

Ansible, Molecule und systemd

Das automatisierte Testen automatisierter Systemkonfigurationen

Orloff

Hackspace Siegen

12. September 2025



① Person und Hintergrund

② Technische Übersicht

Ansible

systemd-nspawn

Molecule

③ Testing

Clean Systems

Testing

④ Outro

Fragen

Info



- ▶ Orloff
- ▶ Hackspace Siegen



- ▶ Init: SuSE 7.0



- ▶ Linux - SysAdmin
- ▶ HA-Cluster (Pacemaker/Corosync) mit LXC
- ▶ SystemD - NSpawn





Grundfunktionen

- ▶ **Systemkonfigurationen**
- ▶ **System-Orchestrierung**
- ▶ **Administration**

Architektur

- ▶ **Module/Rollen/Collections**
- ▶ **Inventar**
- ▶ **Playbooks/Arbeitsanweisungen**



ANSIBLE

► Konfigurationsautomatisierung

```
$ cat apache2.yml
- name: Apache2
  hosts: all

  tasks:
    - name: Include Role
      ansible.builtin.import_role:
        name: apache2
      when: "'Webserver' in group_names"
```

► Job-Automatisierung

```
$ cat update.yml
- name: Packages-Updates
  hosts: all

  tasks:
    - name: Paketupdates (APT)
```



► Was sind Konfigurationen ?

```
vhosts:
  - vhost: wooden-infrastructure.de
    aliases:
      - www.wooden-infrastructure.de
    docRoot: /var/www/wooden-infrastructure.de/
    ssl: true
    ssl_cert: /etc/apache2/ssl/wooden-infrastructure.de/wooden-infrastructure.de.cer
    ssl_key: /etc/apache2/ssl/wooden-infrastructure.de/wooden-infrastructure.de.key
    ssl_ca: /etc/apache2/ssl/wooden-infrastructure.de/ca.cer
```

```
letsencrypt:
  - primary: wooden-infrastructure.de
    secondary:
      - www.wooden-infrastructure.de
```

- Wie kann man diese automatisieren ?
- Wie kann man diese automatisiert testen ?

```
$ cat apache2.yml
- name: Apache2
  hosts: all
```



► Was sind Jobs ?

```

root@ovh-web01:~# apt update && apt upgrade
Hit:1 https://deb.debian.org/debian bookworm InRelease
Get:2 https://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:3 https://security.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Hit:4 https://packages.sury.org/php bookworm InRelease
Ign:5 https://download.onlyoffice.com/repo/debian squeeze InRelease
Hit:6 https://download.onlyoffice.com/repo/debian squeeze Release
Fetched 103 kB in 0s (235 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
7 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  libnode-dev libnode108 libunbound8 libxml2 libxml2-dev nodejs nodejs-doc
7 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/17.1 MB of archives.
After this operation, 348 kB of additional disk space will be used.
Do you want to continue? [Y/n] n

```

- Wie kann man diese automatisieren ?
- Wie kann man diese automatisiert testen ?

```

$ cat update.yml
- name: Packages-Updates
  hosts: all

```



► Was sind AdHoc-Jobs ?

```
root@ovh-web01:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 12 (bookworm)"
NAME="Debian GNU/Linux"
VERSION_ID="12"
VERSION="12 (bookworm)"
VERSION_CODENAME=bookworm
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

► Sollte man diese automatisieren ?

► Wie kann man diese automatisiert testen ?

