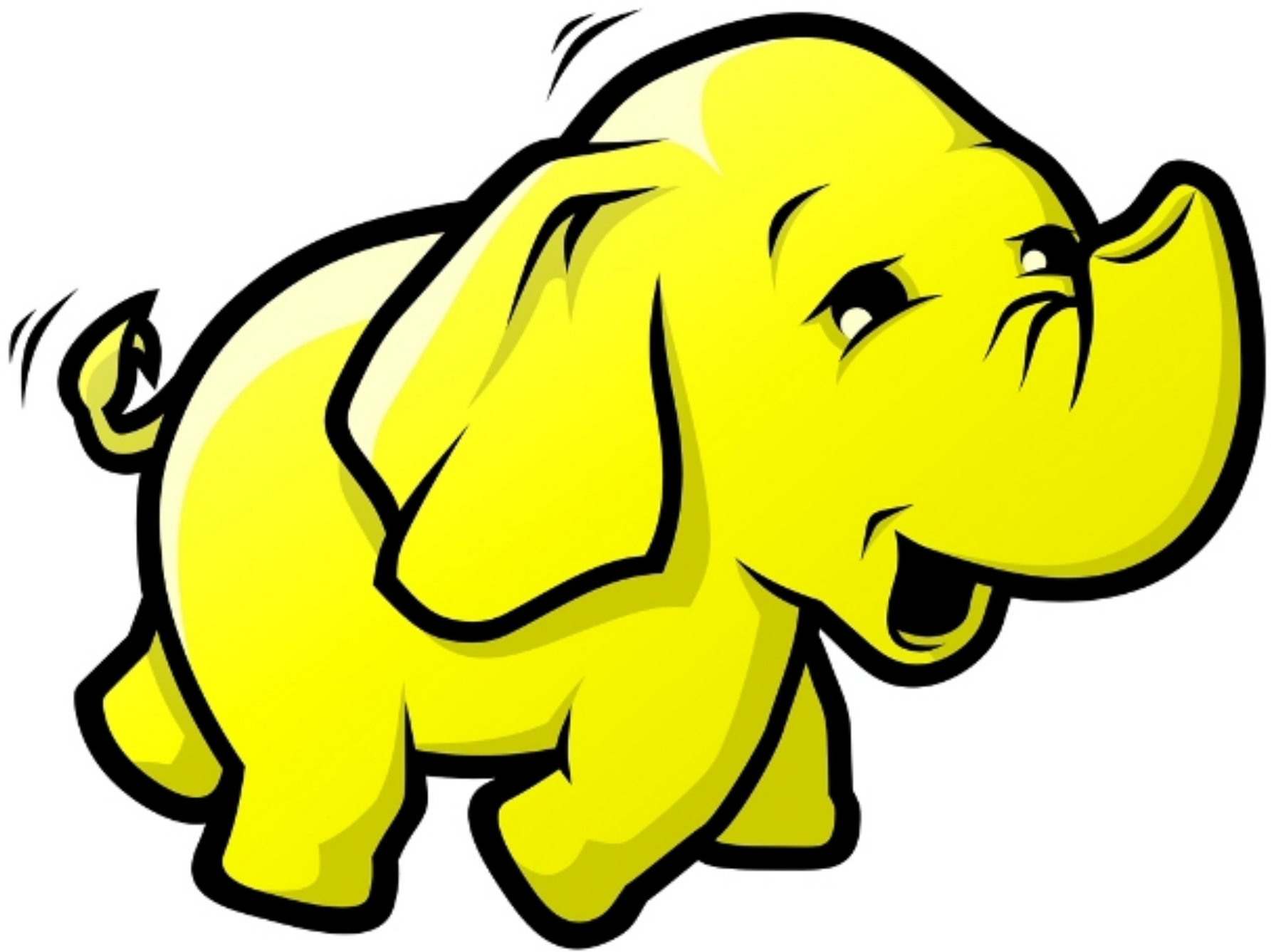


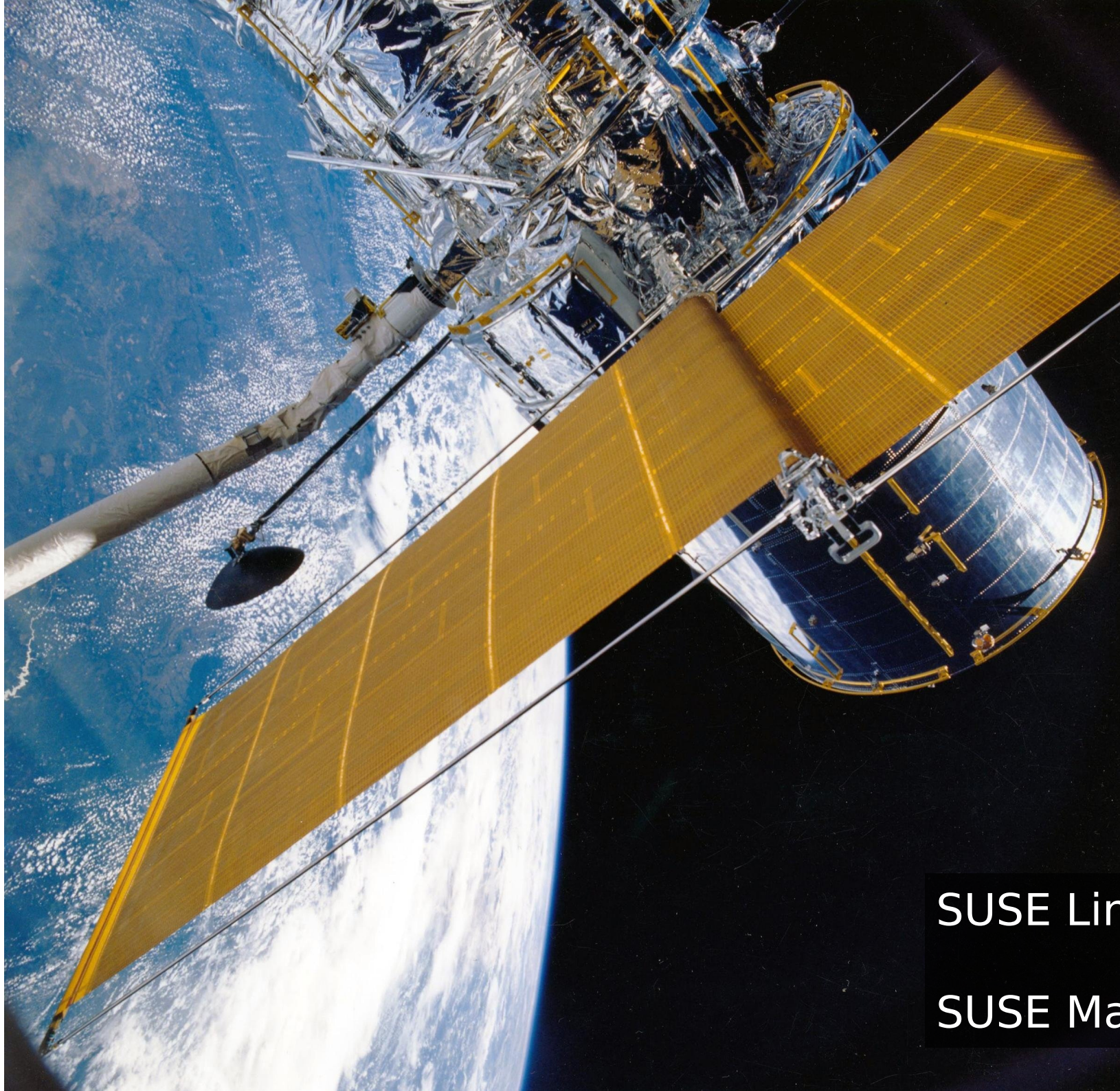


Bare Metal Provisioning with Ansible and Cobbler

Felix Frank
(@felis_rex)

