

Container-Orchestrierung in der Cloud für Profis

IT Tage 2018, Frankfurt am Main

12. Dezember 2018

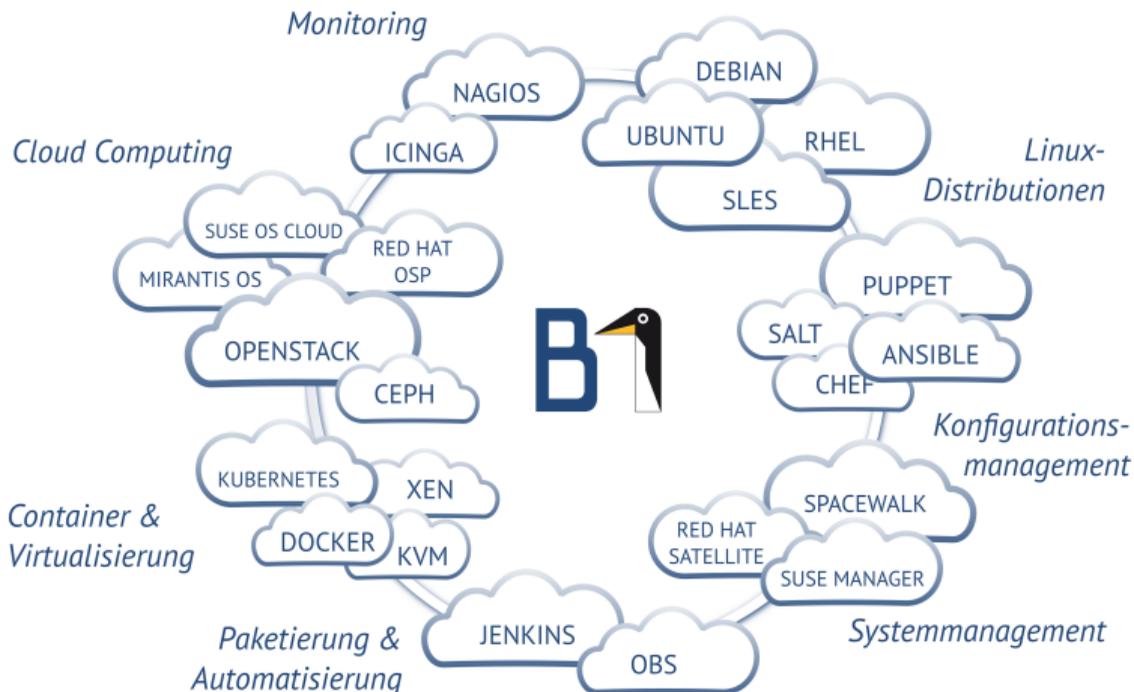


Michel Raabe
Cloud Solution Architect
B1 Systems GmbH
raabe@b1-systems.de

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



Container-Orchestrierung in der Cloud für Profis

Inhalt

- Motivation
- Was ist OpenStack?
- Was ist Kubernetes?
- Was ist OpenStack Heat?
- Was ist OpenStack Magnum?
- Problemstellung
- Magnum Treiber

Motivation

- Praxisbeispiel aus dem Produktiveinsatz
 - OpenStack als Cloud-Stack
 - Kubernetes als Container-Orchestrierung
 - OpenStack Magnum als Verbindungsstück
 - OpenStack-Release Newton im Einsatz (nicht ganz neu)
 - Welche manuellen Anpassungen können nötig sein?

Was ist OpenStack?

