDQEAwIBBjBFBGbnVHSUEPjA8BggrBgEFBQcDAQYIKwYBBQUHAWIGCCsGAQUF
21 +    BwMDBBggrBgEFBQcDBAYIKwYBBQUHAWGCCsGAQUFBwMJMA0GCSqGSIb3DQEBCwUA
22 +    A4IBAQCbmV8QxhLXI9rf2dfLfj8zkxt70u3ms0oxdTMR7cix3Gw57BfALSjcXeOd
23 +    hELNEGhtFUL6xzFDnMCsw3LR8WF0tZtCYN+DFmVpdmAt/JLVvKCsZkGGdeT4GzH
24 +    8zEmBCf4xLrf6uCYwomTY/Zin2wngXlqNpEIDR9jtq6cuy1XhmLaI3gzPTPvabxm
25 +    JjPVQDcgCHEqKNMGICr2JF26izqkKFSC7yusDy/+SYNQRi+YzeoSgP4FP3bS16dc
26 +    GRrCRwdUneiGLmNsYFn8nod12KSxEROftWt0JXVqemQGC9rWz42a/hqzNSu7VamWS
27 +    dJLLRuINSYA1END87glIx6+181ot
28 +     -----END CERTIFICATE-----
29 + kind: ConfigMap
30 + metadata:
31 +   name: trusted-ca-cm
32 +   namespace: cert-manager
33 + ---
34 + apiVersion: v1

```



```
35 + data:
36 +   build-ca.sh: "#!/usr/bin/env bash \nset -euxo pipefail\ntdnf install -y open
37 + kind: ConfigMap
38 + metadata:
39 +   name: rehash-script
40 +   namespace: cert-manager
41 + ---
42 + apiVersion: apps/v1
43 + kind: DaemonSet
44 + metadata:
45 +   name: trusted-ca-updater
46 +   namespace: cert-manager
47 +   labels:
48 +     k8s-app: trusted-ca-updater
49 + spec:
50 +   selector:
51 +     matchLabels:
52 +       name: trusted-ca-updater
53 +   template:
54 +     metadata:
55 +       labels:
56 +         name: trusted-ca-updater
57 +     spec:
58 +       serviceAccountName: cert-manager
59 +       tolerations:
60 +         # this toleration is to have the daemonset runnable on master nodes
61 +         # remove it if your masters can't run pods
62 +         - key: node-role.kubernetes.io/master
63 +           effect: NoSchedule
64 +     initContainers:
65 +       - name: script-runner
66 +         image: photon:3.0
67 +         command: ["/bin/sh", "-c", "/root/build-ca.sh" ]
68 +         volumeMounts:
69 +           - name: update-trusted-certs-script
70 +             mountPath: /root/
71 +           - name: certs-dir
72 +             mountPath: /etc/ssl/certs
73 +           - name: agg-certs-dir
74 +             mountPath: /etc/pki/tls/certs/
75 +         env:
76 +           - name: TRUSTED_CERT
77 +             valueFrom:
78 +               configMapKeyRef:
79 +                 name: trusted-ca-cm
80 +                 key: ca.pem
81 +         resources:
82 +           limits:
83 +             ephemeral-storage: 4G
```

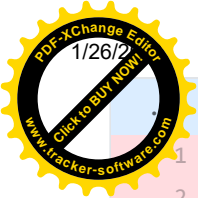



```
84 +     containers:
85 +       - name: sleepy
86 +         image: photon:3.0
87 +         command: ["/bin/sh"]
88 +         args: ["-c", "while true; do sleep 3600;done"]
89 +     volumes:
90 +       - name: update-trusted-certs-script
91 +         configMap:
92 +           name: rehash-script
93 +           defaultMode: 0766
94 +       - name: certs-dir
95 +         hostPath:
96 +           path: /etc/ssl/certs
97 +           type: Directory
98 +       - name: agg-certs-dir
99 +         hostPath:
100 +           path: /etc/pki/tls/certs/
101 +           type: Directory ⊖
```

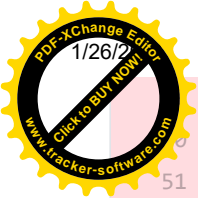
▼ 21 █████ registry/harbor/00-common.yaml 📄

