Cài đặt Ingress Nginx với Helm

**1. Các bước thực hiện**

**- Thêm repository helm chart của Ingress Nginx**

helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx

**- Tùy chỉnh file values của helm chart ArgoCD**

**Tạo 1 file ingress-values.yaml với các tham số như bên dưới**

## nginx configuration

## Ref: https://github.com/kubernetes/ingress-nginx/blob/main/docs/user-guide/nginx-configuration/index.md

##

​

## Overrides for generated resource names

# See templates/\_helpers.tpl

# nameOverride:

# fullnameOverride:

​

# -- Override the deployment namespace; defaults to .Release.Namespace

namespaceOverride: ""

## Labels to apply to all resources

##

commonLabels: {}

# scmhash: abc123

# myLabel: aakkmd

​

controller:

name: controller

enableAnnotationValidations: false

image:

## Keep false as default for now!

chroot: false

registry: registry.k8s.io

image: ingress-nginx/controller

## for backwards compatibility consider setting the full image url via the repository value below

## use \*either\* current default registry/image or repository format or installing chart by providing the values.yaml will fail

## repository:

tag: "v1.9.6"

digest: sha256:1405cc613bd95b2c6edd8b2a152510ae91c7e62aea4698500d23b2145960ab9c

digestChroot: sha256:7eb46ff733429e0e46892903c7394aff149ac6d284d92b3946f3baf7ff26a096

pullPolicy: IfNotPresent

runAsNonRoot: true

# www-data -> uid 101

runAsUser: 101

allowPrivilegeEscalation: false

seccompProfile:

type: RuntimeDefault

readOnlyRootFilesystem: false

# -- Use an existing PSP instead of creating one

existingPsp: ""

# -- Configures the controller container name

containerName: controller

# -- Configures the ports that the nginx-controller listens on

containerPort:

http: 80

https: 443

# -- Will add custom configuration options to Nginx https://kubernetes.github.io/ingress-nginx/user-guide/nginx-configuration/configmap/

config: {}

# -- Annotations to be added to the controller config configuration configmap.

configAnnotations: {}

# -- Will add custom headers before sending traffic to backends according to https://github.com/kubernetes/ingress-nginx/tree/main/docs/examples/customization/custom-headers

proxySetHeaders: {}

# -- Will add custom headers before sending response traffic to the client according to: https://kubernetes.github.io/ingress-nginx/user-guide/nginx-configuration/configmap/#add-headers

addHeaders: {}

# -- Optionally customize the pod dnsConfig.

dnsConfig: {}

# -- Optionally customize the pod hostAliases.

hostAliases: []

# - ip: 127.0.0.1

# hostnames:

# - foo.local

# - bar.local

# - ip: 10.1.2.3

# hostnames:

# - foo.remote

# - bar.remote

# -- Optionally customize the pod hostname.

hostname: {}

# -- Optionally change this to ClusterFirstWithHostNet in case you have 'hostNetwork: true'.

# By default, while using host network, name resolution uses the host's DNS. If you wish nginx-controller

# to keep resolving names inside the k8s network, use ClusterFirstWithHostNet.

dnsPolicy: ClusterFirst

# -- Bare-metal considerations via the host network https://kubernetes.github.io/ingress-nginx/deploy/baremetal/#via-the-host-network

# Ingress status was blank because there is no Service exposing the Ingress-Nginx Controller in a configuration using the host network, the default --publish-service flag used in standard cloud setups does not apply

reportNodeInternalIp: false

# -- Process Ingress objects without ingressClass annotation/ingressClassName field

# Overrides value for --watch-ingress-without-class flag of the controller binary

# Defaults to false

watchIngressWithoutClass: false

# -- Process IngressClass per name (additionally as per spec.controller).

ingressClassByName: false

# -- This configuration enables Topology Aware Routing feature, used together with service annotation service.kubernetes.io/topology-mode="auto"

# Defaults to false

enableTopologyAwareRouting: false

# -- This configuration defines if Ingress Controller should allow users to set

# their own \*-snippet annotations, otherwise this is forbidden / dropped

# when users add those annotations.

# Global snippets in ConfigMap are still respected

90

allowSnippetAnnotations: false

91

# -- Required for use with CNI based kubernetes installations (such as ones set up by kubeadm),

92

# since CNI and hostport don't mix yet. Can be deprecated once https://github.com/kubernetes/kubernetes/issues/23920

93

# is merged

94

hostNetwork: false

95

## Use host ports 80 and 443

96

## Disabled by default

97

hostPort:

98

# -- Enable 'hostPort' or not

99

enabled: true

100

ports:

101

# -- 'hostPort' http port

102

http: 80

103

# -- 'hostPort' https port

104

https: 443

105

# NetworkPolicy for controller component.

106

networkPolicy:

107

# -- Enable 'networkPolicy' or not

108

enabled: false

109

# -- Election ID to use for status update, by default it uses the controller name combined with a suffix of 'leader'

110

electionID: ""

111

## This section refers to the creation of the IngressClass resource

112

## IngressClass resources are supported since k8s >= 1.18 and required since k8s >= 1.19

113

ingressClassResource:

114

# -- Name of the ingressClass

115

name: nginx

116

# -- Is this ingressClass enabled or not

117

enabled: true

118

# -- Is this the default ingressClass for the cluster

119

default: false

120

# -- Controller-value of the controller that is processing this ingressClass

121

controllerValue: "k8s.io/ingress-nginx"

122

# -- Parameters is a link to a custom resource containing additional

123

# configuration for the controller. This is optional if the controller

124

# does not require extra parameters.

125

parameters: {}

126

# -- For backwards compatibility with ingress.class annotation, use ingressClass.

127

# Algorithm is as follows, first ingressClassName is considered, if not present, controller looks for ingress.class annotation

