<https://kubesphere.com.cn/docs/quick-start/minimal-kubesphere-on-k8s/>

**0 查看安装默认版本**

<https://kubesphere.io/zh/docs/quick-start/all-in-one-on-linux/>

安装 KubeSphere 3.2.1 的建议 Kubernetes 版本：1.19.x、1.20.x、1.21.x 或 1.22.x（实验性支持）。如果不指定 Kubernetes 版本，KubeKey 将默认安装 Kubernetes v1.21.5。有关受支持的 Kubernetes 版本的更多信息，请参见[支持矩阵](https://kubesphere.io/zh/docs/installing-on-linux/introduction/kubekey/" \l "%E6%94%AF%E6%8C%81%E7%9F%A9%E9%98%B5)。



**1下载**

# wget <https://github.com/kubesphere/ks-installer/releases/download/v3.2.1/cluster-configuration.yaml>

# wget <https://github.com/kubesphere/ks-installer/releases/download/v3.2.1/kubesphere-installer.yaml>

kubesphere-installer.yaml 不需要修该

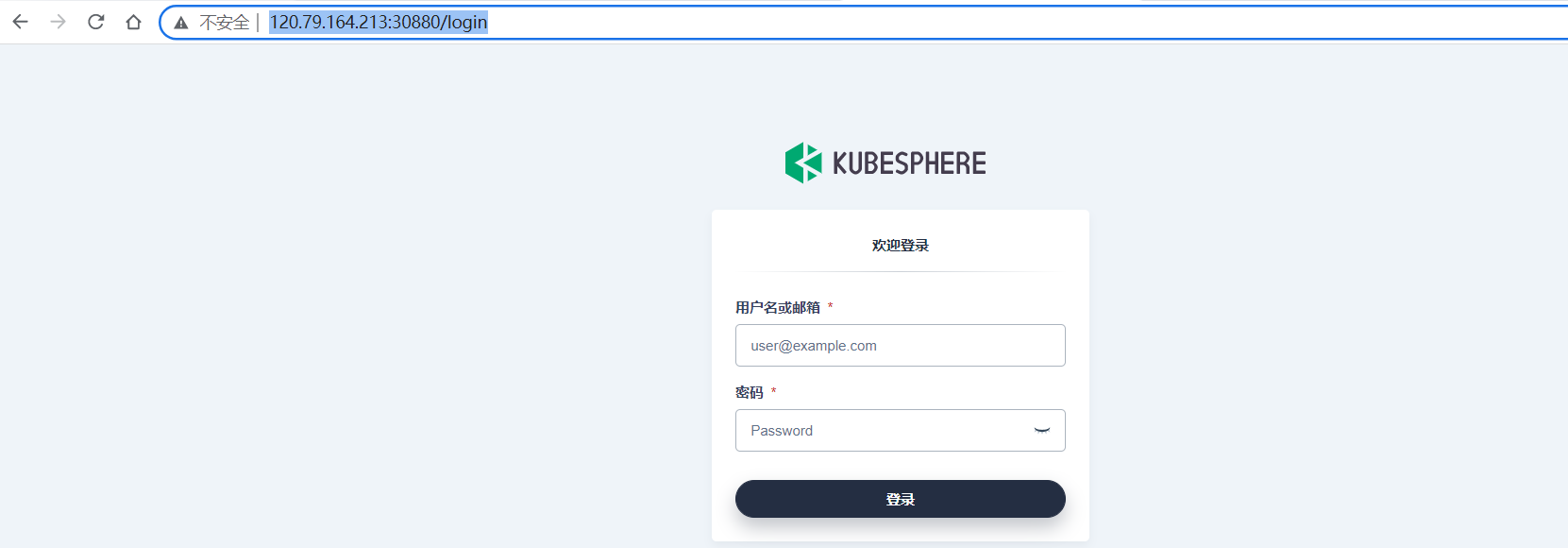
vim cluster-configuration.yaml (开启组件功能)**（也可以默认先最小化安装，不修改）**

