

Automatisiertes Aufsetzen eines Kubernetes-Clusters auf Raspberry Pis mithilfe von Ansible-Playbooks

Automatisiertes Aufsetzen eines Kubernetes-Clusters auf Raspberry Pis mithilfe von Ansible-Playbooks

Agenda

1. Docker
2. Kubernetes
3. Ansible
4. Umsetzung
5. Demo+Austausch

Docker

- Container-virtualisierung
- Isolation von Ressourcen
- Dockerfiles

```
FROM ubuntu:18.04
```

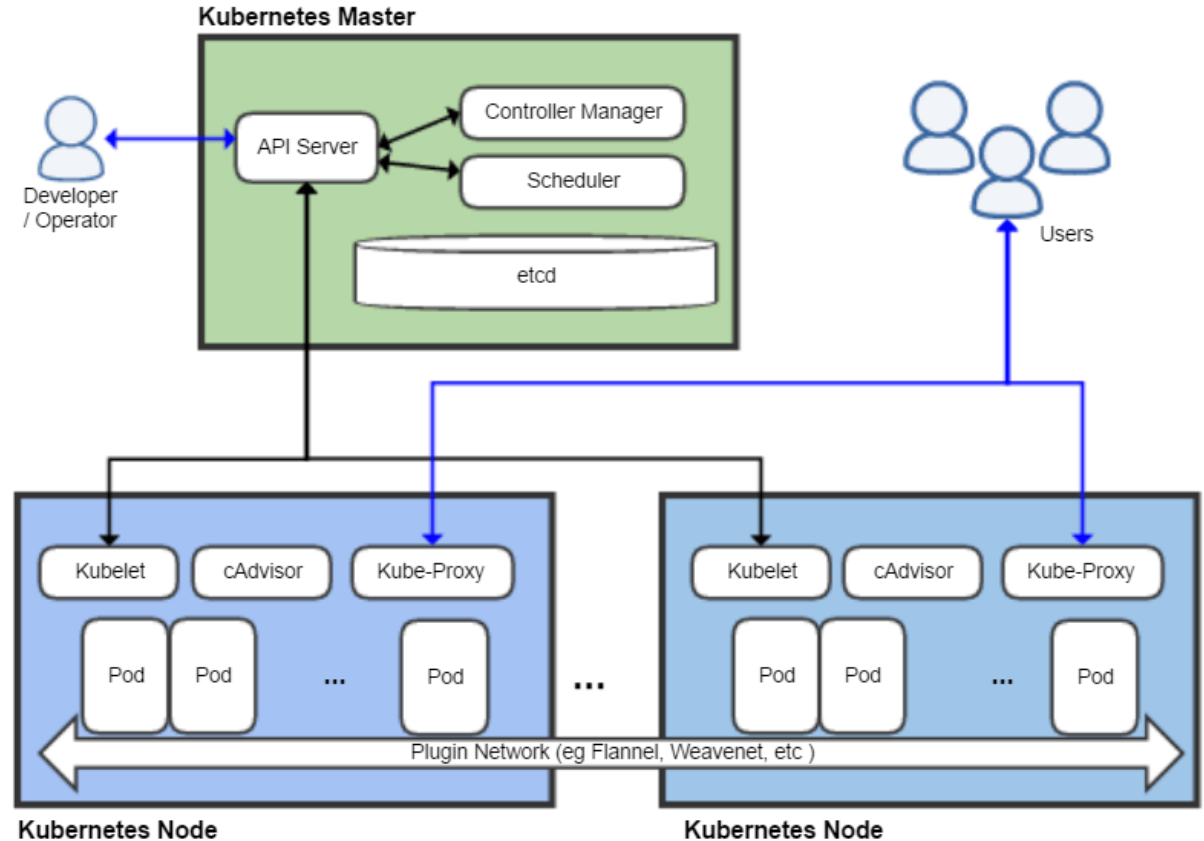
```
RUN apt-get update && \  
    apt-get install -y \  
    redis-server && \  
    apt-get clean
```

```
EXPOSE 6379
```

```
CMD ["redis-server", \  
    "--protected-mode no"]
```