```
$ ansible -i hosts -m command -a "cat /etc/os-release" ovh_web01
ovh_web01 | CHANGED | rc=0 >>
PRETTY_NAME="Debian GNU/Linux 12 (bookworm)"
NAME="Debian GNU/Linux"
VERSION_ID="12"
VERSION="12 (bookworm)"
```



- ▶ **Was ist das ?**
 - ▶ **Container**
 - ▶ **LXC**
 - ▶ **CGroups**
 - ▶ **SystemD-Tools**



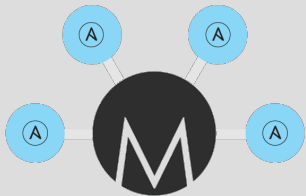
- ▶ **Struktur**
 - ▶ Netzwerk
 - ▶ Parameter
- ▶ **Betrieb**
 - ▶ Services
 - ▶ MachineCTL
- ▶ **Tools**
 - ▶ MachineCTL
 - ▶ SystemCTL



► Testframework

► Testablauf

- create
- dependency
- prepare
- converge
- idempotence
- side_effect
- verify
- cleanup
- destroy



Molecule

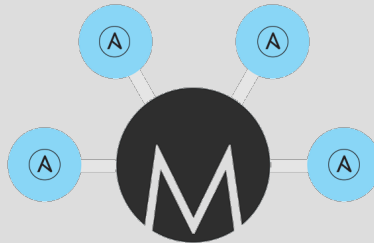


Erwartete Facts

- ▶ Funktionsprüfung
- ▶ Ungegwollte zyklische Anpassungen vermeiden
- ▶ Ergebnis validieren

Environment

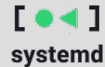
- ▶ Clean Systems
- ▶ Wiederholbarkeit
- ▶ Modifiziertes Environment



Molecule



- ▶ **Docker/Podman**
- ▶ **Kubernetes/etc.**
- ▶ **Cloud-API**
- ▶ **CHRoot on Steroids (systemd-nspawn)**



► Debootstrap

- Debian

- Ubuntu

► YUM/DNF

- fedora

- rhel

- oracle

- CentOS

- Alma Linux

- Rocky Linux

► pacman/pacstrap

- ArchLinux

- EndeavourOS

- Manjaro



ORACLE®
Linux



► Cleanup

```
name: Molecule | Container | Check  
ansible.builtin.shell:  
cmd: "machinectl list-images --no-legend | awk '{ print $1 }'"  
register: japan_images  
run_once: true  
  
tasks:  
- name: Molecule | Container | Check images  
block:  
  - name: Molecule | Container | Check Image  
    ansible.builtin.shell:  
      cmd: "echo $( { if [ ${rc} = 0 ]; then echo $(machinectl show-image {{ item.dist }} + {{ item.release }} ); fi; } && cat /dev/null )" >/tmp/japan_images  
    register: japan_image  
  - name: Molecule | Container | Cleanup Image  
    ansible.builtin.shell:  
      path: "/var/lib/machines/{{ item.dist }} + {{ item.release }}"  
      state: absent  
    when: "  
japan_image.stdout | length == 0"  
  when: "item.dist != '' && item.release != japan_images.stdout_lines"  
- name: Molecule | Container | Check  
ansible.builtin.shell:  
cmd: "machinectl list-images --no-legend | awk '{ print $1 }'"  
register: japan_images  
run_once: true  
  
- name: Molecule | Container | Create Traces  
block:  
  - name: Molecule | Container | Create Debian  
    ansible.builtin.command:  
      cmd: "debostinstall --include={{ basepackages | join(',') }} components-main,contrib,non-free {{ item.release }} {{ item.release }} https://cat.debian.org/debian/"  
      chdir: "/var/lib/machines/  
newsrc: "usr/lib/machine{{ item.name }}"/usr  
when: "item.dist | lower == 'debian'"  
vars:  
  basepackages: ["basepackages", "basepackages_release:item.release | lower; | default([])"]  
  basepackages:  
    - atari68k  
    - cephfs_server  
    - python  
    - python3-gpt  
  basepackages_release:  
    kubernetes  
    dnf4_broker  
    systemd-resolved  
    trisect  
    dnf4_broker  
    systemd-resolved  
when: "item.dist != '' && item.release not in japan_images.stdout_lines"  
- name: Molecule | Container | Create from Image  
ansible.builtin.shell:  
cmd: "machinectl start {{ item.dist }} + {{ item.release }} {{ item.name }}
```



- Create
- Prepare
- Converge
- Idempotence
- Verify
- Cleanup
- Destroy

```
---
driver:
  name: default
platforms:
  name: deb11
  dist: debian
  release: bullseye
  ip: "10"
  groups:
    standalone
- name: deb12
  dist: debian
  release: bookworm
  ip: "11"
  groups:
    - webserver
  name: deb13
  dist: debian
  release: trixie
  ip: "12"
  groups:
    - webserver
```



- ▶ **systemd-nspawn**
- ▶ **Container**
- ▶ **Caching**
- ▶ **Systemkonfigurationen**
- ▶ **Instance-Config**