|  |
| --- |
| ---  apiVersion: installer.kubesphere.io/v1alpha1  kind: ClusterConfiguration  metadata:  name: ks-installer  namespace: kubesphere-system  labels:  version: v3.2.1  spec:  persistence:  storageClass: "" # If there is no default StorageClass in your cluster, you need to specify an existing StorageClass here.  authentication:  jwtSecret: "" # Keep the jwtSecret consistent with the Host Cluster. Retrieve the jwtSecret by executing "kubectl -n kubesphere-system get cm kubesphere-config -o yaml | grep -v "apiVersion" | grep jwtSecret" on the Host Cluster.  local\_registry: "" # Add your private registry address if it is needed.  # dev\_tag: "" # Add your kubesphere image tag you want to install, by default it's same as ks-install release version.  etcd:  #monitoring: false # Enable or disable etcd monitoring dashboard installation. You have to create a Secret for etcd before you enable it.  monitoring: true # Enable or disable etcd monitoring dashboard installation. You have to create a Secret for etcd before you enable it.  #endpointIps: localhost # etcd cluster EndpointIps. It can be a bunch of IPs here.  endpointIps: 172.28.10.18 # etcd cluster EndpointIps. It can be a bunch of IPs here.  port: 2379 # etcd port.  tlsEnable: true  common:  core:  console:  enableMultiLogin: true # Enable or disable simultaneous logins. It allows different users to log in with the same account at the same time.  port: 30880  type: NodePort  # apiserver: # Enlarge the apiserver and controller manager's resource requests and limits for the large cluster  # resources: {}  # controllerManager:  # resources: {}  redis:  #enabled: false  enabled: true  volumeSize: 2Gi # Redis PVC size.  openldap:  #enabled: false  enabled: true  volumeSize: 2Gi # openldap PVC size.  minio:  volumeSize: 20Gi # Minio PVC size.  monitoring:  # type: external # Whether to specify the external prometheus stack, and need to modify the endpoint at the next line.  endpoint: http://prometheus-operated.kubesphere-monitoring-system.svc:9090 # Prometheus endpoint to get metrics data.  GPUMonitoring: # Enable or disable the GPU-related metrics. If you enable this switch but have no GPU resources, Kubesphere will set it to zero.  enabled: false  gpu: # Install GPUKinds. The default GPU kind is nvidia.com/gpu. Other GPU kinds can be added here according to your needs.  kinds:  - resourceName: "nvidia.com/gpu"  resourceType: "GPU"  default: true  es: # Storage backend for logging, events and auditing.  # master:  # volumeSize: 4Gi # The volume size of Elasticsearch master nodes.  # replicas: 1 # The total number of master nodes. Even numbers are not allowed.  # resources: {}  # data:  # volumeSize: 20Gi # The volume size of Elasticsearch data nodes.  # replicas: 1 # The total number of data nodes.  # resources: {}  logMaxAge: 7 # Log retention time in built-in Elasticsearch. It is 7 days by default.  elkPrefix: logstash # The string making up index names. The index name will be formatted as ks-<elk\_prefix>-log.  basicAuth:  enabled: false  username: ""  password: ""  externalElasticsearchUrl: ""  externalElasticsearchPort: ""  alerting: # (CPU: 0.1 Core, Memory: 100 MiB) It enables users to customize alerting policies to send messages to receivers in time with different time intervals and alerting levels to choose from.  #enabled: false # Enable or disable the KubeSphere Alerting System.  enabled: true # Enable or disable the KubeSphere Alerting System.  # thanosruler:  # replicas: 1  # resources: {}  auditing: # Provide a security-relevant chronological set of records，recording the sequence of activities happening on the platform, initiated by different tenants.  #enabled: false # Enable or disable the KubeSphere Auditing Log System.  enabled: true # Enable or disable the KubeSphere Auditing Log System.  # operator:  # resources: {}  # webhook:  # resources: {}  devops: # (CPU: 0.47 Core, Memory: 8.6 G) Provide an out-of-the-box CI/CD system based on Jenkins, and automated workflow tools including Source-to-Image & Binary-to-Image.  #enabled: false # Enable or disable the KubeSphere DevOps System.  enabled: true # Enable or disable the KubeSphere DevOps System.  # resources: {}  jenkinsMemoryLim: 2Gi # Jenkins memory limit.  jenkinsMemoryReq: 1500Mi # Jenkins memory request.  jenkinsVolumeSize: 8Gi # Jenkins volume size.  jenkinsJavaOpts\_Xms: 512m # The following three fields are JVM parameters.  jenkinsJavaOpts\_Xmx: 512m  jenkinsJavaOpts\_MaxRAM: 2g  events: # Provide a graphical web console for Kubernetes Events exporting, filtering and alerting in multi-tenant Kubernetes clusters.  #enabled: false # Enable or disable the KubeSphere Events System.  