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 @@
+ history-07.txt
+ history-06.txt
+ history-05.txt
+ history-04.txt
+ history-03.txt
+ registry/harbor/history-02.txt
+ ingress/examples/https-ingress/check-history.txt
```

57 ingress/contour/02-certs-ca-issued.yaml

```
@@ -0,0 +1,57 @@
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: contour-cert
+   namespace: tanzu-system-ingress
+ spec:
+   secretName: contourcert
+   duration: 8760h
+   renewBefore: 360h
+   organization:
+   - Project Contour
+   commonName: contour
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   dnsNames:
+   - contour
+   - contour.tanzu-system-ingress
+   - contour.tanzu-system-ingress.svc
+   - contour.tanzu-system-ingress.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
```

```
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: envoy-cert
+   namespace: tanzu-system-ingress
+ spec:
+   secretName: envoycert
+   duration: 8760h
+   renewBefore: 360h
+   organization:
+   - Project Contour
+   commonName: envoy
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - client auth
+   dnsNames:
+   - envoy
+   - envoy.tanzu-system-ingress
+   - envoy.tanzu-system-ingress.svc
+   - envoy.tanzu-system-ingress.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
```

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

Load diff

This file was deleted.

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

```
  subjects:
  #@overlay/match by=kind.serviceaccount
  - kind: ServiceAccount
-   namespace: #@ values.contour.namespace
+   namespace: #@ values.contour.namespace
```

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

```
17 17          app: hello
18 18          tier: web
19 19      spec:
   20 +       serviceAccountName: sample
20 21          containers:
21 22          - name: hello-app
22 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 @@
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: grafana-tls-cert
+   namespace: tanzu-system-monitoring
+ spec:
+   secretName: grafana-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Grafana
+   commonName: grafana
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - grafana.tanzu.ne.local
+   - notary.grafana.tanzu.ne.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     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 @@
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: prometheus-ca
```

