

Kubernetes mit kubeadm installieren

GUUG-Frühjahrsfachgespräch 2019

11. April 2019

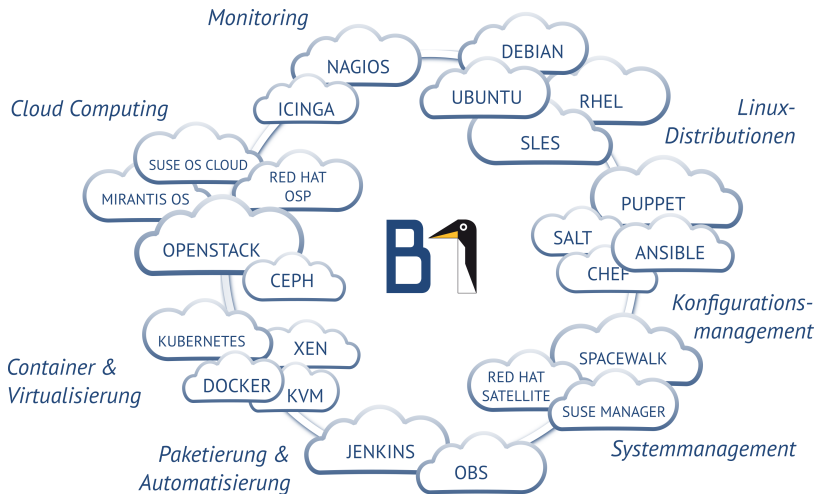


Christian Rost
Linux Consultant & Trainer
B1 Systems GmbH
rost@b1-systems.de

Vorstellung B1 Systems

- gegründet 2004
- primär Linux/Open Source-Themen
- national & international tätig
- über 100 Mitarbeiter
- unabhängig von Soft- und Hardware-Herstellern
- Leistungsangebot:
 - Beratung & Consulting
 - Support
 - Entwicklung
 - Training
 - Betrieb
 - Lösungen
- Standorte in Rockolding, Köln, Berlin & Dresden

Schwerpunkte



Kubernetes

Kubernetes

Open Source Plattform zur Automatisierung von

- Bereitstellung,
- Skalierung und
- Betrieb

von Anwendungscontainern

Komponenten

Kubernetes Cluster

Master (Controlplane)

kube-apiserver

kube-controller-manager

kube-scheduler

etcd

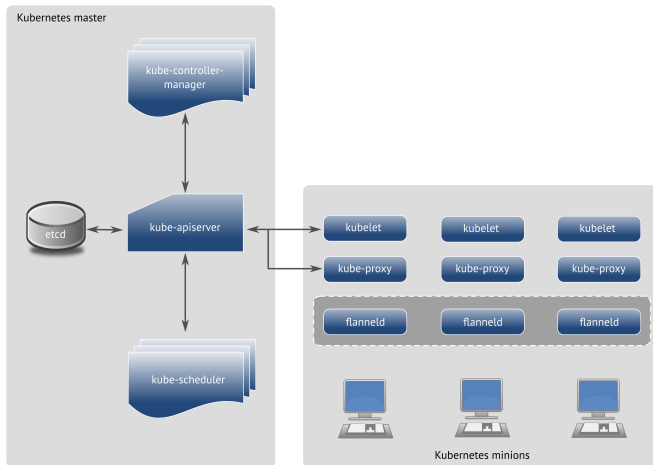
Worker

kubelet

kube-proxy

Container Runtime

Schematischer Aufbau eines Kubernetes Clusters



source: <https://kubernetes.io/docs/concepts/architecture/cloud-controller/>

Kubeadm

- Kubernetes Cluster erstellen
- Single-/Multi-Master
- Master-Komponenten als *Static Pod*

Kubeadm – einzelne Phasen

- 1 preflight
- 2 kubelet-start
- 3 certs
- 4 kubeconfig
- 5 control-plane
- 6 etcd
- 7 upload-config
- 8 upload-certs
- 9 mark-control-plane
- 10 bootstrap-token
- 11 addon