enabled: true # Enable or disable the KubeSphere Events System.  # operator:  # resources: {}  # exporter:  # resources: {}  # ruler:  # enabled: true  # replicas: 2  # resources: {}  logging: # (CPU: 57 m, Memory: 2.76 G) Flexible logging functions are provided for log query, collection and management in a unified console. Additional log collectors can be added, such as Elasticsearch, Kafka and Fluentd.  #enabled: false # Enable or disable the KubeSphere Logging System.  enabled: true # Enable or disable the KubeSphere Logging System.  containerruntime: docker  logsidecar:  enabled: true  replicas: 2  # resources: {}  metrics\_server: # (CPU: 56 m, Memory: 44.35 MiB) It enables HPA (Horizontal Pod Autoscaler).  enabled: false # Enable or disable metrics-server.  monitoring:  storageClass: "" # If there is an independent StorageClass you need for Prometheus, you can specify it here. The default StorageClass is used by default.  # kube\_rbac\_proxy:  # resources: {}  # kube\_state\_metrics:  # resources: {}  # prometheus:  # replicas: 1 # Prometheus replicas are responsible for monitoring different segments of data source and providing high availability.  # volumeSize: 20Gi # Prometheus PVC size.  # resources: {}  # operator:  # resources: {}  # adapter:  # resources: {}  # node\_exporter:  # resources: {}  # alertmanager:  # replicas: 1 # AlertManager Replicas.  # resources: {}  # notification\_manager:  # resources: {}  # operator:  # resources: {}  # proxy:  # resources: {}  gpu: # GPU monitoring-related plug-in installation.  nvidia\_dcgm\_exporter: # Ensure that gpu resources on your hosts can be used normally, otherwise this plug-in will not work properly.  enabled: false # Check whether the labels on the GPU hosts contain "nvidia.com/gpu.present=true" to ensure that the DCGM pod is scheduled to these nodes.  # resources: {}  multicluster:  clusterRole: none # host | member | none # You can install a solo cluster, or specify it as the Host or Member Cluster.  network:  networkpolicy: # Network policies allow network isolation within the same cluster, which means firewalls can be set up between certain instances (Pods).  # Make sure that the CNI network plugin used by the cluster supports NetworkPolicy. There are a number of CNI network plugins that support NetworkPolicy, including Calico, Cilium, Kube-router, Romana and Weave Net.  #enabled: false # Enable or disable network policies.  enabled: true # Enable or disable network policies.  ippool: # Use Pod IP Pools to manage the Pod network address space. Pods to be created can be assigned IP addresses from a Pod IP Pool.  #type: none # Specify "calico" for this field if Calico is used as your CNI plugin. "none" means that Pod IP Pools are disabled.  type: calico # Specify "calico" for this field if Calico is used as your CNI plugin. "none" means that Pod IP Pools are disabled.  topology: # Use Service Topology to view Service-to-Service communication based on Weave Scope.  type: none # Specify "weave-scope" for this field to enable Service Topology. "none" means that Service Topology is disabled.  openpitrix: # An App Store that is accessible to all platform tenants. You can use it to manage apps across their entire lifecycle.  store:  #enabled: false # Enable or disable the KubeSphere App Store.  enabled: true # Enable or disable the KubeSphere App Store.  servicemesh: # (0.3 Core, 300 MiB) Provide fine-grained traffic management, observability and tracing, and visualized traffic topology.  #enabled: false # Base component (pilot). Enable or disable KubeSphere Service Mesh (Istio-based).  enabled: true # Base component (pilot). Enable or disable KubeSphere Service Mesh (Istio-based).  kubeedge: # Add edge nodes to your cluster and deploy workloads on edge nodes.  #enabled: false # Enable or disable KubeEdge.  enabled: true # Enable or disable KubeEdge.（可不开启）  cloudCore:  nodeSelector: {"node-role.kubernetes.io/worker": ""}  tolerations: []  cloudhubPort: "10000"  cloudhubQuicPort: "10001"  cloudhubHttpsPort: "10002"  cloudstreamPort: "10003"  tunnelPort: "10004"  cloudHub:  advertiseAddress: # At least a public IP address or an IP address which can be accessed by edge nodes must be provided.  - "" # Note that once KubeEdge is enabled, CloudCore will malfunction if the address is not provided.  nodeLimit: "100"  service:  cloudhubNodePort: "30000"  cloudhubQuicNodePort: "30001"  cloudhubHttpsNodePort: "30002"  cloudstreamNodePort: "30003"  tunnelNodePort: "30004"  edgeWatcher:  nodeSelector: {"node-role.kubernetes.io/worker": ""}  tolerations: []  edgeWatcherAgent:  nodeSelector: {"node-role.kubernetes.io/worker": ""}  tolerations: [] |