```
+   namespace: tanzu-system-monitoring
+ spec:
+   secretName: prometheus-ca-key-pair
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Prometheus
+   commonName: Prometheus CA
+   isCA: true
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - prometheusca
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: prometheus-tls-cert
+   namespace: tanzu-system-monitoring
+ spec:
+   secretName: prometheus-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Prometheus
+   commonName: prometheus
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - prometheus.tanzu.ne.local
+   - notary.prometheus.tanzu.ne.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     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  +     DQEJARYbZGFuaWVsZS51bHJpY2hAbmljZW5lYXN5LmNoMQswCQYDVQQGEwJDSDEN
7  +     MAsGA1UECAwEQmVybjEWMBQGA1UEBwwNTcO8aGxldGh1cm5lbjETMBEGA1UECgwK
8  +     bmljZW5lYXN5IDEUMBIGA1UECwwLRGV2ZWxvcG1lbnQxGDAWBgNVBAMMD2NhLm5p
9  +     Y2VuZWFzeS5jaDAeFw0yMTAxMTgwMDIzMDRaFw0yNDAxMTgwMDIzMDRaMIGlMSow
10 +     KAYJKoZIhvcNAQkBFhtkYW5pZWxlLnVscmljaEBuaWNlbmVhc3kuY2gxCzAJBgNV
11 +     BAYTAkNIMQ0wCwYDVQQIDARCZXJuMRYwFAYDVQQHDA1Nw7xobGV0aHVybmVuMRMw
12 +     EQYDVQQKDApuaWNlbmVhc3kgMRQwEgYDVQQLDAtEZXZlbG9wbWVudDEYMBYGA1UE
13 +     AwwPY2EubmljZW5lYXN5LmNoMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
14 +     AQEAu6t+hNEwhxyIWyW/PbhYZK4DcsTBIFOjJrAQYK1HDDjiZXVf9G63bFipWyRa
15 +     +wb3KN9pJZO75yiiGr5lx/keJNQ5zICEP2Jx2jhOc1u0sETGHUg7LgU17CE6YAXt
16 +     dHNBBEAWGrl7XtRWUbae+30HYDXDxwBza0M1jpbE3u/EPVgFRtQelSyupNJHC6xE
17 +     N19IUrTYQcrjhyRQsEjnl6VAxNg3qhzI1fY5c9s08ur9KM1eA6vniCKbCQP1V2dk
18 +     aqqEng7c1fky9sL0Et8GnlUFd8ev4IgPtPIRejK2WK9RzM4CB6YORMLyO34WYx8y
19 +     3S2zDyHg2QtGW6qCeQ3DKDy+NwIDAQABo2cwZTAPBgNVHRMECDAGAQH/AgEDMAsG
20 +     A1UdDwQEAwIBBjBFBgNVHSUEPjA8BggrBgEFBQcDAQYIKwYBBQUHAwIGCCsGAQUF
21 +     BwMDBggrBgEFBQcDBAYIKwYBBQUHAwgGCCsGAQUFBwMJMA0GCSqGSIb3DQEBCwUA
22 +     A4IBAQCBmV8QxhLXI9rf2dfLfj8zkxt70u3ms0oxdTMR7cix3Gw57BfALSjcXeOd
23 +     hELNEgHtFUL6xznFDnMCsw3LR8WF0tZtCYN+DFmVpdmAt/JLVvKCszkGGdeT4GzH
24 +     8zEmBCf4xLrf6uCYwomTY/ZiN2wngXlqNpEIDR9jtq6cuy1XhmLaI3gzPTPvabxm
25 +     JjPVQDcgcHEqKNMGICr2JF26izqkKFSC7yusDy/+SYNQRi+YzeoSgP4FP3bSl6dc
26 +     GRrCRwdUneiGLmNsYFn8nodl2KSxEROfWt0JXVqemQGc9rWz42a/hqzNSu7VamWS
27 +     dJLLRulNSYA1END87glIx6+181ot
28 +     -----END CERTIFICATE-----
29 + kind: ConfigMap
30 + metadata:
31 +   name: trusted-ca-cm
32 +   namespace: cert-manager
33 + ---
34 + apiVersion: v1
```

```
+ data:
+     build-ca.sh: "#!/usr/bin/env bash \nset -euxo pipefail\ntdnf install -y open -c
+ kind: ConfigMap
+ metadata:
+     name: rehash-script
+     namespace: cert-manager
+ ---
+ apiVersion: apps/v1
+ kind: DaemonSet
+ metadata:
+   name: trusted-ca-updater
+   namespace: cert-manager
+   labels:
+     k8s-app: trusted-ca-updater
+ spec:
+   selector:
+     matchLabels:
+       name: trusted-ca-updater
+   template:
+     metadata:
+       labels:
+         name: trusted-ca-updater
+     spec:
+       serviceAccountName: cert-manager
+       tolerations:
+       # this toleration is to have the daemonset runnable on master nodes
+       # remove it if your masters can't run pods
+       - key: node-role.kubernetes.io/master
+         effect: NoSchedule
+       initContainers:
+       - name: script-runner
+         image: photon:3.0
+         command: ["/bin/sh", "-c", "/root/build-ca.sh" ]
+         volumeMounts:
+         - name: update-trusted-certs-script
+           mountPath: /root/
+         - name: certs-dir
+           mountPath: /etc/ssl/certs
+         - name: agg-certs-dir
+           mountPath: /etc/pki/tls/certs/
+         env:
+         - name: TRUSTED_CERT
+           valueFrom:
+             configMapKeyRef:
+               name: trusted-ca-cm
+               key: ca.pem
+         resources:
+             limits:
+               ephemeral-storage: 4G
```

```
+       containers:
+       - name: sleepy
+         image: photon:3.0
+         command: ["/bin/sh"]
+         args: ["-c", "while true; do sleep 3600;done"]
+       volumes:
+       - name: update-trusted-certs-script
+         configMap:
+             name: rehash-script
+             defaultMode: 0766
+       - name: certs-dir
+         hostPath:
+           path: /etc/ssl/certs
+           type: Directory
+       - name: agg-certs-dir
+         hostPath:
+           path: /etc/pki/tls/certs/
+           type: Directory
```

21 registry/harbor/00-common.yaml