```
- name: Create testcontainers based on systemd-nspawn
block:
  name: Molecule | Nspawn | Network | SysCTL
  ansible.postix.system1:
    name: net.ipv4.ip_forward
    value: "1"
    state: present
    sysctl set: true
    reload: true

  name: Molecule | Nspawn | Network | NetDev
  ansible.builtin.copy:
    src: mol0.netdev
    dest: /etc/systemd/network/xol0.netdev
    owner: root
    group: root
    mode: '0644'

  name: Molecule | Nspawn | Network | NetworkdInitrd
  ansible.builtin.copy:
    src: mol0.network
    dest: /etc/systemd/network/xol0.network
    owner: root
    group: root
    mode: '0644'

  name: Molecule | Nspawn | Network | Resolved
  community.general.ini_file:
    path: /etc/systemd/resolved.conf
    action: resolve
    option: DNSStubListenerExtra
    value: "172.21.42.254"
    exclusive: false
    owner: root
    group: root
    mode: '0644'

  name: Molecule | Nspawn | Path
  ansible.builtin.file:
    path: /etc/systemd/nspawn
    state: directory
    owner: root
    group: root
    mode: '0755'

  name: Molecule | Nspawn | Configs
  ansible.builtin.template:
    src: nspawn.t2
    dest: /etc/systemd/nspawn/{{ item.name }}.nspawn
    owner: root
    group: root
    mode: '0644'

  vars:
    hosttrave: "{{ item.name }}"
    with_items: "{{ molecule_yaml.platforms }}"
```



- **Software-Installation**
- **Software-Konfiguration**
- **Service-Verfügbarkeit**
- **Prepare ENV**

```
- name: Molecule | Pebble | Install deps
  community.general.pacman:
    name:
      - go
    when: "ansible_os_family | lower == 'archlinux'"

- name: Molecule | Pebble | GIT
  ansible.builtin.git:
    repo: https://github.com/letsencrypt/pebble.git
    dest: /tmp/pebble
```

```
name: Molecule | Pebble | SSL Key
community.crypto.openssl_privatekey:
  path: /tmp/pebble/pebble.key
  mode: '0644'

- name: Molecule | Pebble | SSL-CSR
  community.crypto.openssl_csr:
    path: /tmp/pebble/pebble.csr
    privatekey_path: /tmp/pebble/pebble.key
    common_name: "172.21.42.254"
    subject_alt_name: 'DNS:172.21.42.254,DNS:pebble.molecule.local'
    key_usage:
      - digitalSignature
    key_usage_critical: true
    extended_key_usage:
      - serverAuth
      - clientAuth
```



- Prepare
- Playbooks

```
tasks:
  - name: Molecule | Converge | Prepare needed packages for webserver
    ansible.builtin.apt:
      name:
        - curl
        - socat
        - apache2
        - php-fpm
      when: "ansible_os_family | lower == 'debian' and 'webserver' in group_names"

  - name: Molecule | Load role-defaults
    ansible.builtin.include_vars:
      file: ../../../../roles/apache2/defaults/main.yml
```

```
- name: LetsEncrypt | Testing
  hosts: webserver
  tasks:
    - name: Letsencrypt | Role-Test
      ansible.builtin.include_role:
        name: ../../roles/letsencrypt
      tasks from: main.yml
```





- Prepare
- Playbooks

```
PLAY RECAP *****
deb11      : ok=29   changed=8    unreachable=0    failed=0    skipped=23    rescued=0    ignored=0
deb12      : ok=50   changed=22   unreachable=0    failed=0    skipped=23    rescued=0    ignored=0
deb13      : ok=50   changed=22   unreachable=0    failed=0    skipped=23    rescued=0    ignored=0
```

```
PLAY RECAP *****
deb11      : ok=25   changed=0     unreachable=0    failed=0    skipped=2 /    rescued=0    ignored=0
deb12      : ok=46   changed=0     unreachable=0    failed=0    skipped=26    rescued=0    ignored=0
deb13      : ok=46   changed=0     unreachable=0    failed=0    skipped=28    rescued=0    ignored=0

INFO      default 1 idempotence: Idempotence completed successfully.
INFO      default 1 idempotence: Completed
```



► Prepare

► Assert