128

ingressClass: nginx

129

# -- Labels to add to the pod container metadata

130

podLabels: {}

131

# key: value

132

​

133

# -- Security context for controller pods

134

podSecurityContext: {}

135

# -- sysctls for controller pods

136

## Ref: https://kubernetes.io/docs/tasks/administer-cluster/sysctl-cluster/

137

sysctls: {}

138

# sysctls:

139

# "net.core.somaxconn": "8192"

140

# -- Security context for controller containers

141

containerSecurityContext: {}

142

# -- Allows customization of the source of the IP address or FQDN to report

143

# in the ingress status field. By default, it reads the information provided

144

# by the service. If disable, the status field reports the IP address of the

145

# node or nodes where an ingress controller pod is running.

146

publishService:

147

# -- Enable 'publishService' or not

148

enabled: true

149

# -- Allows overriding of the publish service to bind to

150

# Must be <namespace>/<service\_name>

151

pathOverride: ""

152

# Limit the scope of the controller to a specific namespace

153

scope:

154

# -- Enable 'scope' or not

155

enabled: false

156

# -- Namespace to limit the controller to; defaults to $(POD\_NAMESPACE)

157

namespace: ""

158

# -- When scope.enabled == false, instead of watching all namespaces, we watching namespaces whose labels

159

# only match with namespaceSelector. Format like foo=bar. Defaults to empty, means watching all namespaces.

160

namespaceSelector: ""

161

# -- Allows customization of the configmap / nginx-configmap namespace; defaults to $(POD\_NAMESPACE)

162

configMapNamespace: ""

163

tcp:

164

# -- Allows customization of the tcp-services-configmap; defaults to $(POD\_NAMESPACE)

165

configMapNamespace: ""

166

# -- Annotations to be added to the tcp config configmap

167

annotations: {}

168

udp:

169

# -- Allows customization of the udp-services-configmap; defaults to $(POD\_NAMESPACE)

170

configMapNamespace: ""

171

# -- Annotations to be added to the udp config configmap

172

annotations: {}

173

# -- Maxmind license key to download GeoLite2 Databases.

174

## https://blog.maxmind.com/2019/12/18/significant-changes-to-accessing-and-using-geolite2-databases

175

maxmindLicenseKey: ""

176

# -- Additional command line arguments to pass to Ingress-Nginx Controller

177

# E.g. to specify the default SSL certificate you can use

178

extraArgs: {}

179

## extraArgs:

180

## default-ssl-certificate: "<namespace>/<secret\_name>"

181

## time-buckets: "0.005,0.01,0.025,0.05,0.1,0.25,0.5,1,2.5,5,10"

182

## length-buckets: "10,20,30,40,50,60,70,80,90,100"

183

## size-buckets: "10,100,1000,10000,100000,1e+06,1e+07"

184

​

185

# -- Additional environment variables to set

186

extraEnvs: []

187

# extraEnvs:

188

# - name: FOO

189

# valueFrom:

190

# secretKeyRef:

191

# key: FOO

192

# name: secret-resource

193

​

194

# -- Use a `DaemonSet` or `Deployment`

195

kind: DaemonSet

196

# -- Annotations to be added to the controller Deployment or DaemonSet

197

##

198

annotations: {}

199

# keel.sh/pollSchedule: "@every 60m"

200

​

201

# -- Labels to be added to the controller Deployment or DaemonSet and other resources that do not have option to specify labels

202

##

203

labels: {}

204

# keel.sh/policy: patch

205

# keel.sh/trigger: poll

206

​

207

# -- The update strategy to apply to the Deployment or DaemonSet

208

##

209

updateStrategy: {}

210

# rollingUpdate:

211

# maxUnavailable: 1

212

# type: RollingUpdate

213

​

214

# -- `minReadySeconds` to avoid killing pods before we are ready

215

##

216

minReadySeconds: 0

217

# -- Node tolerations for server scheduling to nodes with taints

218

## Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-node/

219

##

220

tolerations: []

221

# - key: "key"

222

# operator: "Equal|Exists"

223

# value: "value"

224

# effect: "NoSchedule|PreferNoSchedule|NoExecute(1.6 only)"

225

​

226

# -- Affinity and anti-affinity rules for server scheduling to nodes

227

## Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-node/#affinity-and-anti-affinity

228

##

229

affinity: {}

230

# # An example of preferred pod anti-affinity, weight is in the range 1-100

231

# podAntiAffinity:

232

# preferredDuringSchedulingIgnoredDuringExecution:

233

# - weight: 100

234

# podAffinityTerm:

235

# labelSelector:

236

# matchExpressions:

237

# - key: app.kubernetes.io/name

238

# operator: In

239

# values:

240

# - ingress-nginx

241

# - key: app.kubernetes.io/instance

242

# operator: In

243

# values:

244

# - ingress-nginx

245

# - key: app.kubernetes.io/component

246

# operator: In

247

# values:

248

# - controller

249

# topologyKey: kubernetes.io/hostname

250

​

251

# # An example of required pod anti-affinity

252

# podAntiAffinity:

253

# requiredDuringSchedulingIgnoredDuringExecution:

254

# - labelSelector:

255

# matchExpressions:

256

# - key: app.kubernetes.io/name

257

# operator: In

258

# values:

259

# - ingress-nginx

260

# - key: app.kubernetes.io/instance

261

# operator: In

262

# values:

263

# - ingress-nginx

264

# - key: app.kubernetes.io/component

265

# operator: In

266

# values:

267

# - controller

268

# topologyKey: "kubernetes.io/hostname"

269

​

270

# -- Topology spread constraints rely on node labels to identify the topology domain(s) that each Node is in.

271

## Ref: https://kubernetes.io/docs/concepts/workloads/pods/pod-topology-spread-constraints/

272

##

273

topologySpreadConstraints: []