```
    name: harbor-system
    labels:
      app: "harbor"
+ ---
+ apiVersion: v1
+ kind: ServiceAccount
+ metadata:
+   name: harbor
+   namespace: "harbor-system"
+   labels:
+     app: "harbor"
+ ---
+ apiVersion: rbac.authorization.k8s.io/v1
+ kind: ClusterRoleBinding
+ metadata:
+   name: harbor
+ roleRef:
+   apiGroup: rbac.authorization.k8s.io
+   kind: ClusterRole
+   name: psp:vmware-system-privileged
+ subjects:
+ - kind: ServiceAccount
+   name: harbor
+   namespace: "tanzu-system-registry"
```

352 registry/harbor/01-certificate.yaml

```
...  @@ -1,89 +1,267 @@
- #@ load("/values.star", "values")
- #@ load("/helpers.star", "generate_harbor_tls")
- #@ load("@ytt:base64", "base64")
- #@ load("@ytt:overlay", "overlay")
- #@ load("@ytt:template", "template")
- #@ load("certificates.lib.yaml", "generate_dns_names", "generate_self_signed_issuer",
-
- #@ harbor_name = "harbor"
- #@ harbor_namespace = values.namespace
- #@ harbor_organization = "Project Harbor"
- #@ harbor_self_signed_ca_issuer = harbor_name + "-self-signed-ca-issuer"
- #@ harbor_ca = harbor_name + "-ca"
- #@ harbor_ca_common_name = "Harbor CA"
- #@ harbor_ca_dns_name = harbor_name + "ca"
- #@ harbor_ca_key_pair = harbor_name + "-ca-key-pair"
- #@ harbor_ca_issuer = harbor_name + "-ca-issuer"
- #@ harbor_cert = harbor_name + "-cert"
- #@ harbor_cert_duration = "87600h"
- #@ harbor_cert_renew_before = "360h"
-
- --- #@ generate_self_signed_issuer(harbor_self_signed_ca_issuer, harbor_namespace)
- --- #@ generate_ca_certificate(harbor_ca, harbor_namespace, harbor_cert_duration, har
- --- #@ generate_ca_issuer(harbor_ca_issuer, harbor_namespace, harbor_ca_key_pair)
-
- #@ harbor_clair_internal_tls_cert = "harbor-clair-internal-cert"
- #@ harbor_clair_internal_tls_secret = "harbor-clair-internal-tls"
- #@ harbor_clair_common_name = "harbor-clair"
- --- #@ generate_certificate(harbor_clair_internal_tls_cert, harbor_namespace, harbor_
-
- #@ harbor_core_internal_tls_cert = "harbor-core-internal-cert"
- #@ harbor_core_internal_tls_secret = "harbor-core-internal-tls"
- #@ harbor_core_common_name = "harbor-core"
- --- #@ generate_certificate(harbor_core_internal_tls_cert, harbor_namespace, harbor_c
-
- #@ harbor_jobservice_internal_tls_cert = "harbor-jobservice-internal-cert"
- #@ harbor_jobservice_internal_tls_secret = "harbor-jobservice-internal-tls"
- #@ harbor_jobservice_common_name = "harbor-jobservice"
- --- #@ generate_certificate(harbor_jobservice_internal_tls_cert, harbor_namespace, ha
-
- #@ harbor_portal_internal_tls_cert = "harbor-portal-internal-cert"
- #@ harbor_portal_internal_tls_secret = "harbor-portal-internal-tls"
- #@ harbor_portal_common_name = "harbor-portal"
- --- #@ generate_certificate(harbor_portal_internal_tls_cert, harbor_namespace, harbor
-
- #@ harbor_registry_internal_tls_cert = "harbor-registry-internal-cert"
- #@ harbor_registry_internal_tls_secret = "harbor-registry-internal-tls"
- #@ harbor_registry_common_name = "harbor-registry"
- --- #@ generate_certificate(harbor_registry_internal_tls_cert, harbor_namespace, harb
```