```
5 | 5 |     name: harbor-system
6 | 6 |     labels:
7 | 7 |       app: "harbor"
8 | 8 | + ---
9 | 9 | + apiVersion: v1
10 | 10 | + kind: ServiceAccount
11 | 11 | + metadata:
12 | 12 | +   name: harbor
13 | 13 | +   namespace: "harbor-system"
14 | 14 | +   labels:
15 | 15 | +     app: "harbor"
16 | 16 | + ---
17 | 17 | + apiVersion: rbac.authorization.k8s.io/v1
18 | 18 | + kind: ClusterRoleBinding
19 | 19 | + metadata:
20 | 20 | +   name: harbor
21 | 21 | + roleRef:
22 | 22 | +   apiGroup: rbac.authorization.k8s.io
23 | 23 | +   kind: ClusterRole
24 | 24 | +   name: psp:vmware-system-privileged
25 | 25 | + subjects:
26 | 26 | + - kind: ServiceAccount
27 | 27 | +   name: harbor
28 | 28 | +   namespace: "tanzu-system-registry" ⊖
```

▼ 352 █████ registry/harbor/01-certificate.yaml 📄



```
... @@ -1,89 +1,267 @@
1 - #@ load("/values.star", "values")
2 - #@ load("/helpers.star", "generate_harbor_tls")
3 - #@ load("@ytt:base64", "base64")
4 - #@ load("@ytt:overlay", "overlay")
5 - #@ load("@ytt:template", "template")
6 - #@ load("certificates.lib.yaml", "generate_dns_names", "generate_self_signed_issuer",
7 -
8 - #@ harbor_name = "harbor"
9 - #@ harbor_namespace = values.namespace
10 - #@ harbor_organization = "Project Harbor"
11 - #@ harbor_self_signed_ca_issuer = harbor_name + "-self-signed-ca-issuer"
12 - #@ harbor_ca = harbor_name + "-ca"
13 - #@ harbor_ca_common_name = "Harbor CA"
14 - #@ harbor_ca_dns_name = harbor_name + "ca"
15 - #@ harbor_ca_key_pair = harbor_name + "-ca-key-pair"
16 - #@ harbor_ca_issuer = harbor_name + "-ca-issuer"
17 - #@ harbor_cert = harbor_name + "-cert"
18 - #@ harbor_cert_duration = "87600h"
19 - #@ harbor_cert_renew_before = "360h"
20 -
21 - --- #@ generate_self_signed_issuer(harbor_self_signed_ca_issuer, harbor_namespace)
22 - --- #@ generate_ca_certificate(harbor_ca, harbor_namespace, harbor_cert_duration, har
23 - --- #@ generate_ca_issuer(harbor_ca_issuer, harbor_namespace, harbor_ca_key_pair)
24 -
25 - #@ harbor_clair_internal_tls_cert = "harbor-clair-internal-cert"
26 - #@ harbor_clair_internal_tls_secret = "harbor-clair-internal-tls"
27 - #@ harbor_clair_common_name = "harbor-clair"
28 - --- #@ generate_certificate(harbor_clair_internal_tls_cert, harbor_namespace, harbor_
29 -
30 - #@ harbor_core_internal_tls_cert = "harbor-core-internal-cert"
31 - #@ harbor_core_internal_tls_secret = "harbor-core-internal-tls"
32 - #@ harbor_core_common_name = "harbor-core"
33 - --- #@ generate_certificate(harbor_core_internal_tls_cert, harbor_namespace, harbor_c
34 -
35 - #@ harbor_jobservice_internal_tls_cert = "harbor-jobservice-internal-cert"
36 - #@ harbor_jobservice_internal_tls_secret = "harbor-jobservice-internal-tls"
37 - #@ harbor_jobservice_common_name = "harbor-jobservice"
38 - --- #@ generate_certificate(harbor_jobservice_internal_tls_cert, harbor_namespace, ha
39 -
40 - #@ harbor_portal_internal_tls_cert = "harbor-portal-internal-cert"
41 - #@ harbor_portal_internal_tls_secret = "harbor-portal-internal-tls"
42 - #@ harbor_portal_common_name = "harbor-portal"
43 - --- #@ generate_certificate(harbor_portal_internal_tls_cert, harbor_namespace, harbor
44 -
45 - #@ harbor_registry_internal_tls_cert = "harbor-registry-internal-cert"
46 - #@ harbor_registry_internal_tls_secret = "harbor-registry-internal-tls"
47 - #@ harbor_registry_common_name = "harbor-registry"
48 - --- #@ generate_certificate(harbor_registry_internal_tls_cert, harbor_namespace, harb