- Cloud Computing Plattform
- Infrastructure as a Service
- Apache-Lizenz
- in Python geschrieben

Versionen

- Release: <year>.<version> (e.g. 2010.1)
- Codenamen alphabetisch
- Erste Version: 2010.1 (“Austin”), Oktober 2010
- Aktuelle Version: 2018.1 (“Queens”), Februar 2018
- Dieser Vortrag: 2016.2 (“Newton”)

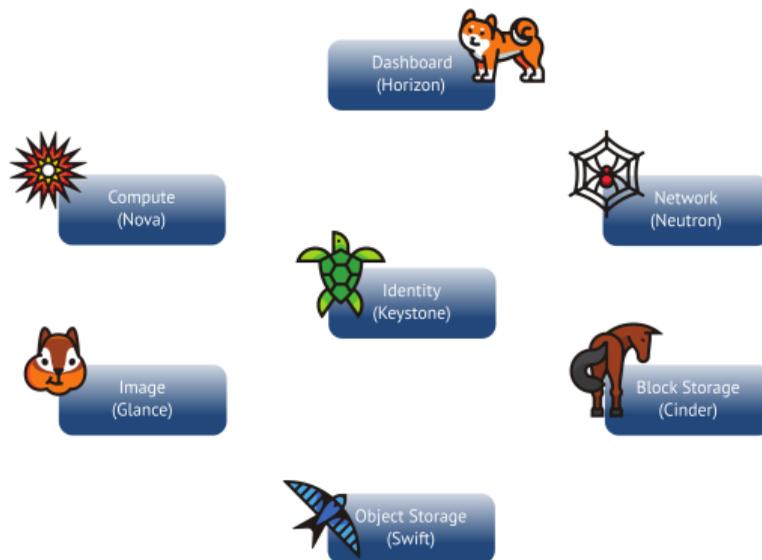
OpenStack Distributionen

- SUSE OpenStack Cloud (SOC)
- Red Hat OpenStack Platform (RHOSP)
- Mirantis Cloud Platform (MCP)
- Canonical OpenStack
- ...

Aufbau

- modularer Aufbau
- 6-7 Kernkomponenten
- insgesamt mehrere dutzend Komponenten

Kernkomponenten



Copyright (project mascots): openstack.org (CC-BY-ND)

Was ist Kubernetes?

- Container-Orchestrierung über mehrere Hosts
- standardisiertes Deployment von Microservices
- Apache-Lizenz

Aufbau von Kubernetes

Master beherbergt die Control Plane

Nodes Hosts für die eigentliche Container-Workload

Cluster besteht aus Nodes und Master und ggf. externem etcd

Was ist Heat?

- Orchestrierung von OpenStack-Ressourcen
- Automatische Konfiguration von Cloud-Applikationen

Aufbau von Heat

Stack eine Menge von Ressourcen, die von Heat verwaltet wird

HOT Heat Orchestration Template – Template für einen Stack

Was ist Magnum?

- OpenStack-Komponente
- Container-Orchestrierung in der Cloud
- mit Heat und z.B. Kubernetes

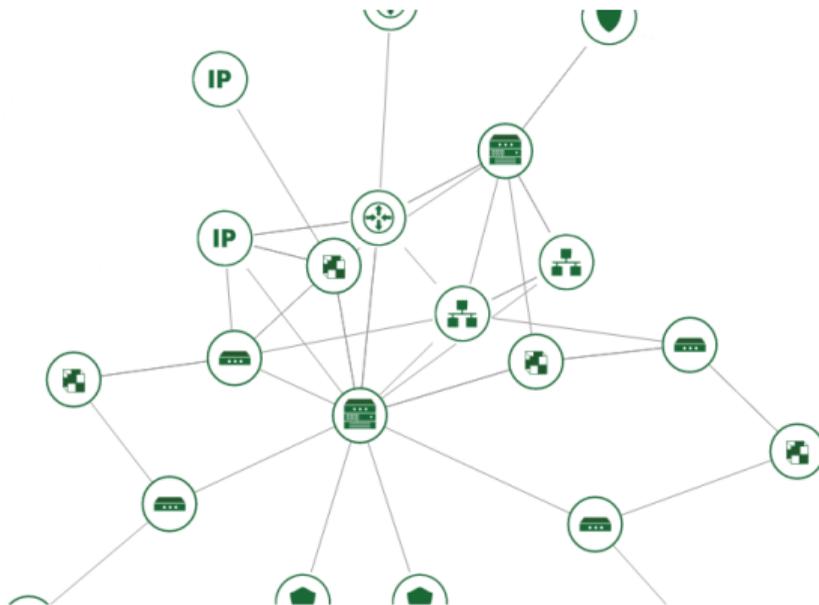
Magnum – Funktionsweise

- startet VMs oder Server
- erstellt Netzwerkressourcen zum Zugriff darauf
- installiert eine Cluster Orchestration Engine
- konfiguriert die VMs oder Server