```
-
- #@ harbor_trivy_internal_tls_cert = "harbor-trivy-internal-cert"
- #@ harbor_trivy_internal_tls_secret = "harbor-trivy-internal-tls"
- #@ harbor_trivy_common_name = "harbor-trivy"
- --- #@ generate_certificate(harbor_trivy_internal_tls_cert, harbor_namespace, harbor_
-
- #@ harbor_token_service_cert = "harbor-token-service-cert"
- #@ harbor_token_service_secret = "harbor-token-service"
- #@ harbor_token_service_common_name = "harbor-token-service"
- --- #@ generate_certificate(harbor_token_service_cert, harbor_namespace, harbor_cert_
-
- #@ harbor_notary_signer_cert = "harbor-notary-signer-cert"
- #@ harbor_notary_signer_secret = "harbor-notary-signer"
- #@ harbor_notary_signer_common_name = "harbor-notary-signer"
- --- #@ generate_certificate(harbor_notary_signer_cert, harbor_namespace, harbor_cert_
-
-
- #@ harbor_tls_cert = "harbor-tls-cert"
- #@ harbor_tls_secret = "harbor-tls"
- #@ harbor_tls_common_name = "harbor"
- #@ if generate_harbor_tls():
- --- #@ generate_certificate(harbor_tls_cert, harbor_namespace, harbor_cert_duration,
- #@ end
-
- #@ if not generate_harbor_tls():
- #@ ca_crt = getattr(values.tlsCertificate, "ca.crt")
- #@ tls_crt = getattr(values.tlsCertificate, "tls.crt")
- #@ tls_key = getattr(values.tlsCertificate, "tls.key")
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-clair-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-clair-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-clair
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-clair
```

```
+   - harbor-clair.tanzu-system-registry
+   - harbor-clair.tanzu-system-registry.svc
+   - harbor-clair.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-core-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-core-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-core
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-core
+   - harbor-core.tanzu-system-registry
+   - harbor-core.tanzu-system-registry.svc
+   - harbor-core.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-jobservice-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-jobservice-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
```

```
+   - Project Harbor
+   commonName: harbor-jobservice
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-jobservice
+   - harbor-jobservice.tanzu-system-registry
+   - harbor-jobservice.tanzu-system-registry.svc
+   - harbor-jobservice.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-portal-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-portal-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-portal
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-portal
+   - harbor-portal.tanzu-system-registry
+   - harbor-portal.tanzu-system-registry.svc
+   - harbor-portal.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
```

```
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-registry-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-registry-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-registry
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-registry
+   - harbor-registry.tanzu-system-registry
+   - harbor-registry.tanzu-system-registry.svc
+   - harbor-registry.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+   name: harbor-trivy-internal-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-trivy-internal-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-trivy
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
```

```
+    - client auth
+    dnsNames:
+    - harbor-trivy
+    - harbor-trivy.tanzu-system-registry
+    - harbor-trivy.tanzu-system-registry.svc
+    - harbor-trivy.tanzu-system-registry.svc.cluster.local
+    ipAddresses: []
+    issuerRef:
+      name: niceneasy-ca
+      kind: ClusterIssuer
+      group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+    name: harbor-token-service-cert
+    namespace: tanzu-system-registry
+ spec:
+    secretName: harbor-token-service
+    duration: 87600h
+    renewBefore: 360h
+    organization:
+    - Project Harbor
+    commonName: harbor-token-service
+    isCA: false
+    keySize: 2048
+    keyAlgorithm: rsa
+    keyEncoding: pkcs1
+    usages:
+    - server auth
+    - client auth
+    dnsNames:
+    - harbor-token-service
+    - harbor-token-service.tanzu-system-registry
+    - harbor-token-service.tanzu-system-registry.svc
+    - harbor-token-service.tanzu-system-registry.svc.cluster.local
+    ipAddresses: []
+    issuerRef:
+      name: niceneasy-ca
+      kind: ClusterIssuer
+      group: cert-manager.io
+ ---
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
+ metadata:
+    name: harbor-notary-signer-cert
+    namespace: tanzu-system-registry
+ spec:
+    secretName: harbor-notary-signer
```