```



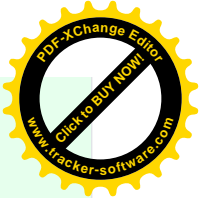
```
51 -
52 - #@ harbor_trivy_internal_tls_cert = "harbor-trivy-internal-cert"
53 - #@ harbor_trivy_internal_tls_secret = "harbor-trivy-internal-tls"
54 - #@ harbor_trivy_common_name = "harbor-trivy"
55 - --- #@ generate_certificate(harbor_trivy_internal_tls_cert, harbor_namespace, harbor_
56 -
57 - #@ harbor_token_service_cert = "harbor-token-service-cert"
58 - #@ harbor_token_service_secret = "harbor-token-service"
59 - #@ harbor_token_service_common_name = "harbor-token-service"
60 - --- #@ generate_certificate(harbor_token_service_cert, harbor_namespace, harbor_cert_
61 -
62 - #@ harbor_notary_signer_cert = "harbor-notary-signer-cert"
63 - #@ harbor_notary_signer_secret = "harbor-notary-signer"
64 - #@ harbor_notary_signer_common_name = "harbor-notary-signer"
65 - --- #@ generate_certificate(harbor_notary_signer_cert, harbor_namespace, harbor_cert_
66 -
67 - #@ harbor_tls_cert = "harbor-tls-cert"
68 - #@ harbor_tls_secret = "harbor-tls"
69 - #@ harbor_tls_common_name = "harbor"
70 - #@ if generate_harbor_tls():
71 - --- #@ generate_certificate(harbor_tls_cert, harbor_namespace, harbor_cert_duration,
72 - #@ end
73 -
74 - #@ if not generate_harbor_tls():
75 - #@ ca_cert = getattr(values.tlsCertificate, "ca.crt")
76 - #@ tls_cert = getattr(values.tlsCertificate, "tls.crt")
77 - #@ tls_key = getattr(values.tlsCertificate, "tls.key")
78
79 + apiVersion: cert-manager.io/v1alpha2
80 + kind: Certificate
81 + metadata:
82 +   name: harbor-clair-internal-cert
83 +   namespace: tanzu-system-registry
84 + spec:
85 +   secretName: harbor-clair-internal-tls
86 +   duration: 87600h
87 +   renewBefore: 360h
88 +   organization:
89 +     - Project Harbor
90 +   commonName: harbor-clair
91 +   isCA: false
92 +   keySize: 2048
93 +   keyAlgorithm: rsa
94 +   keyEncoding: pkcs1
95 +   usages:
96 +     - server auth
97 +     - client auth
98 +   dnsNames:
99 +     - harbor-clair
```



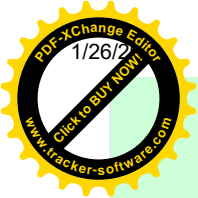
```
22 + - harbor-clair.tanzu-system-registry
23 + - harbor-clair.tanzu-system-registry.svc
24 + - harbor-clair.tanzu-system-registry.svc.cluster.local
25 +   ipAddresses: []
26 +   issuerRef:
27 +     name: niceneasy-ca
28 +     kind: ClusterIssuer
29 +     group: cert-manager.io
30 + ---
31 + apiVersion: cert-manager.io/v1alpha2
32 + kind: Certificate
33 + metadata:
34 +   name: harbor-core-internal-cert
35 +   namespace: tanzu-system-registry
36 + spec:
37 +   secretName: harbor-core-internal-tls
38 +   duration: 87600h
39 +   renewBefore: 360h
40 +   organization:
41 +     - Project Harbor
42 +   commonName: harbor-core
43 +   isCA: false
44 +   keySize: 2048
45 +   keyAlgorithm: rsa
46 +   keyEncoding: pkcs1
47 +   usages:
48 +     - server auth
49 +     - client auth
50 +   dnsNames:
51 +     - harbor-core
52 +     - harbor-core.tanzu-system-registry
53 +     - harbor-core.tanzu-system-registry.svc
54 +     - harbor-core.tanzu-system-registry.svc.cluster.local
55 +   ipAddresses: []
56 +   issuerRef:
57 +     name: niceneasy-ca
58 +     kind: ClusterIssuer
59 +     group: cert-manager.io
60 + ---
61 + apiVersion: cert-manager.io/v1alpha2
62 + kind: Certificate
63 + metadata:
64 +   name: harbor-jobservice-internal-cert
65 +   namespace: tanzu-system-registry
66 + spec:
67 +   secretName: harbor-jobservice-internal-tls
68 +   duration: 87600h
69 +   renewBefore: 360h
70 +   organization:
```