274

# - labelSelector:

275

# matchLabels:

276

# app.kubernetes.io/name: '{{ include "ingress-nginx.name" . }}'

277

# app.kubernetes.io/instance: '{{ .Release.Name }}'

278

# app.kubernetes.io/component: controller

279

# topologyKey: topology.kubernetes.io/zone

280

# maxSkew: 1

281

# whenUnsatisfiable: ScheduleAnyway

282

# - labelSelector:

283

# matchLabels:

284

# app.kubernetes.io/name: '{{ include "ingress-nginx.name" . }}'

285

# app.kubernetes.io/instance: '{{ .Release.Name }}'

286

# app.kubernetes.io/component: controller

287

# topologyKey: kubernetes.io/hostname

288

# maxSkew: 1

289

# whenUnsatisfiable: ScheduleAnyway

290

​

291

# -- `terminationGracePeriodSeconds` to avoid killing pods before we are ready

292

## wait up to five minutes for the drain of connections

293

##

294

terminationGracePeriodSeconds: 300

295

# -- Node labels for controller pod assignment

296

## Ref: https://kubernetes.io/docs/concepts/scheduling-eviction/assign-pod-node/

297

##

298

nodeSelector:

299

kubernetes.io/os: linux

300

## Liveness and readiness probe values

301

## Ref: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#container-probes

302

##

303

## startupProbe:

304

## httpGet:

305

## # should match container.healthCheckPath

306

## path: "/healthz"

307

## port: 10254

308

## scheme: HTTP

309

## initialDelaySeconds: 5

310

## periodSeconds: 5

311

## timeoutSeconds: 2

312

## successThreshold: 1

313

## failureThreshold: 5

314

livenessProbe:

315

httpGet:

316

# should match container.healthCheckPath

317

path: "/healthz"

318

port: 10254

319

scheme: HTTP

320

initialDelaySeconds: 10

321

periodSeconds: 10

322

timeoutSeconds: 1

323

successThreshold: 1

324

failureThreshold: 5

325

readinessProbe:

326

httpGet:

327

# should match container.healthCheckPath

328

path: "/healthz"

329

port: 10254

330

scheme: HTTP

331

initialDelaySeconds: 10

332

periodSeconds: 10

333

timeoutSeconds: 1

334

successThreshold: 1

335

failureThreshold: 3

336

# -- Path of the health check endpoint. All requests received on the port defined by

337

# the healthz-port parameter are forwarded internally to this path.

338

healthCheckPath: "/healthz"

339

# -- Address to bind the health check endpoint.

340

# It is better to set this option to the internal node address

341

# if the Ingress-Nginx Controller is running in the `hostNetwork: true` mode.

342

healthCheckHost: ""

343

# -- Annotations to be added to controller pods

344

##

345

podAnnotations: {}

346

replicaCount: 1

347

# -- Minimum available pods set in PodDisruptionBudget.

348

# Define either 'minAvailable' or 'maxUnavailable', never both.

349

minAvailable: 1

350

# -- Maximum unavailable pods set in PodDisruptionBudget. If set, 'minAvailable' is ignored.

351

# maxUnavailable: 1

352

​

353

## Define requests resources to avoid probe issues due to CPU utilization in busy nodes

354

## ref: https://github.com/kubernetes/ingress-nginx/issues/4735#issuecomment-551204903

355

## Ideally, there should be no limits.

356

## https://engineering.indeedblog.com/blog/2019/12/cpu-throttling-regression-fix/

357

resources:

358

## limits:

359

## cpu: 100m

360

## memory: 90Mi

361

requests:

362

cpu: 100m

363

memory: 90Mi

364

# Mutually exclusive with keda autoscaling

365

autoscaling:

366

enabled: false

367

annotations: {}

368

minReplicas: 1

369

maxReplicas: 11

370

targetCPUUtilizationPercentage: 50

371

targetMemoryUtilizationPercentage: 50

372

behavior: {}

373

# scaleDown:

374

# stabilizationWindowSeconds: 300

375

# policies:

376

# - type: Pods

377

# value: 1

378

# periodSeconds: 180

379

# scaleUp:

380

# stabilizationWindowSeconds: 300

381

# policies:

382

# - type: Pods

383

# value: 2

384

# periodSeconds: 60

385

autoscalingTemplate: []

386

# Custom or additional autoscaling metrics

387

# ref: https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/#support-for-custom-metrics

388

# - type: Pods

389

# pods:

390

# metric:

391

# name: nginx\_ingress\_controller\_nginx\_process\_requests\_total

392

# target:

393

# type: AverageValue

394

# averageValue: 10000m

395

​

396

# Mutually exclusive with hpa autoscaling

397

keda:

398

apiVersion: "keda.sh/v1alpha1"

399

## apiVersion changes with keda 1.x vs 2.x

400

## 2.x = keda.sh/v1alpha1

401

## 1.x = keda.k8s.io/v1alpha1

402

enabled: false

403

minReplicas: 1

404

maxReplicas: 11

405

pollingInterval: 30

406

cooldownPeriod: 300

407

# fallback:

408

# failureThreshold: 3

409

# replicas: 11

410

restoreToOriginalReplicaCount: false

411

scaledObject:

412

annotations: {}

413

# Custom annotations for ScaledObject resource

414

# annotations:

415

# key: value

416

triggers: []

417

# - type: prometheus

418

# metadata:

419

# serverAddress: http://<prometheus-host>:9090

420

# metricName: http\_requests\_total

421

# threshold: '100'

422

# query: sum(rate(http\_requests\_total{deployment="my-deployment"}[2m]))

423

​

424

behavior: {}

425

# scaleDown:

426

# stabilizationWindowSeconds: 300

427

# policies:

428

# - type: Pods

429

# value: 1

430

# periodSeconds: 180

431

# scaleUp:

432

# stabilizationWindowSeconds: 300

433

# policies:

434

# - type: Pods

435

# value: 2

436

# periodSeconds: 60

437

# -- Enable mimalloc as a drop-in replacement for malloc.

438

## ref: https://github.com/microsoft/mimalloc

439

##

440

enableMimalloc: true

441

## Override NGINX template

442

customTemplate:

443

configMapName: ""

444

configMapKey: ""

445

service:

446

# -- Enable controller services or not. This does not influence the creation of either the admission webhook or the metrics service.