Kubernetes

- Skalierung und Verwaltung von Containern
- Verteilte Hosts/Cloud



Raspberry Pi

- Einplatinencomputer
- ARM-Architektur
- Günstig und sparsam



Ansible

- Automatisierung von Rollouts
- Infrastructure as code
- YAML-Playbooks

```
---
- hosts: all
  become: yes
  tasks:
    - name: Set master flag
      set_fact: master=yes
      when: ansible_host == masterIp

    - name: Set timezone
      timezone:
        name: "{{ timezone }}"

    - name: Upgrade apt packages
      apt:
        upgrade: yes
        update_cache: yes
```

Automatisiertes Aufsetzen eines Kubernetes-Clusters auf Raspberry Pis mithilfe von Ansible-Playbooks

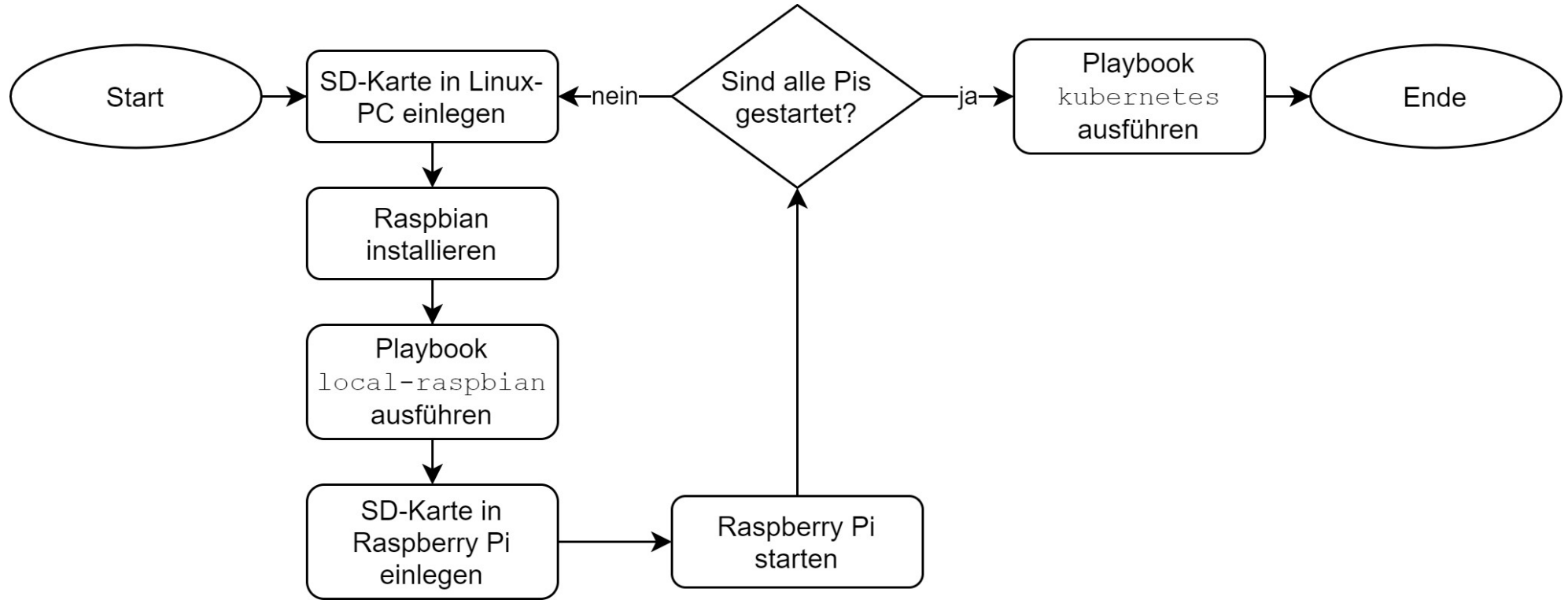
Aufgaben

- Raspbian installieren
- WiFi einrichten (Zugangsdaten, statische IP)
- Swapfile deaktivieren
- Control Groups aktivieren
- SSH-Keys hinterlegen
- Softwarepakete aktualisieren
- Docker installieren
- Kubernetes installieren
- Master Node initialisieren
- Weitere Nodes zum Cluster hinzufügen
- Virtuelles Netzwerk installieren

`local-raspbian.yaml`

`kubernetes.yaml`

Ablauf der Einrichtung



„Live“-Demo

Fragen?

Danke!