```
71 + - Project Harbor
72 +   commonName: harbor-jobservice
73 +   isCA: false
74 +   keySize: 2048
75 +   keyAlgorithm: rsa
76 +   keyEncoding: pkcs1
77 +   usages:
78 +     - server auth
79 +     - client auth
80 +   dnsNames:
81 +     - harbor-jobservice
82 +     - harbor-jobservice.tanzu-system-registry
83 +     - harbor-jobservice.tanzu-system-registry.svc
84 +     - harbor-jobservice.tanzu-system-registry.svc.cluster.local
85 +   ipAddresses: []
86 +   issuerRef:
87 +     name: niceneasy-ca
88 +     kind: ClusterIssuer
89 +     group: cert-manager.io
90 + ---
91 + apiVersion: cert-manager.io/v1alpha2
92 + kind: Certificate
93 + metadata:
94 +   name: harbor-portal-internal-cert
95 +   namespace: tanzu-system-registry
96 + spec:
97 +   secretName: harbor-portal-internal-tls
98 +   duration: 87600h
99 +   renewBefore: 360h
100 +   organization:
101 +     - Project Harbor
102 +     commonName: harbor-portal
103 +     isCA: false
104 +     keySize: 2048
105 +     keyAlgorithm: rsa
106 +     keyEncoding: pkcs1
107 +     usages:
108 +       - server auth
109 +       - client auth
110 +     dnsNames:
111 +       - harbor-portal
112 +       - harbor-portal.tanzu-system-registry
113 +       - harbor-portal.tanzu-system-registry.svc
114 +       - harbor-portal.tanzu-system-registry.svc.cluster.local
115 +     ipAddresses: []
116 +     issuerRef:
117 +       name: niceneasy-ca
118 +       kind: ClusterIssuer
119 +       group: cert-manager.io
```



```
120 + ---
121 + apiVersion: cert-manager.io/v1alpha2
122 + kind: Certificate
123 + metadata:
124 +   name: harbor-registry-internal-cert
125 +   namespace: tanzu-system-registry
126 + spec:
127 +   secretName: harbor-registry-internal-tls
128 +   duration: 87600h
129 +   renewBefore: 360h
130 +   organization:
131 +     - Project Harbor
132 +   commonName: harbor-registry
133 +   isCA: false
134 +   keySize: 2048
135 +   keyAlgorithm: rsa
136 +   keyEncoding: pkcs1
137 +   usages:
138 +     - server auth
139 +     - client auth
140 +   dnsNames:
141 +     - harbor-registry
142 +     - harbor-registry.tanzu-system-registry
143 +     - harbor-registry.tanzu-system-registry.svc
144 +     - harbor-registry.tanzu-system-registry.svc.cluster.local
145 +   ipAddresses: []
146 +   issuerRef:
147 +     name: niceneasy-ca
148 +     kind: ClusterIssuer
149 +     group: cert-manager.io
150 + ---
151 + apiVersion: cert-manager.io/v1alpha2
152 + kind: Certificate
153 + metadata:
154 +   name: harbor-trivy-internal-cert
155 +   namespace: tanzu-system-registry
156 + spec:
157 +   secretName: harbor-trivy-internal-tls
158 +   duration: 87600h
159 +   renewBefore: 360h
160 +   organization:
161 +     - Project Harbor
162 +   commonName: harbor-trivy
163 +   isCA: false
164 +   keySize: 2048
165 +   keyAlgorithm: rsa
166 +   keyEncoding: pkcs1
167 +   usages:
168 +     - server auth
```