447

enabled: true

448

external:

449

# -- Enable the external controller service or not. Useful for internal-only deployments.

450

enabled: true

451

# -- Annotations to be added to the external controller service. See `controller.service.internal.annotations` for annotations to be added to the internal controller service.

452

annotations: {}

453

# -- Labels to be added to both controller services.

454

labels: {}

455

# -- Type of the external controller service.

456

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types

457

type: LoadBalancer

458

# -- Pre-defined cluster internal IP address of the external controller service. Take care of collisions with existing services.

459

# This value is immutable. Set once, it can not be changed without deleting and re-creating the service.

460

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#choosing-your-own-ip-address

461

clusterIP: ""

462

# -- List of node IP addresses at which the external controller service is available.

463

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#external-ips

464

externalIPs: []

465

# -- Deprecated: Pre-defined IP address of the external controller service. Used by cloud providers to connect the resulting load balancer service to a pre-existing static IP.

466

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#loadbalancer

467

loadBalancerIP: ""

468

# -- Restrict access to the external controller service. Values must be CIDRs. Allows any source address by default.

469

loadBalancerSourceRanges: []

470

# -- Load balancer class of the external controller service. Used by cloud providers to select a load balancer implementation other than the cloud provider default.

471

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#load-balancer-class

472

loadBalancerClass: ""

473

# -- Enable node port allocation for the external controller service or not. Applies to type `LoadBalancer` only.

474

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#load-balancer-nodeport-allocation

475

# allocateLoadBalancerNodePorts: true

476

​

477

# -- External traffic policy of the external controller service. Set to "Local" to preserve source IP on providers supporting it.

478

# Ref: https://kubernetes.io/docs/tasks/access-application-cluster/create-external-load-balancer/#preserving-the-client-source-ip

479

externalTrafficPolicy: ""

480

# -- Session affinity of the external controller service. Must be either "None" or "ClientIP" if set. Defaults to "None".

481

# Ref: https://kubernetes.io/docs/reference/networking/virtual-ips/#session-affinity

482

sessionAffinity: ""

483

# -- Specifies the health check node port (numeric port number) for the external controller service.

484

# If not specified, the service controller allocates a port from your cluster's node port range.

485

# Ref: https://kubernetes.io/docs/tasks/access-application-cluster/create-external-load-balancer/#preserving-the-client-source-ip

486

# healthCheckNodePort: 0

487

​

488

# -- Represents the dual-stack capabilities of the external controller service. Possible values are SingleStack, PreferDualStack or RequireDualStack.

489

# Fields `ipFamilies` and `clusterIP` depend on the value of this field.

490

# Ref: https://kubernetes.io/docs/concepts/services-networking/dual-stack/#services

491

ipFamilyPolicy: SingleStack

492

# -- List of IP families (e.g. IPv4, IPv6) assigned to the external controller service. This field is usually assigned automatically based on cluster configuration and the `ipFamilyPolicy` field.

493

# Ref: https://kubernetes.io/docs/concepts/services-networking/dual-stack/#services

494

ipFamilies:

495

- IPv4

496

# -- Enable the HTTP listener on both controller services or not.

497

enableHttp: true

498

# -- Enable the HTTPS listener on both controller services or not.

499

enableHttps: true

500

ports:

501

# -- Port the external HTTP listener is published with.

502

http: 80

503

# -- Port the external HTTPS listener is published with.

504

https: 443

505

targetPorts:

506

# -- Port of the ingress controller the external HTTP listener is mapped to.

507

http: http

508

# -- Port of the ingress controller the external HTTPS listener is mapped to.

509

https: https

510

# -- Declare the app protocol of the external HTTP and HTTPS listeners or not. Supersedes provider-specific annotations for declaring the backend protocol.

511

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#application-protocol

512

appProtocol: true

513

nodePorts:

514

# -- Node port allocated for the external HTTP listener. If left empty, the service controller allocates one from the configured node port range.

515

http: ""

516

# -- Node port allocated for the external HTTPS listener. If left empty, the service controller allocates one from the configured node port range.

517

https: ""

518

# -- Node port mapping for external TCP listeners. If left empty, the service controller allocates them from the configured node port range.

519

# Example:

520

# tcp:

521

# 8080: 30080

522

tcp: {}

523

# -- Node port mapping for external UDP listeners. If left empty, the service controller allocates them from the configured node port range.

524

# Example:

525

# udp:

526

# 53: 30053

527

udp: {}

528

internal:

529

# -- Enable the internal controller service or not. Remember to configure `controller.service.internal.annotations` when enabling this.

530

enabled: false

531

# -- Annotations to be added to the internal controller service. Mandatory for the internal controller service to be created. Varies with the cloud service.

532

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#internal-load-balancer

533

annotations: {}

534

# -- Type of the internal controller service.

535

# Defaults to the value of `controller.service.type`.

536

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types

537

type: ""

538

# -- Pre-defined cluster internal IP address of the internal controller service. Take care of collisions with existing services.

539

# This value is immutable. Set once, it can not be changed without deleting and re-creating the service.

540

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#choosing-your-own-ip-address

541

clusterIP: ""

542

# -- List of node IP addresses at which the internal controller service is available.

543

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#external-ips

544

externalIPs: []

545

# -- Deprecated: Pre-defined IP address of the internal controller service. Used by cloud providers to connect the resulting load balancer service to a pre-existing static IP.