Installation

- ① Pakete installieren:
 - Container Runtime, z. B. Docker
 - kubelet
 - kubeadm
 - kubectl
- ② Load Balancer installieren
- ③ Cluster initialisieren
- ④ Netzwerk-Plugin ausrollen; hier: Flannel
- ⑤ Nodes hinzufügen

Repository einbinden

Repository einbinden

```
# cat <<EOF > /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/\
    repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/\
    yum-key.gpg https://packages.cloud.google.com/yum/\
    doc/rpm-package-key.gpg
exclude=kube*
EOF
```

Pakete installieren und OS anpassen

Pakete installieren und OS anpassen

```
# cat <<EOF > /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
# sysctl --system
# setenforce 0
# sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/'\
/etc/selinux/config
# yum install -y kubelet kubeadm kubectl docker \
--disableexcludes=kubernetes
# systemctl enable kubelet && systemctl start kubelet
# systemctl enable docker && systemctl start docker
```

kubeadm Konfig erzeugen und anpassen

Pakete installieren und OS anpassen

```
# kubeadm config print init-defaults > kubeadm.yml
[...]  
    key: node-role.kubernetes.io/master  
---  
apiServer:  
  timeoutForControlPlane: 4m0s  
  extraArgs:  
    bind-address: "ipaddr_master1"  
    advertise-address: "ipaddr_VIP"  
    secure-port: "6443"  
apiVersion: kubeadm.k8s.io/v1beta1  
[...]
```

Static Pod für keepalived und haproxy

- Static Pod Config
 - `https://raw.githubusercontent.com/grischdian/ffg-2019-guug/master/kubeadm-multi-master/ka-hp.yaml`
 - unter `/etc/kubernetes/manifests` ablegen
- Wird von kubelet automatisch gestartet

kubeadm Master provisionieren 1/3

k8s-master-01 initialisieren/Zertifikate auf andere Master kopieren

```
# kubeadm init --config kubeadm.yml
# tar czf k8s-tls.tar \  
  /etc/kubernetes/admin.conf \  
  /etc/kubernetes/pki/*ca* \  
  /etc/kubernetes/pki/etcd/ca* \  
  /etc/kubernetes/pki/sa* \  
  /etc/kubernetes/manifests/ka-hp.yaml
# scp k8s-tls.tar k8s-master-02
# scp k8s-tls.tar k8s-master-03
```

kubeadm Master provisionieren 2/3

kubeadm Config ändern

```
# kubeadm config view > $file  
# kubeadm config upload from-file --config $file
```


kubeadm Master provisionieren 3/3

Andere Master joinen

```
# kubeadm join VIP:VIP_PORT --token <token> \  
--discovery-token-ca-cert-hash sha256:<hash> \  
--experimental-control-plane \  
--ignore-preflight-errors=DirAvailable--etc-kubernetes-manifests
```

Netzwerk-Plugin einrichten

Flannel installieren

```
# curl -OL https://raw.githubusercontent.com/coreos/flannel/  
/a70459be0084506e4ec919aa1c114638878db11b/  
Documentation/kube-flannel.yml  
# kubectl apply -f kube-flannel.yaml
```

Worker joinen

Worker joinen

```
# kubeadm join VIP:VIP_PORT --token <token> \  
--discovery-token-ca-cert-hash sha256:<hash> \
```



Demotime

Änderungen am Cluster

Änderungen am Cluster durchführen

```
# kubeadm config upload from-file --config $file  
# kubeadm init phase control-plane \  
    all|apiserver|controller-manager|scheduler
```

Zertifikat erneuern

- Zertifikate 1 Jahr gültig
- Selfsigned
- Renew manuell

Zertifikate erneuern

```
# kubeadm alpha certs renew [ all | apiserver | ... ]
```

K8s-Benutzer hinzufügen

- mehrere Wege
- Beispiel: kubeadm

Benutzer erstellen

```
# kubeadm alpha kubeconfig user \  
--apiserver-advertise-address VIP:VIP_PORT \  
--client-name thorsten \  
--org user | system:masters
```

Demo

Demodateien zu finden unter:

<https://github.com/grischdian/ffg-2019-guug>

Vielen Dank für Ihre Aufmerksamkeit!

Bei weiteren Fragen wenden Sie sich bitte an info@b1-systems.de
oder +49 (0)8457 - 931096