```
169 + - client auth
170 + dnsNames:
171 + - harbor-trivy
172 + - harbor-trivy.tanzu-system-registry
173 + - harbor-trivy.tanzu-system-registry.svc
174 + - harbor-trivy.tanzu-system-registry.svc.cluster.local
175 + ipAddresses: []
176 + issuerRef:
177 +   name: niceneasy-ca
178 +   kind: ClusterIssuer
179 +   group: cert-manager.io
180 + ---
181 + apiVersion: cert-manager.io/v1alpha2
182 + kind: Certificate
183 + metadata:
184 +   name: harbor-token-service-cert
185 +   namespace: tanzu-system-registry
186 + spec:
187 +   secretName: harbor-token-service
188 +   duration: 87600h
189 +   renewBefore: 360h
190 +   organization:
191 +     - Project Harbor
192 +   commonName: harbor-token-service
193 +   isCA: false
194 +   keySize: 2048
195 +   keyAlgorithm: rsa
196 +   keyEncoding: pkcs1
197 +   usages:
198 +     - server auth
199 +     - client auth
200 +   dnsNames:
201 +     - harbor-token-service
202 +     - harbor-token-service.tanzu-system-registry
203 +     - harbor-token-service.tanzu-system-registry.svc
204 +     - harbor-token-service.tanzu-system-registry.svc.cluster.local
205 +   ipAddresses: []
206 +   issuerRef:
207 +     name: niceneasy-ca
208 +     kind: ClusterIssuer
209 +     group: cert-manager.io
210 + ---
211 + apiVersion: cert-manager.io/v1alpha2
212 + kind: Certificate
213 + metadata:
214 +   name: harbor-notary-signer-cert
215 +   namespace: tanzu-system-registry
216 + spec:
217 +   secretName: harbor-notary-signer
```



```
218 + duration: 87600h
219 + renewBefore: 360h
220 + organization:
221 + - Project Harbor
222 + commonName: harbor-notary-signer
223 + isCA: false
224 + keySize: 2048
225 + keyAlgorithm: rsa
226 + keyEncoding: pkcs1
227 + usages:
228 + - server auth
229 + - client auth
230 + dnsNames:
231 + - harbor-notary-signer
232 + - harbor-notary-signer.tanzu-system-registry
233 + - harbor-notary-signer.tanzu-system-registry.svc
234 + - harbor-notary-signer.tanzu-system-registry.svc.cluster.local
235 + ipAddresses: []
236 + issuerRef:
237 +   name: niceeasy-ca
238 +   kind: ClusterIssuer
239 +   group: cert-manager.io
77 | 240 | ---
78 |   | - apiVersion: v1
79 |   | - kind: Secret
   | 241 | + apiVersion: cert-manager.io/v1alpha2
   | 242 | + kind: Certificate
80 | 243 | metadata:
81 |   | - name: #@ harbor_tls_secret
82 |   | - namespace: #@ harbor_namespace
83 |   | - type: kubernetes.io/tls
84 |   | - data:
85 |   | - #@ if/end ca_cert:
86 |   | - ca.crt: #@ base64.encode(ca_cert)
87 |   | - tls.crt: #@ base64.encode(tls_cert)
88 |   | - tls.key: #@ base64.encode(tls_key)
89 |   | - #@ end ⊖
   | 244 | + name: harbor-tls-cert
   | 245 | + namespace: tanzu-system-registry
   | 246 | + spec:
   | 247 | +   secretName: harbor-tls
   | 248 | +   duration: 87600h
   | 249 | +   renewBefore: 360h
   | 250 | +   organization:
   | 251 | +   - Project Harbor
   | 252 | +   commonName: harbor
   | 253 | +   isCA: false
   | 254 | +   keySize: 2048
   | 255 | +   keyAlgorithm: rsa
```




```

256 +   keyEncoding: pkcs1
257 +   usages:
258 +     - server auth
259 +     - client auth
260 +   dnsNames:
261 +     - harbor.ne.local
262 +     - notary.harbor.ne.local
263 +   ipAddresses: []
264 +   issuerRef:
265 +     name: niceeasy-ca
266 +     kind: ClusterIssuer
267 +     group: cert-manager.io

```

▼ 1 ■■■■■ registry/harbor/02-clair.yaml 📄

```

19 | 19 |         app: "harbor"
20 | 20 |         component: clair
21 | 21 |     spec:
22 | 22 | +     serviceAccountName: "harbor"
23 | 23 |     securityContext:
24 | 24 |         fsGroup: 10000
25 | 25 |     containers:

```

▼ 1 ■■■■■ registry/harbor/03-core.yaml 📄

```

77 | 77 |         app: "harbor"
78 | 78 |         component: core
79 | 79 |     spec:
80 | 80 | +     serviceAccountName: "harbor"
81 | 81 |     securityContext:
82 | 82 |         fsGroup: 10000
83 | 83 |     containers:

```

▼ 1 ■■■■■ registry/harbor/04-database.yaml 📄

```

31 | 31 |         app: "harbor"
32 | 32 |         component: database
33 | 33 |     spec:
34 | 34 | +     serviceAccountName: "harbor"
35 | 35 |     initContainers:
36 | 36 |         - name: "change-permission-of-directory"
37 | 37 |     securityContext:

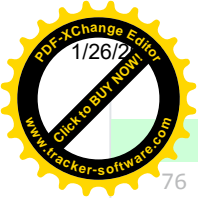
```

▼ 1 ■■■■■ registry/harbor/06-jobservice.yaml 📄

```

73 | 73 |         app: "harbor"
74 | 74 |         component: jobservice

```



75		spec:
76	+	serviceAccountName: "harbor"
76	77	securityContext:
77	78	fsGroup: 10000
78	79	containers:

▼ 1 ■■■ registry/harbor/07-notary.yaml

32	32	app: "harbor"
33	33	component: notary-server
34	34	spec:
35	+	serviceAccountName: "harbor"
35	36	securityContext:
36	37	fsGroup: 10000
37	38	containers:

▼ 1 ■■■ registry/harbor/08-portal.yaml

67	67	app: "harbor"
68	68	component: portal
69	69	spec:
70	+	serviceAccountName: "harbor"
70	71	containers:
71	72	- name: portal
72	73	image: goharbor/harbor-portal:v2.0.2

▼ 1 ■■■ registry/harbor/09-redis.yaml

34	34	app: "harbor"
35	35	component: redis
36	36	spec:
37	+	serviceAccountName: "harbor"
37	38	securityContext:
38	39	fsGroup: 999
39	40	containers:

▼ 1 ■■■ registry/harbor/10-registry.yaml

76	76	app: "harbor"
77	77	component: registry
78	78	spec:
79	+	serviceAccountName: "harbor"
79	80	securityContext:
80	81	fsGroup: 10000
81	82	containers:

▼ 1 ■■■ registry/harbor/11-trivy.yaml



```

32     app: "harbor"
33     component: trivy
34     spec:
35 +   serviceAccountName: "harbor"
36     securityContext:
37       runAsNonRoot: true
38       runAsUser: 10000

```

5 registry/harbor/overlays/change-namespace.yaml

```

21     metadata:
22       namespace: #@ values.namespace
23
24 +   #@overlay/match by=overlay.and_op(kind.serviceaccount, harbor_app),expects="0+"
25 +   ---
26 +   metadata:
27 +     namespace: #@ values.namespace
28 +
29     #@overlay/match by=overlay.and_op(kind.configmap, harbor_app),expects="0+"
30     ---
31     metadata:


```

19 registry/harbor/values.yaml

```

12     namespace: tanzu-system-registry
13
14     # The FQDN for accessing Harbor admin UI and Registry service.
15 -   hostname: core.harbor.domain
16 +   hostname: harbor.ne.local
17
18     # The network port of the Envoy service in Contour or other Ingress Controller.
19     port:
20       https: 443
21
22     enableContourHttpProxy: true
23
24     # [Required] The initial password of Harbor admin.
25 -   harborAdminPassword:
26 +   harborAdminPassword:
27
28     # [Required] The secret key used for encryption. Must be a string of 16 chars.
29 -   secretKey:
30 +   secretKey: 27sXXX0MsZ056LCT
31
32     database:
33       # [Required] The initial password of the postgres database.
34 -   password:
35 +   password: ugi9CNpdWccT2thZ
36
37     core:
38       replicas: 1

```

46		# [Required] Secret is used when core server communicates with other componen
	-	secret:
47	+	secret: <u>G08UMgV2X6KW8Y2z</u>
48	48	# [Required] The XSRF key. Must be a string of 32 chars.
49	-	xsrfKey:
49	+	xsrfKey: gmjmRQEhoZw930xg2NDSAbFcYtPQJos6
50	50	+
50	51	jobservice:
51	52	replicas: 1
52	53	# [Required] Secret is used when job service communicates with other components.
53	-	secret:
54	+	secret: <u>XSLncDXvbHrCY1CP</u>
54	55	registry:
55	56	replicas: 1
56	57	# [Required] Secret is used to secure the upload state from client
57	58	# and registry storage backend.
58	59	# See: https://github.com/docker/distribution/blob/master/docs/configuration.md#htt
59	-	secret:
60	+	secret: <u>ktvJ0mrQdn90DMHp</u>
60	61	notary:
61	62	# Whether to install Notary
62	63	enabled: true
209	210	proxy:
210	211	httpProxy:
211	212	httpsProxy:
212	-	noProxy: 127.0.0.1,localhost,.local,.internal
213	+	noProxy: 127.0.0.1,localhost,.local,.internal 



No commit comments for this range