546

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#loadbalancer

547

loadBalancerIP: ""

548

# -- Restrict access to the internal controller service. Values must be CIDRs. Allows any source address by default.

549

loadBalancerSourceRanges: []

550

# -- Load balancer class of the internal controller service. Used by cloud providers to select a load balancer implementation other than the cloud provider default.

551

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#load-balancer-class

552

loadBalancerClass: ""

553

# -- Enable node port allocation for the internal controller service or not. Applies to type `LoadBalancer` only.

554

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#load-balancer-nodeport-allocation

555

# allocateLoadBalancerNodePorts: true

556

​

557

# -- External traffic policy of the internal controller service. Set to "Local" to preserve source IP on providers supporting it.

558

# Ref: https://kubernetes.io/docs/tasks/access-application-cluster/create-external-load-balancer/#preserving-the-client-source-ip

559

externalTrafficPolicy: ""

560

# -- Session affinity of the internal controller service. Must be either "None" or "ClientIP" if set. Defaults to "None".

561

# Ref: https://kubernetes.io/docs/reference/networking/virtual-ips/#session-affinity

562

sessionAffinity: ""

563

# -- Specifies the health check node port (numeric port number) for the internal controller service.

564

# If not specified, the service controller allocates a port from your cluster's node port range.

565

# Ref: https://kubernetes.io/docs/tasks/access-application-cluster/create-external-load-balancer/#preserving-the-client-source-ip

566

# healthCheckNodePort: 0

567

​

568

# -- Represents the dual-stack capabilities of the internal controller service. Possible values are SingleStack, PreferDualStack or RequireDualStack.

569

# Fields `ipFamilies` and `clusterIP` depend on the value of this field.

570

# Ref: https://kubernetes.io/docs/concepts/services-networking/dual-stack/#services

571

ipFamilyPolicy: SingleStack

572

# -- List of IP families (e.g. IPv4, IPv6) assigned to the internal controller service. This field is usually assigned automatically based on cluster configuration and the `ipFamilyPolicy` field.

573

# Ref: https://kubernetes.io/docs/concepts/services-networking/dual-stack/#services

574

ipFamilies:

575

- IPv4

576

ports: {}

577

# -- Port the internal HTTP listener is published with.

578

# Defaults to the value of `controller.service.ports.http`.

579

# http: 80

580

# -- Port the internal HTTPS listener is published with.

581

# Defaults to the value of `controller.service.ports.https`.

582

# https: 443

583

​

584

targetPorts: {}

585

# -- Port of the ingress controller the internal HTTP listener is mapped to.

586

# Defaults to the value of `controller.service.targetPorts.http`.

587

# http: http

588

# -- Port of the ingress controller the internal HTTPS listener is mapped to.

589

# Defaults to the value of `controller.service.targetPorts.https`.

590

# https: https

591

​

592

# -- Declare the app protocol of the internal HTTP and HTTPS listeners or not. Supersedes provider-specific annotations for declaring the backend protocol.

593

# Ref: https://kubernetes.io/docs/concepts/services-networking/service/#application-protocol

594

appProtocol: true

595

nodePorts:

596

# -- Node port allocated for the internal HTTP listener. If left empty, the service controller allocates one from the configured node port range.

597

http: ""

598

# -- Node port allocated for the internal HTTPS listener. If left empty, the service controller allocates one from the configured node port range.

599

https: ""

600

# -- Node port mapping for internal TCP listeners. If left empty, the service controller allocates them from the configured node port range.

601

# Example:

602

# tcp:

603

# 8080: 30080

604

tcp: {}

605

# -- Node port mapping for internal UDP listeners. If left empty, the service controller allocates them from the configured node port range.

606

# Example:

607

# udp:

608

# 53: 30053

609

udp: {}

610

# shareProcessNamespace enables process namespace sharing within the pod.

611

# This can be used for example to signal log rotation using `kill -USR1` from a sidecar.

612

shareProcessNamespace: false

613

# -- Additional containers to be added to the controller pod.

614

# See https://github.com/lemonldap-ng-controller/lemonldap-ng-controller as example.

615

extraContainers: []

616

# - name: my-sidecar

617

# image: nginx:latest

618

# - name: lemonldap-ng-controller

619

# image: lemonldapng/lemonldap-ng-controller:0.2.0

620

# args:

621

# - /lemonldap-ng-controller

622

# - --alsologtostderr

623

# - --configmap=$(POD\_NAMESPACE)/lemonldap-ng-configuration

624

# env:

625

# - name: POD\_NAME

626

# valueFrom:

627

# fieldRef:

628

# fieldPath: metadata.name

629

# - name: POD\_NAMESPACE

630

# valueFrom:

631

# fieldRef:

632

# fieldPath: metadata.namespace

633

# volumeMounts:

634

# - name: copy-portal-skins

635

# mountPath: /srv/var/lib/lemonldap-ng/portal/skins

636

​

637

# -- Additional volumeMounts to the controller main container.

638

extraVolumeMounts: []

639

# - name: copy-portal-skins

640

# mountPath: /var/lib/lemonldap-ng/portal/skins

641

​

642

# -- Additional volumes to the controller pod.

643

extraVolumes: []

644

# - name: copy-portal-skins

645

# emptyDir: {}

646

​

647

# -- Containers, which are run before the app containers are started.

648

extraInitContainers: []

649

# - name: init-myservice

650

# image: busybox

651

# command: ['sh', '-c', 'until nslookup myservice; do echo waiting for myservice; sleep 2; done;']

652

​

653

# -- Modules, which are mounted into the core nginx image. See values.yaml for a sample to add opentelemetry module

654

extraModules: []

655

# - name: mytestmodule

656

# image:

657

# registry: registry.k8s.io

658

# image: ingress-nginx/mytestmodule

659

# ## for backwards compatibility consider setting the full image url via the repository value below

660

# ## use \*either\* current default registry/image or repository format or installing chart by providing the values.yaml will fail