Magnum – Cluster

- VMs oder Server mit COE
- Netzwerkressourcen
- entspricht einem Kubernetes-Cluster

Einfacher Cluster



Cluster-Template

- Template für einen Cluster
- Basis Glance-Image
- Cluster Orchestration Engine
- SSH-Keypairs
- ...

Cluster-Treiber

- Integration zwischen COE und OpenStack
- Heat-Templates
- Python-Code
- Shell-Skripte
- Container-Images

Wieso Cluster-Treiber anpassen?

- Standard-Treiber in Magnum integriert
- werden für eine Image-Version geschrieben
- Glance-Images verändern sich
- Kunden setzen oft älteres OpenStack ein, aber wollen aktuelles Linux

Allgemeine Probleme

- Cluster-Treiber werden für eine Image-Version geschrieben
- Glance-Images verändern sich
- Kunden setzen oft älteres OpenStack ein, aber wollen aktuelles Linux/Kubernetes

Konkrete Probleme

- Fedora 25 statt Fedora 26
- Folgeprobleme durch Umstellung auf Fedora 26
- Fehlende *HTTPS_PROXY* und *NO_PROXY* Variablen für Docker
- Fehlende NTP-Konfiguration
- Self-signed Root CA

Cluster-Treiber Aufbau

- `magnum/drivers/common/k8s_fedora_atomic_v1`
- `__init__.py`
- `template_def.py`
- `version.py`
- `templates`

template_def.py 1/2

template_def.py

```
...  
class AtomicK8sTemplateDefinition(kftd.K8sFedoraTemplateDefinition):  
    """Kubernetes template for a Fedora Atomic VM."""  
  
    provides = [  
        {'server_type': 'vm',  
         'os': 'fedora-atomic',  
         'coe': 'kubernetes'},  
    ]  
...
```

template_def.py 2/2

template_def.py

```
...  
    @property  
    def driver_module_path(self):  
        return __name__[:__name__.rindex('.')]  
  
    @property  
    def template_path(self):  
        return os.path.join(os.path.dirname(os.path.realpath(__file__)),  
                             'templates/kubeccluster.yaml')
```

kubeccluster.yaml 1/3

- Heat Orchestration Template
- definiert Magnum Cluster
- bindet rekursiv weitere HOTS und Skripte ein

kubecuster.yaml 2/3

templates/kubecuster.yaml

```
...
  kube_masters:
    type: OS::Heat::ResourceGroup
    depends_on:
      - extrouter_inside
    properties:
      count: {get_param: number_of_masters}
      resource_def:
        type: kubemaster.yaml
...

```

kubecuster.yaml 3/3

templates/kubecuster.yaml

```
...
  kube_minions:
    type: OS::Heat::ResourceGroup
    depends_on:
      - extrouter_inside
    properties:
      count: {get_param: number_of_minions}
      removal_policies: [{resource_list: {get_param: minions_to_remove}}]
      resource_def:
        type: kubeminion.yaml
  ...
```

kubemaster.yaml kubeminion.yaml

- Heat Orchestration Templates
- beschreiben Kubemaster und Kubeminion Nodes
- binden Shellskripte als SoftwareConfig ein

kubemaster.yaml

templates/kubemaster.yaml

```
...
  configure_etcd:
    type: OS::Heat::SoftwareConfig
    properties:
      group: ungrouped
      config: {get_file: ../../common/templates/kubernetes/fragments/configure-etcd.sh}
...

```

Vielen Dank für Ihre Aufmerksamkeit!

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