



# Ansible Grundlagen

# Kurz zu mir

Name: Christian Frost

Beruf: Irgendwas mit IT und Linux

Arbeitgeber: regio**com** GmbH

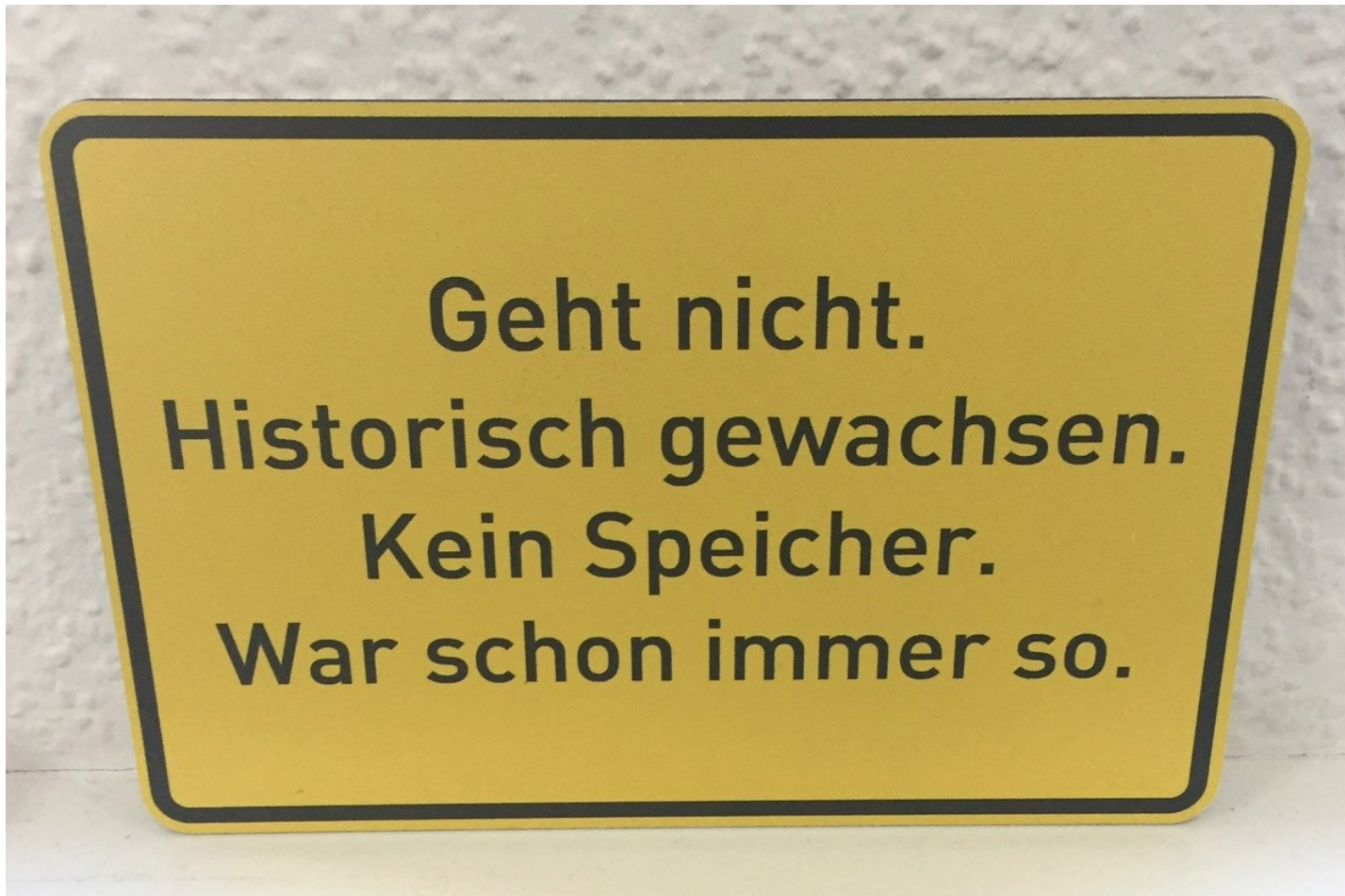
Hobbies: IT, Linux, ...



# Inhalte

- Konfigurationsmanagement
- Warum Ansible?
- Installation von Ansible
- Inventory
- Ad-hoc Befehle
- Ansible Module
- Playbook
- Variablen
- Includes
- Roles
- Facts
- Live Demo
- Ansible Galaxy

# Konfigurations- management



# Konfigurations- management

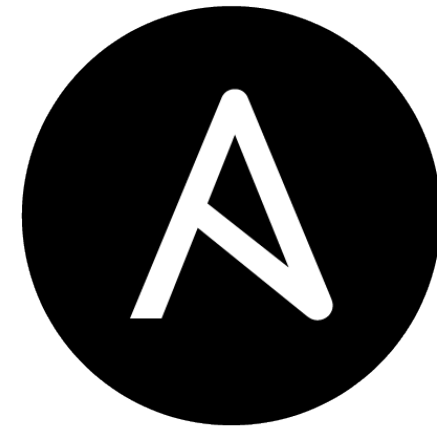
- Regelmäßige/Wiederkehrende Tätigkeiten abnehmen
- Zeitersparnis
- Vermeidung von Fehler (Copy/Past etc.)
- Konfiguration an einem Ort und versioniert
- Nachvollziehbarkeit / Reproduzierbarkeit