661

# ## repository:

662

# tag: "v1.0.0"

663

# digest: ""

664

# distroless: false

665

# containerSecurityContext:

666

# runAsNonRoot: true

667

# runAsUser: <user-id>

668

# allowPrivilegeEscalation: false

669

# seccompProfile:

670

# type: RuntimeDefault

671

# capabilities:

672

# drop:

673

# - ALL

674

# readOnlyRootFilesystem: true

675

# resources: {}

676

#

677

# The image must contain a `/usr/local/bin/init\_module.sh` executable, which

678

# will be executed as initContainers, to move its config files within the

679

# mounted volume.

680

​

681

opentelemetry:

682

enabled: false

683

name: opentelemetry

684

image:

685

registry: registry.k8s.io

686

image: ingress-nginx/opentelemetry

687

## for backwards compatibility consider setting the full image url via the repository value below

688

## use \*either\* current default registry/image or repository format or installing chart by providing the values.yaml will fail

689

## repository:

690

tag: "v20230721-3e2062ee5"

691

digest: sha256:13bee3f5223883d3ca62fee7309ad02d22ec00ff0d7033e3e9aca7a9f60fd472

692

distroless: true

693

containerSecurityContext:

694

runAsNonRoot: true

695

# -- The image's default user, inherited from its base image `cgr.dev/chainguard/static`.

696

runAsUser: 65532

697

allowPrivilegeEscalation: false

698

seccompProfile:

699

type: RuntimeDefault

700

capabilities:

701

drop:

702

- ALL

703

readOnlyRootFilesystem: true

704

resources: {}

705

admissionWebhooks:

706

name: admission

707

annotations: {}

708

# ignore-check.kube-linter.io/no-read-only-rootfs: "This deployment needs write access to root filesystem".

709

​

710

## Additional annotations to the admission webhooks.

711

## These annotations will be added to the ValidatingWebhookConfiguration and

712

## the Jobs Spec of the admission webhooks.

713

enabled: true

714

# -- Additional environment variables to set

715

extraEnvs: []

716

# extraEnvs:

717

# - name: FOO

718

# valueFrom:

719

# secretKeyRef:

720

# key: FOO

721

# name: secret-resource

722

# -- Admission Webhook failure policy to use

723

failurePolicy: Fail

724

# timeoutSeconds: 10

725

port: 8443

726

certificate: "/usr/local/certificates/cert"

727

key: "/usr/local/certificates/key"

728

namespaceSelector: {}

729

objectSelector: {}

730

# -- Labels to be added to admission webhooks

731

labels: {}

732

# -- Use an existing PSP instead of creating one

733

existingPsp: ""

734

service:

735

annotations: {}

736

# clusterIP: ""

737

externalIPs: []

738

# loadBalancerIP: ""

739

loadBalancerSourceRanges: []

740

servicePort: 443

741

type: ClusterIP

742

createSecretJob:

743

name: create

744

# -- Security context for secret creation containers

745

securityContext:

746

runAsNonRoot: true

747

runAsUser: 65532

748

allowPrivilegeEscalation: false

749

seccompProfile:

750

type: RuntimeDefault

751

capabilities:

752

drop:

753

- ALL

754

readOnlyRootFilesystem: true

755

resources: {}

756

# limits:

757

# cpu: 10m

758

# memory: 20Mi

759

# requests:

760

# cpu: 10m

761

# memory: 20Mi

762

patchWebhookJob:

763

name: patch

764

# -- Security context for webhook patch containers

765

securityContext:

766

runAsNonRoot: true

767

runAsUser: 65532

768

allowPrivilegeEscalation: false

769

seccompProfile:

770

type: RuntimeDefault

771

capabilities:

772

drop:

773

- ALL

774

readOnlyRootFilesystem: true

775

resources: {}

776

patch:

777

enabled: true

778

image:

779

registry: registry.k8s.io

780

image: ingress-nginx/kube-webhook-certgen

781

## for backwards compatibility consider setting the full image url via the repository value below

782

## use \*either\* current default registry/image or repository format or installing chart by providing the values.yaml will fail

783

## repository:

784

tag: v20231226-1a7112e06

785

digest: sha256:25d6a5f11211cc5c3f9f2bf552b585374af287b4debf693cacbe2da47daa5084

786

pullPolicy: IfNotPresent

787

# -- Provide a priority class name to the webhook patching job

788

##

789

priorityClassName: ""

790

podAnnotations: {}

791

# NetworkPolicy for webhook patch

792

networkPolicy:

793

# -- Enable 'networkPolicy' or not

794

enabled: false

795

nodeSelector:

796

kubernetes.io/os: linux

797

tolerations: []

798

# -- Labels to be added to patch job resources

799

labels: {}

800

# -- Security context for secret creation & webhook patch pods

801

securityContext: {}

802

# Use certmanager to generate webhook certs

803

certManager:

804

enabled: false

805

# self-signed root certificate

806

rootCert:

807

# default to be 5y

808

duration: ""

809

admissionCert:

810

# default to be 1y

811

duration: ""

812

# issuerRef:

813

# name: "issuer"

814

# kind: "ClusterIssuer"

815

metrics:

816

port: 10254

817

portName: metrics

818

# if this port is changed, change healthz-port: in extraArgs: accordingly

819

enabled: false

820

service:

821

annotations: {}

822

# prometheus.io/scrape: "true"

823

# prometheus.io/port: "10254"

824

# -- Labels to be added to the metrics service resource

825

labels: {}

826

# clusterIP: ""

827

​

828

# -- List of IP addresses at which the stats-exporter service is available

829

## Ref: https://kubernetes.io/docs/concepts/services-networking/service/#external-ips

830

##

831

externalIPs: []

832

# loadBalancerIP: ""

833

loadBalancerSourceRanges: []