The unbelievable
Machine Company





SUSE Linux
SUSE Manager



Provisioning Stack
Cobbler
HPE ProLiant Gen10
VMWARE
+ one surprise tool



Platform Stack

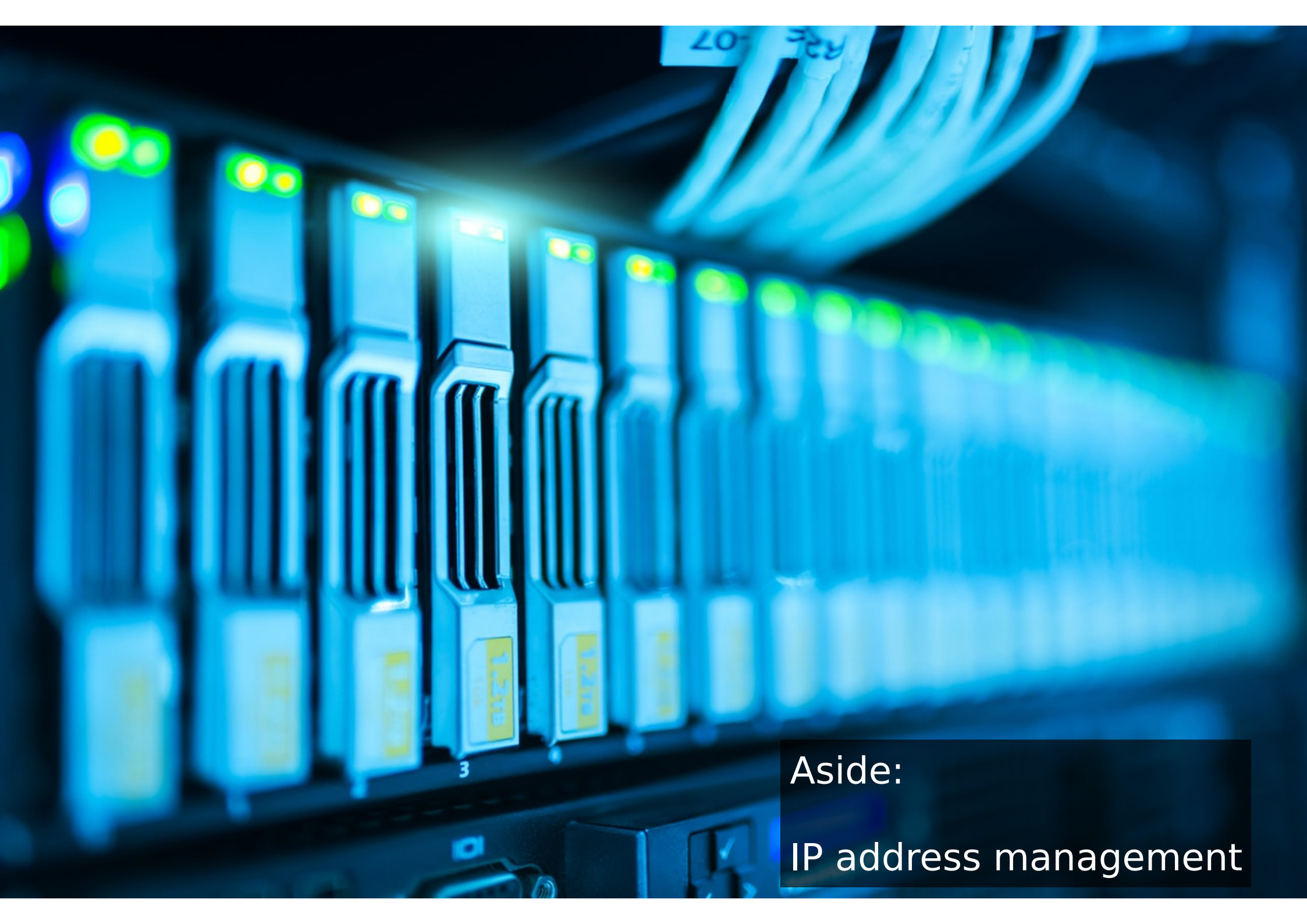
FreeIPA

- DNS

- LDAP

Kerberos

Salt



Aside:
IP address management



Add prefix

Query took 0.05 seconds.

search help?