```
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor-notary-signer
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor-notary-signer
+   - harbor-notary-signer.tanzu-system-registry
+   - harbor-notary-signer.tanzu-system-registry.svc
+   - harbor-notary-signer.tanzu-system-registry.svc.cluster.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
  ---
- apiVersion: v1
- kind: Secret
+ apiVersion: cert-manager.io/v1alpha2
+ kind: Certificate
  metadata:
-   name: #@ harbor_tls_secret
-   namespace: #@ harbor_namespace
- type: kubernetes.io/tls
- data:
-   #@ if/end ca_crt:
-   ca.crt: #@ base64.encode(ca_crt)
-   tls.crt: #@ base64.encode(tls_crt)
-   tls.key: #@ base64.encode(tls_key)
- #@ end
+   name: harbor-tls-cert
+   namespace: tanzu-system-registry
+ spec:
+   secretName: harbor-tls
+   duration: 87600h
+   renewBefore: 360h
+   organization:
+   - Project Harbor
+   commonName: harbor
+   isCA: false
+   keySize: 2048
+   keyAlgorithm: rsa
```

```
+   keyEncoding: pkcs1
+   usages:
+   - server auth
+   - client auth
+   dnsNames:
+   - harbor.ne.local
+   - notary.harbor.ne.local
+   ipAddresses: []
+   issuerRef:
+     name: niceneasy-ca
+     kind: ClusterIssuer
+     group: cert-manager.io
```

1 registry/harbor/02-clair.yaml

```
          app: "harbor"
          component: clair
      spec:
+       serviceAccountName: "harbor"
        securityContext:
          fsGroup: 10000
        containers:
```

1 registry/harbor/03-core.yaml

```
          app: "harbor"
          component: core
      spec:
+       serviceAccountName: "harbor"
        securityContext:
          fsGroup: 10000
        containers:
```

1 registry/harbor/04-database.yaml

```
          app: "harbor"
          component: database
      spec:
+       serviceAccountName: "harbor"
        initContainers:
          - name: "change-permission-of-directory"
            securityContext:
```

1 registry/harbor/06-jobservice.yaml

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

19 registry/harbor/values.yaml

```
12  12  namespace: tanzu-system-registry
13  13
14  14  # The FQDN for accessing Harbor admin UI and Registry service.
15  -   hostname: core.harbor.domain
    15  hostname: harbor.ne.local
16  16  # The network port of the Envoy service in Contour or other Ingress Controller.
17  17  port:
18  18    https: 443
32  32  enableContourHttpProxy: true
33  33
34  34  # [Required] The initial password of Harbor admin.
35  -   harborAdminPassword:
    35  harborAdminPassword:
36  36
37  37  # [Required] The secret key used for encryption. Must be a string of 16 chars.
38  -   secretKey:
    38  secretKey: 27sXXX0MsZO56LCT
39  39
40  40  database:
41  41    # [Required] The initial password of the postgres database.
42  -     password:
    42    password: ugi9CNpdWccT2thZ
43  43
44  44  core:
45  45    replicas: 1
```

```
46  46      # [Required] Secret is used when core server communicates with other componen
    -       secret:
    47  +   secret: GO8UMgV2X6KW8Y2z
48  48      # [Required] The XSRF key. Must be a string of 32 chars.
49  -       xsrfKey:
    49  +   xsrfKey: gmjmRQEhoZw93Oxg2NDSAbFcYtPQJos6
    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: XSLncDXvbHrCY1CP
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: ktvJ0mrQdn90DMHp
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