834

servicePort: 10254

835

type: ClusterIP

836

# externalTrafficPolicy: ""

837

# nodePort: ""

838

serviceMonitor:

839

enabled: false

840

additionalLabels: {}

841

annotations: {}

842

## The label to use to retrieve the job name from.

843

## jobLabel: "app.kubernetes.io/name"

844

namespace: ""

845

namespaceSelector: {}

846

## Default: scrape .Release.Namespace or namespaceOverride only

847

## To scrape all, use the following:

848

## namespaceSelector:

849

## any: true

850

scrapeInterval: 30s

851

# honorLabels: true

852

targetLabels: []

853

relabelings: []

854

metricRelabelings: []

855

prometheusRule:

856

enabled: false

857

additionalLabels: {}

858

# namespace: ""

859

rules: []

860

# # These are just examples rules, please adapt them to your needs

861

# - alert: NGINXConfigFailed

862

# expr: count(nginx\_ingress\_controller\_config\_last\_reload\_successful == 0) > 0

863

# for: 1s

864

# labels:

865

# severity: critical

866

# annotations:

867

# description: bad ingress config - nginx config test failed

868

# summary: uninstall the latest ingress changes to allow config reloads to resume

869

# # By default a fake self-signed certificate is generated as default and

870

# # it is fine if it expires. If `--default-ssl-certificate` flag is used

871

# # and a valid certificate passed please do not filter for `host` label!

872

# # (i.e. delete `{host!="\_"}` so also the default SSL certificate is

873

# # checked for expiration)

874

# - alert: NGINXCertificateExpiry

875

# expr: (avg(nginx\_ingress\_controller\_ssl\_expire\_time\_seconds{host!="\_"}) by (host) - time()) < 604800

876

# for: 1s

877

# labels:

878

# severity: critical

879

# annotations:

880

# description: ssl certificate(s) will expire in less then a week

881

# summary: renew expiring certificates to avoid downtime

882

# - alert: NGINXTooMany500s

883

# expr: 100 \* ( sum( nginx\_ingress\_controller\_requests{status=~"5.+"} ) / sum(nginx\_ingress\_controller\_requests) ) > 5

884

# for: 1m

885

# labels:

886

# severity: warning

887

# annotations:

888

# description: Too many 5XXs

889

# summary: More than 5% of all requests returned 5XX, this requires your attention

890

# - alert: NGINXTooMany400s

891

# expr: 100 \* ( sum( nginx\_ingress\_controller\_requests{status=~"4.+"} ) / sum(nginx\_ingress\_controller\_requests) ) > 5

892

# for: 1m

893

# labels:

894

# severity: warning

895

# annotations:

896

# description: Too many 4XXs

897

# summary: More than 5% of all requests returned 4XX, this requires your attention

898

# -- Improve connection draining when ingress controller pod is deleted using a lifecycle hook:

899

# With this new hook, we increased the default terminationGracePeriodSeconds from 30 seconds

900

# to 300, allowing the draining of connections up to five minutes.

901

# If the active connections end before that, the pod will terminate gracefully at that time.

902

# To effectively take advantage of this feature, the Configmap feature

903

# worker-shutdown-timeout new value is 240s instead of 10s.

904

##

905

lifecycle:

906

preStop:

907

exec:

908

command:

909

- /wait-shutdown

910

priorityClassName: ""

911

# -- Rollback limit

912

##

913

revisionHistoryLimit: 10

914

## Default 404 backend

915

##

916

defaultBackend:

917

##

918

enabled: false

919

name: defaultbackend

920

image:

921

registry: registry.k8s.io

922

image: defaultbackend-amd64

923

## for backwards compatibility consider setting the full image url via the repository value below

924

## use \*either\* current default registry/image or repository format or installing chart by providing the values.yaml will fail

925

## repository:

926

tag: "1.5"

927

pullPolicy: IfNotPresent

928

runAsNonRoot: true

929

# nobody user -> uid 65534

930

runAsUser: 65534

931

allowPrivilegeEscalation: false

932

seccompProfile:

933

type: RuntimeDefault

934

readOnlyRootFilesystem: true

935

# -- Use an existing PSP instead of creating one

936

existingPsp: ""

937

extraArgs: {}

938

serviceAccount:

939

create: true

940

name: ""

941

automountServiceAccountToken: true

942

# -- Additional environment variables to set for defaultBackend pods

943

extraEnvs: []

944

port: 8080

945

## Readiness and liveness probes for default backend

946

## Ref: https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-probes/

947

##

948

livenessProbe:

949

failureThreshold: 3

950

initialDelaySeconds: 30

951

periodSeconds: 10

952

successThreshold: 1

953

timeoutSeconds: 5

954

readinessProbe:

955

failureThreshold: 6

956

initialDelaySeconds: 0

957

periodSeconds: 5

958

successThreshold: 1

959

timeoutSeconds: 5

960

# -- The update strategy to apply to the Deployment or DaemonSet

961

##

962

updateStrategy: {}

963

# rollingUpdate:

964

# maxUnavailable: 1

965

# type: RollingUpdate

966

​

967

# -- `minReadySeconds` to avoid killing pods before we are ready

968

##

969

minReadySeconds: 0

970

# -- Node tolerations for server scheduling to nodes with taints

971

## Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-node/

972

##

973

tolerations: []

974

# - key: "key"

975

# operator: "Equal|Exists"

976

# value: "value"

977

# effect: "NoSchedule|PreferNoSchedule|NoExecute(1.6 only)"

978

​

979

affinity: {}

980

# -- Security context for default backend pods

981

podSecurityContext: {}

982

# -- Security context for default backend containers

983

containerSecurityContext: {}

984

# -- Labels to add to the pod container metadata

985

podLabels: {}

986

# key: value

987

​

988

# -- Node labels for default backend pod assignment

989

## Ref: https://kubernetes.io/docs/concepts/scheduling-eviction/assign-pod-node/

990

##

991

nodeSelector:

992

kubernetes.io/os: linux

993

# -- Annotations to be added to default backend pods

994

##

995

podAnnotations: {}

996

replicaCount: 1

997

minAvailable: 1

998

resources: {}

999

# limits:

1000

# cpu: 10m

1001

# memory: 20Mi

1002

# requests:

1003

# cpu: 10m

1004

# memory: 20Mi

1005

​

1006

extraVolumeMounts: []

1007

## Additional volumeMounts to the default backend container.

1008

# - name: copy-portal-skins

1009

# mountPath: /var/lib/lemonldap-ng/portal/skins

1010

​

1011

extraVolumes: []

1012

## Additional volumes to the default backend pod.

1013

# - name: copy-portal-skins

1014

# emptyDir: {}

1015

​

1016

extraConfigMaps: []

1017

## Additional configmaps to the default backend pod.

1018

# - name: my-extra-configmap-1

1019

# labels:

1020

# type: config-1

1021

# data:

1022

# extra\_file\_1.html: |

1023

# <!-- Extra HTML content for ConfigMap 1 -->

1024

# - name: my-extra-configmap-2

1025

# labels:

1026

# type: config-2

1027

# data:

1028

# extra\_file\_2.html: |

1029

# <!-- Extra HTML content for ConfigMap 2 -->

1030

​

1031

autoscaling:

1032

annotations: {}

1033

enabled: false

1034

minReplicas: 1

1035

maxReplicas: 2

1036

targetCPUUtilizationPercentage: 50

1037

targetMemoryUtilizationPercentage: 50

1038

# NetworkPolicy for default backend component.

1039

networkPolicy:

1040

# -- Enable 'networkPolicy' or not

1041

enabled: false

1042

service:

1043

annotations: {}

1044

# clusterIP: ""

1045

​

1046

# -- List of IP addresses at which the default backend service is available

1047

## Ref: https://kubernetes.io/docs/concepts/services-networking/service/#external-ips

1048

##

1049

externalIPs: []

1050

# loadBalancerIP: ""

1051

loadBalancerSourceRanges: []

1052

servicePort: 80

1053

type: ClusterIP

1054

priorityClassName: ""

1055

# -- Labels to be added to the default backend resources

1056

labels: {}

1057

## Enable RBAC as per https://github.com/kubernetes/ingress-nginx/blob/main/docs/deploy/rbac.md and https://github.com/kubernetes/ingress-nginx/issues/266

1058

rbac:

1059

create: true

1060

scope: false

1061

## If true, create & use Pod Security Policy resources

1062

## https://kubernetes.io/docs/concepts/policy/pod-security-policy/

1063

podSecurityPolicy:

1064

enabled: false

1065

serviceAccount:

1066

create: true

1067

name: ""

1068

automountServiceAccountToken: true

1069

# -- Annotations for the controller service account

1070

annotations: {}

1071

# -- Optional array of imagePullSecrets containing private registry credentials

1072

## Ref: https://kubernetes.io/docs/tasks/configure-pod-container/pull-image-private-registry/

1073

imagePullSecrets: []

1074

# - name: secretName

1075

​

1076

# -- TCP service key-value pairs

1077

## Ref: https://github.com/kubernetes/ingress-nginx/blob/main/docs/user-guide/exposing-tcp-udp-services.md

1078

##

1079

tcp: {}

1080

# 8080: "default/example-tcp-svc:9000"

1081

​

1082

# -- UDP service key-value pairs

1083

## Ref: https://github.com/kubernetes/ingress-nginx/blob/main/docs/user-guide/exposing-tcp-udp-services.md

1084

##

1085

udp: {}

1086

# 53: "kube-system/kube-dns:53"

1087

​

1088

# -- Prefix for TCP and UDP ports names in ingress controller service

1089

## Some cloud providers, like Yandex Cloud may have a requirements for a port name regex to support cloud load balancer integration

1090

portNamePrefix: ""

1091

# -- (string) A base64-encoded Diffie-Hellman parameter.

1092

# This can be generated with: `openssl dhparam 4096 2> /dev/null | base64`

1093

## Ref: https://github.com/kubernetes/ingress-nginx/tree/main/docs/examples/customization/ssl-dh-param

1094

dhParam: ""

Ở file values trên  có vài tham số đã được chỉnh sửa như

1

kind: DaemonSet

chạy ingress controller trên tất cả các node worker

1

  hostPort:

2

    enabled: true

mở port 80,443 cho các ingress controller chạy trên các  node có thể tiếp nhân request từ LB

**- Cài đăt Ingress Nginx với version cụ thể**

1

helm install ingress-nginx ingress-nginx/ingress-nginx --version 4.9.1 --values=ingress-values.yaml --namespace ingress-nginx --create-namespace

Ở lệnh trên helm chart version củaI ngress là 4.9.0 tương ứng với Nginx version 1.9.5 và lấy giá trị các tham số từ file ingress-values.yaml, Ingress Controller  sẽ được cài đặt trên namespace ingress-nginx

**- Để tìm version Nginx để chỉ định có thể dùng lệnh sau**

1

helm search repo ingress-nginx -l

**- Gỡ cài đăt Ingress Nginx ta dùng lênh sau**

1

helm uninstall ingress-nginx -n ingress-nginx

**- Dưới đây là ví dụ mẫu về cách tạo 1 Ingress Nginx**

1

apiVersion: networking.k8s.io/v1

2

kind: Ingress

3

metadata:

4

annotations:

5

name: test-ingress

6

spec:

7

ingressClassName: nginx

8

rules:

9

- host: test.example.com

10

http:

11

paths:

12

- backend:

13

service:

14

name: test-svc

15

port:

16

number: 80

17

path: /

18

pathType: Prefix

19

tls:

20

- hosts:

21

- test.example.com

22

secretName: test-tls