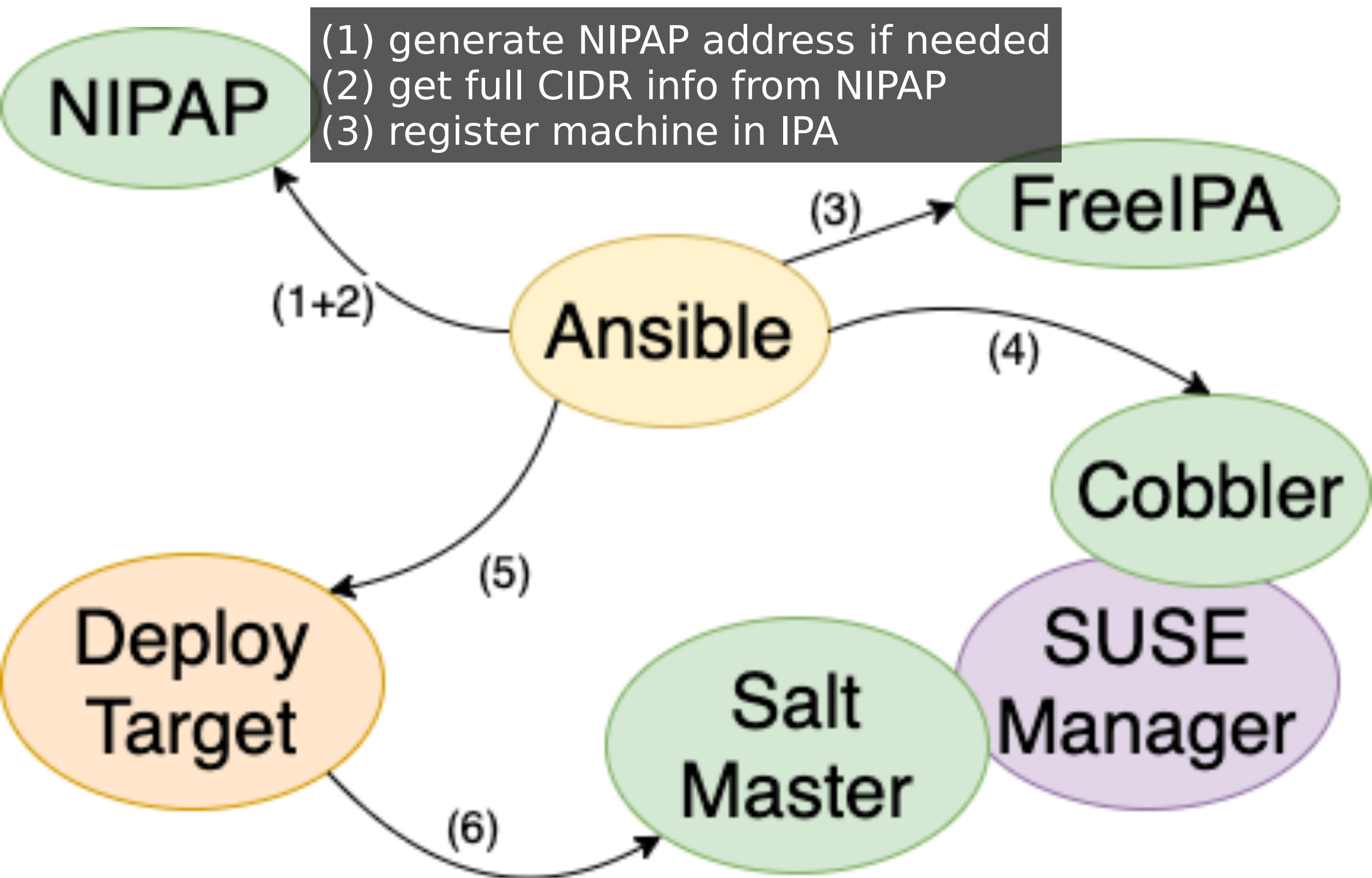
Search interpretation : text matching "