# Konfigurations- management



# Warum Ansible?

- Einsteigerfreundlich
- Geringe Anforderungen
  - SSH
  - Python 2.x
- Kein Server
- Kein Agent
- YAML / jinja2



ANSIBLE

# Installation via pip

- [https://docs.ansible.com/ansible/latest/intro\\_installation.html](https://docs.ansible.com/ansible/latest/intro_installation.html)

```
~# sudo pip install ansible
```



# Installation via Yum

- [https://docs.ansible.com/ansible/latest/intro\\_installation.html](https://docs.ansible.com/ansible/latest/intro_installation.html)
  - RHEL/CentOS/Scientific Linux 7 (Extras Channel/Repo)
  - RHEL/CentOS/Scientific Linux 6 (EPEL Repo)

```
~# sudo yum install ansible
```

# Inventory

```
~# cat /etc/ansible/hosts
```

```
# Ungruppiert  
172.16.123.123  
mail01  
mail02
```

```
# Gruppe  
[mailserver]  
mail01  
mail02
```

```
# Fortlaufend  
[mailserver]  
mail[01:02]
```

# Ad-hoc Befehle

```
ansible <hosts> -m <modul> -a <args>
```

```
~# ansible all -m ping
```

```
~# ansible mailserver -m command -a "uptime"
```

```
~# ansible mail01 -m yum -a "name=curl state=present"
```

# Ansible Module

- In Python geschrieben
- Sind das eigentliche Werkzeug
- Beispiel copy, file, yum, apt

[http://docs.ansible.com/ansible/latest/modules\\_by\\_category.html](http://docs.ansible.com/ansible/latest/modules_by_category.html)

[http://docs.ansible.com/ansible/latest/list\\_of\\_all\\_modules.html](http://docs.ansible.com/ansible/latest/list_of_all_modules.html)

# Playbook

- Beschreibung YAML
- Besteht aus Play's
  - Diese bestehen wiederum aus Tasks

# Playbook

---

- hosts: mailserver
- tasks:
  - name: Installation von screen
  - yum:
    - name: screen
    - state: present

# Playbook

---

- hosts: mailserver  
tasks:
  - name: ensure postfix is at the latest version  
yum: name: postfix state: latest
  - name: Postfix Konfiguration erstellen  
template:
    - src: postfix.j2
    - dest: /etc/postfix/main.cf
    - mode: 0644
    - notify:
      - restart postfix
- handlers:
  - name: restart postfix  
service: name=postfix state=restarted

# Playbook

- Playbooks können mehrere Plays enthalten bzw. mehrere Tasks
- Große Playbooks werden schnell unübersichtlich
- Aufteilung in einzelne Aufgaben
- Durch die Aufteilung eine Wiederverwendbarkeit schaffen



# Variablen

```
mailserver/defaults/main.yml
```

```
- relay: relayhost = mailrelay.intern
```

```
mailserver/tasks/main.yml
```

```
- name: Add mailrelay to postfix config
  blockinfile:
    path: /etc/postfix/main.cf
    block: |
      {{ relay }}
```

# Include

---

- hosts: mailserver
- tasks:
  - include postfix.yml
  - include dovecot.yml

**DEPRECATED**

Siehe  
[https://docs.ansible.com/ansible/latest/include\\_module.html](https://docs.ansible.com/ansible/latest/include_module.html)

# Roles

- Rollen sind Möglichkeiten zum automatischen einbinden von `var_files`, `tasks` und `handlers`
- Dies erfolgt auf Basis einer festgelegten Dateistruktur

# Roles

---

- hosts: mailserver
- roles:
  - postfix
  - dovecot

# Roles – Datei Struktur

mailserver.yml

roles/

  postfix/

    tasks/       - tasks

    handlers/   - notify

    files/       - Dateien

    templates/  - Vorlagen

    vars/        - Variablen mit hoher Priorität

    defaults/   - Variablen mit geringer Priorität

    meta/        - Abhängigkeiten

# Facts

- Ansible sammelt vor jedem Start Daten (Facts) über ein Zielsystem
- Können auch über einen Ad-hoc Befehl abgerufen werden

```
~# ansible <host> -m setup
```

- Diese Facts können auch als Variable verwendet werden

```
common/templates/motd.j2
```

```
#####
```

```
# Servername: {{ ansible_hostname }}
```

```
# OS          : {{ ansible_distribution }}
```

```
#####
```

# Live Demo

# Ansible Galaxy

- <https://galaxy.ansible.com>
- Fertige Rollen zum Download verfügbar
- Werden durch Community Mitglieder erstellt/gepflegt
- Einfach in der Verwendung

```
~# ansible-galaxy install rollen.name  
- hosts: mailserver  
  roles:  
    - { role: rollen.name }
```



# Fragen?



Vielen Dank für eure Aufmerksamkeit :)