```
tasks:
  - name: Verify | File | CA Cert
    ansible.builtin.stat:
      path: "{{ letencrypt_ca_certs }}"
    register: _ca_cert
  - name: Verify | File | CA-Cert
    ansible.builtin.assert:
      that:
        - "_ca_cert.stat.exists"
        - "_ca_cert.stat.mode == '0644'"
  - name: Verify | File | acme.conf
    ansible.builtin.stat:
      path: "/etc/apache2/conf-available/acme.conf"
    register: _acme_file
  - name: Verify | File | acme.conf
    ansible.builtin.assert:
      that:
        - "_acme_file.stat.exists"
        - "_acme_file.stat.mode == '0644'"
  - name: verify | file | acme.conf (link)
    ansible.builtin.stat:
      path: "/etc/apache2/conf-enabled/acme.conf"
    register: _acme_link
  - name: Verify | File | acme.conf (link)
    ansible.builtin.assert:
      that:
        - "_acme_link.stat.exists"
        - "_acme_link.stat.islnk"
        - "_acme_link.stat.lnk_target == './conf-available/acme.conf'"
        - "_acme_link.stat.lnk_source == '/etc/apache2/conf-available/acme.conf'"
  - name: Verify | Letsencrypt | Get default-ca
    ansible.builtin.shell: >
      {{ letencrypt_acmeh.installpath }}/acme.sh --info | grep DEFAULT_ACME_SERVER | cut -d '"' -f 2 | sed -e s/'/'//g
    register: _acme_default_ca
  - name: Verify | Letsencrypt | Get account-email
    ansible.builtin.shell: >
      {{ letencrypt_acmeh.installpath }}/acme.sh --info | grep ACCOUNT_EMAIL | cut -d '"' -f 2 | sed -e s/'/'//g
    register: _acme_account_email
  - name: Verify | Letsencrypt | Get ca bundle
    ansible.builtin.shell: >
      {{ letencrypt_acmeh.installpath }}/acme.sh --info | grep CA_BUNDLE | cut -d '"' -f 2 | sed -e s/'/'//g
    register: _acme_ca_bundle
  - name: Verify | Letsencrypt | Assert
    ansible.builtin.assert:
      that:
        - "_acme_default_ca.stdout == letencrypt_default_server"
        - "letencrypt_default_server is defined"
  - name: verify | Letsencrypt | Assert
    ansible.builtin.assert:
      that:
        - "_acme_account_email.stdout == letencrypt_account_email"
        - "_acme_ca_bundle.stdout == letencrypt_ca_bundle"
```

- name: Verify



Wir bereinigen das Hostsystem

► Shutdown

► Cleanup

```
- name: Molecule | Pebble | Stop service
  ansible.builtin.systemd:
    name: pebble.service
    state: stopped
```

```
- name: Molecule | Pebble | Remove infos
  ansible.builtin.file:
    path: "{{ item }}"
    state: absent
  with_items:
    - "/usr/local/bin/pebble"
    - "/tmp/pebble"
    - "/etc/systemd/system/pebble.service"
```

- Shutdown
- Cleanup
- Inventory

```
- name: Molecule | Container | Shutdown
  ansible.builtin.systemd:
    name: systemd-nspawn@{{ item.name }}
    state: stopped
  with_items: "{{ molecule_yml.platforms }}"
```

```
- name: Molecule | Container | Remove paths
  ansible.builtin.file:
    path: /var/lib/machines/{{ item.name }}
    state: absent
  with_items: "{{ molecule_yml.platforms }}"

- name: Molecule | Nspawn | Remove configs
  ansible.builtin.file:
    path: /etc/systemd/nspawn/{{ item.name }}.nspawn
    state: absent
  with_items: "{{ molecule_yml.platforms }}"

- name: Molecule | Nspawn | Network | ResolveD
  community.general.ini_file:
    path: /etc/systemd/resolved.conf
    section: Resolve
    option: DNSStubListenerExtra
    value: "172.21.42.254"
    exclusive: false
    state: absent
    owner: root
    group: root
    mode: '0644'

  name: Molecule | Nspawn | Restart Services
  ansible.builtin.systemd:
    state: restarted
    name: "{{ item }}"
  with_items:
    - systemd-resolved
```



Ansible,
Molecule und
systemd

Orloff

Person und
Hintergrund

Technische
Übersicht

Testing

Outro

Fragen

Info



Fragen ?



(a) Infra-Blog



(b) Git-Repo

Ansible,
Molecule und
systemd

Orloff

Person und
Hintergrund

Technische
Übersicht

Testing

Outro

Fragen

Info

Dies war mein erster Vortrag!
Habt ihr auch ein Thema ?