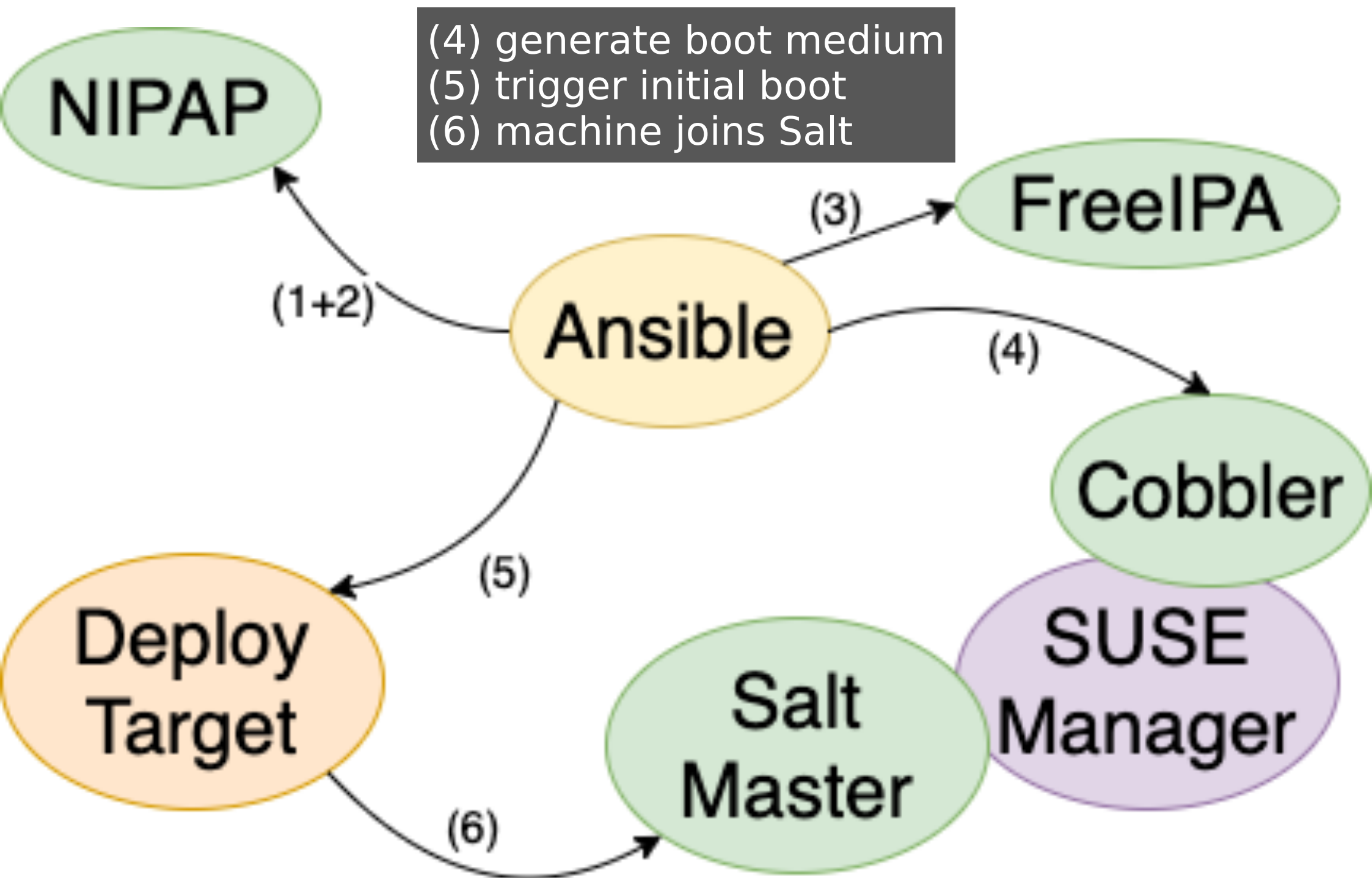
VRF	Prefix	Order	Node	Description
1357985792:199118	+ 80.241.56.0/21	R		
	+ 80.241.56.0/22	R		
	+ 80.241.56.0/23	R		
	+ 80.241.56.0/24	A		
	- 80.241.57.0/24	A		
	80.241.57.1/24	H		
	80.241.57.2/24	H		
	+ 80.241.58.0/23	R		
	+ 80.241.58.0/24	A		
	+ 80.241.59.0/24	A		
	+ 80.241.60.0/22	R		
	+ 80.241.60.0/29	A		
	+ 80.241.60.64/26	A		
	- 2001:67c:2050::/48	R		
	+ 2001:67c:2050::/56	R		
	- 2001:67c:2050::/64	A		
	2001:67c:2050:1/64	H		
	2001:67c:2050:2/64	H		
	- 2001:67c:2050:1::/64	A		
	2001:67c:2050:1::1/64	H		
	- 2001:67c:2050:100::/56	R		
	+ 2001:67c:2050:100::/64	A		