**检查安装日志**

|  |
| --- |
| kubectl logs -n kubesphere-system $(kubectl get pod -n kubesphere-system -l app=ks-install -o jsonpath='{.items[0].metadata.name}') -f |

<http://120.79.164.213:30880/login>



Console: http://172.28.10.18:30880

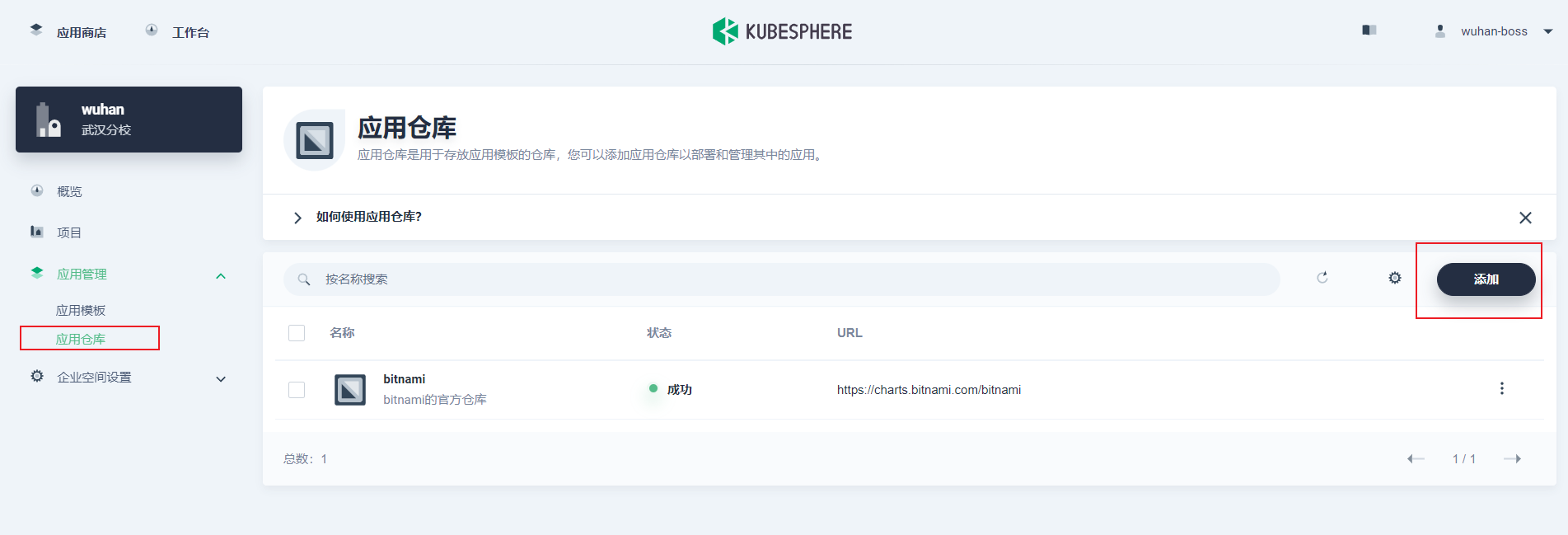
Account: admin

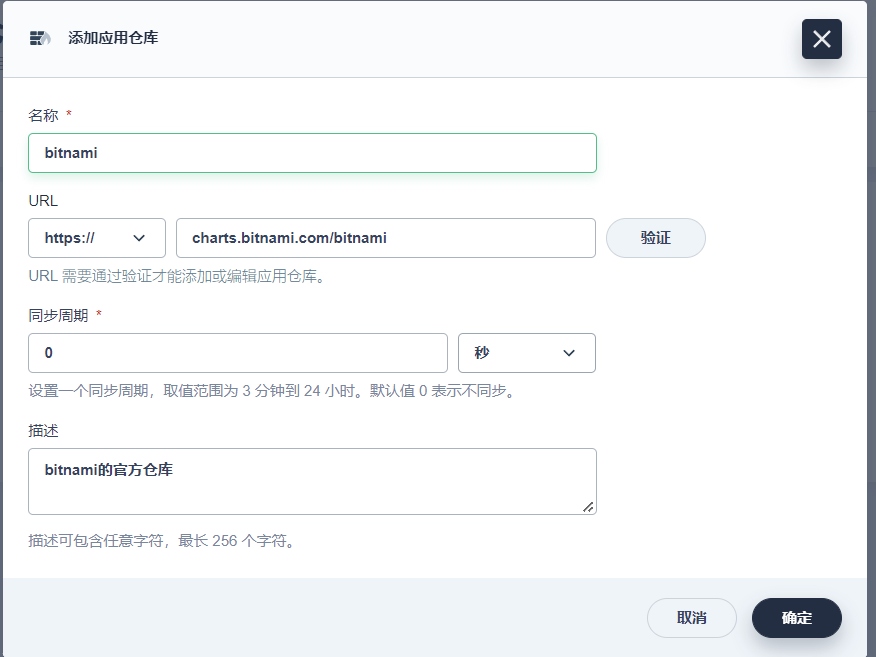
Password: P@88w0rd

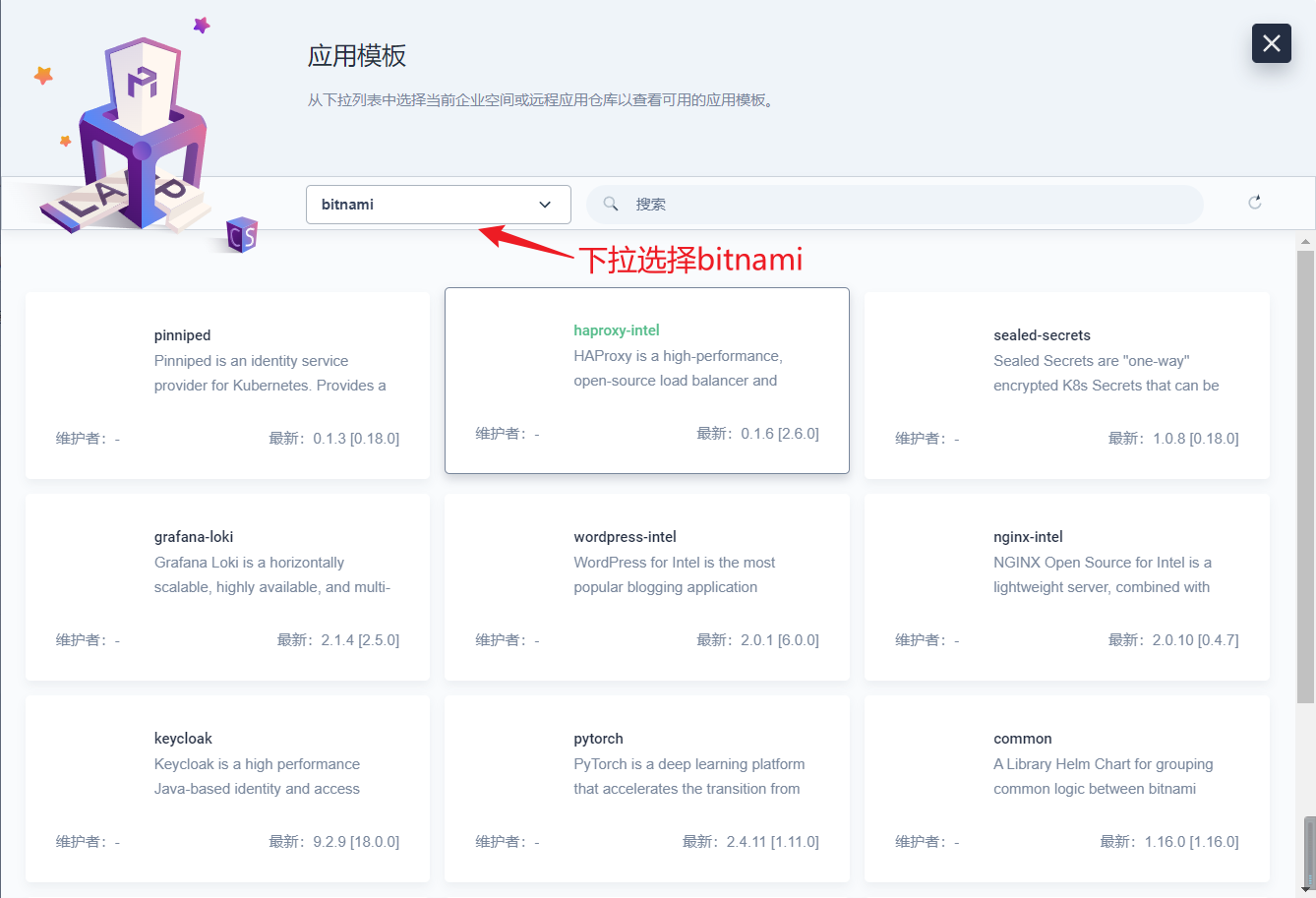
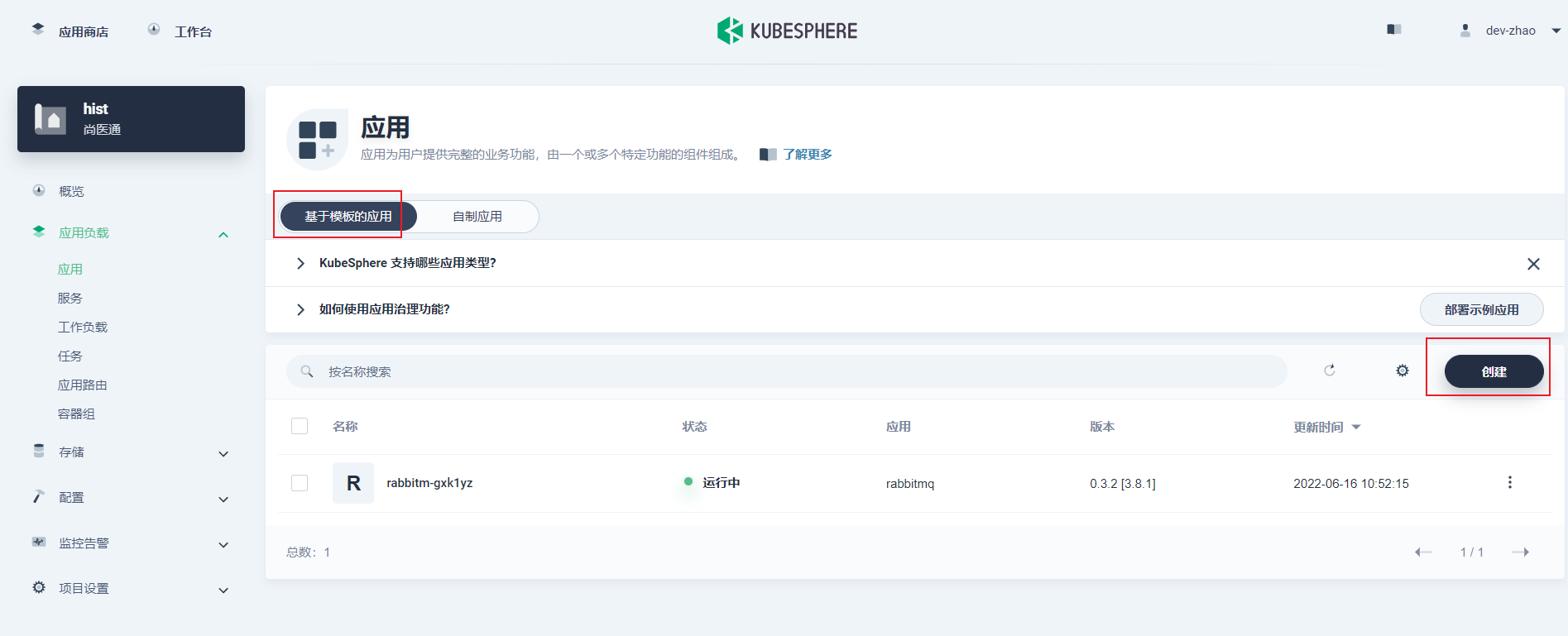
**在kubesphere平台开启组件**

<https://kubesphere.com.cn/docs/pluggable-components/>

添加模版仓库







**卸载kubesphere**

<https://kubesphere.com.cn/docs/installing-on-kubernetes/uninstall-kubesphere-from-k8s/>

wget https://raw.githubusercontent.com/kubesphere/ks-installer/release-3.1/scripts/kubesphere-delete.sh