Introducing
NIPAP

Hierarchical subnet planning in NIPAP

- 10.139.0.0/19 Subnet for Berlin location
 - 10.139.0.0/20 Location DC Berlin 01
 - + 10.139.0.0/26 VIRTUALIZATION
 - + 10.139.0.64/27 SERVICE
 - + 10.139.0.96/27 INTEGRATION CLUSTER 01
 - ...
 - 10.139.16.0/20 Location DC Berlin 02
 - .
 - .
 - .
- 10.184.202.0/22 Subnet for Atlanta location
 - ...







```
- name: "Register host in nipap"
  shell: next-host.sh -e -y {{ host_name }} {{ host_net | upper }}
  delegate_to: "{{ nipap_server }}"

- name: "Get base address information from nipap"
  shell: nipap address list -c node,prefix | grep {{ host_name }}
  register: nipap_info
  changed_when: False
  failed_when: nipap_info is failed or nipap_info.stdout == ""
  check_mode: no

- name: "Get vlan from nipap"
  shell: nipap address view {{ address }} vrf {{ host_vrf }}
        | awk '/VLAN/ { print $3 }'
  register: vlan
  vars:
    address: {{ nipap_info.stdout | ipaddr('network/prefix') }}
  changed_when: False
  check_mode: no
```



```
# generating boot media
```

```
- name: "mkiso_sles: Create autoyast bootcd with mksusedc"
```

```
  command: >
```

```
    mksusedc
```

```
    --create {{ boot_iso_path }}
```

```
    --nano --uefi --gpt
```

```
    -boot "{{ boot_params }}" {{ suma_iso_path }}
```

```
  vars:
```

```
    ifcfg_option: >-
```

```
    {{ iface }}={{ cidr }},{{ gateway }},{{ nameserver }},{{ domain }}
```

```
    boot_params: >-
```

```
    ksdevice=bootif lang=en_US pt.options=+self_update
```

```
    install={{ cobbler_dist_url }}
```

```
    hostname={{ host_name_short }}
```

```
    self_update=0 text autoyast={{ cobbler_kickstart_url }}
```

```
    ksdevice=bootif textmode=1
```

```
    {% if machine_category == 'HW' %}
```

```
    biosdevname=1
```

```
    {% endif %}
```

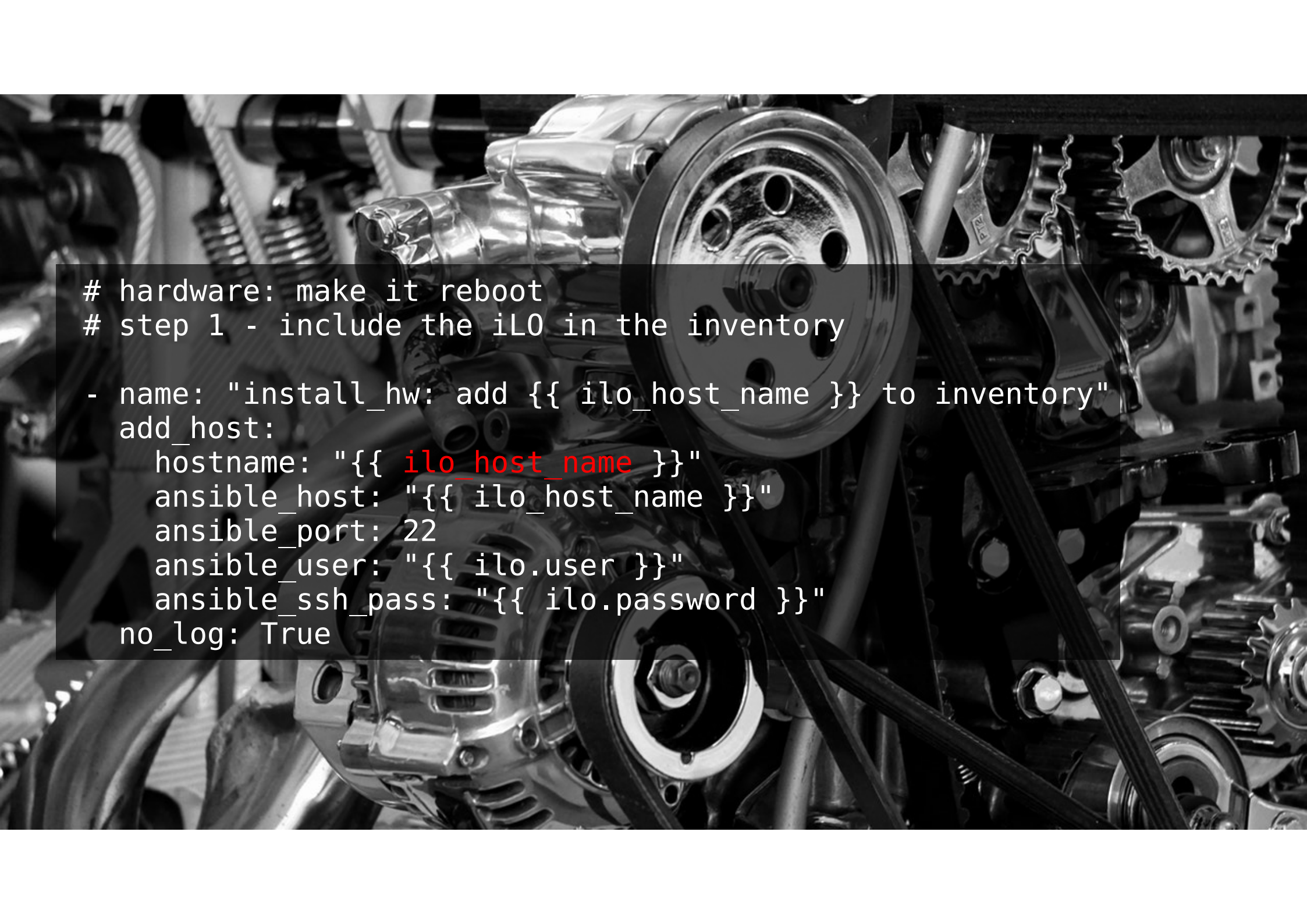
```
    ifcfg={{ ifcfg_option }}
```



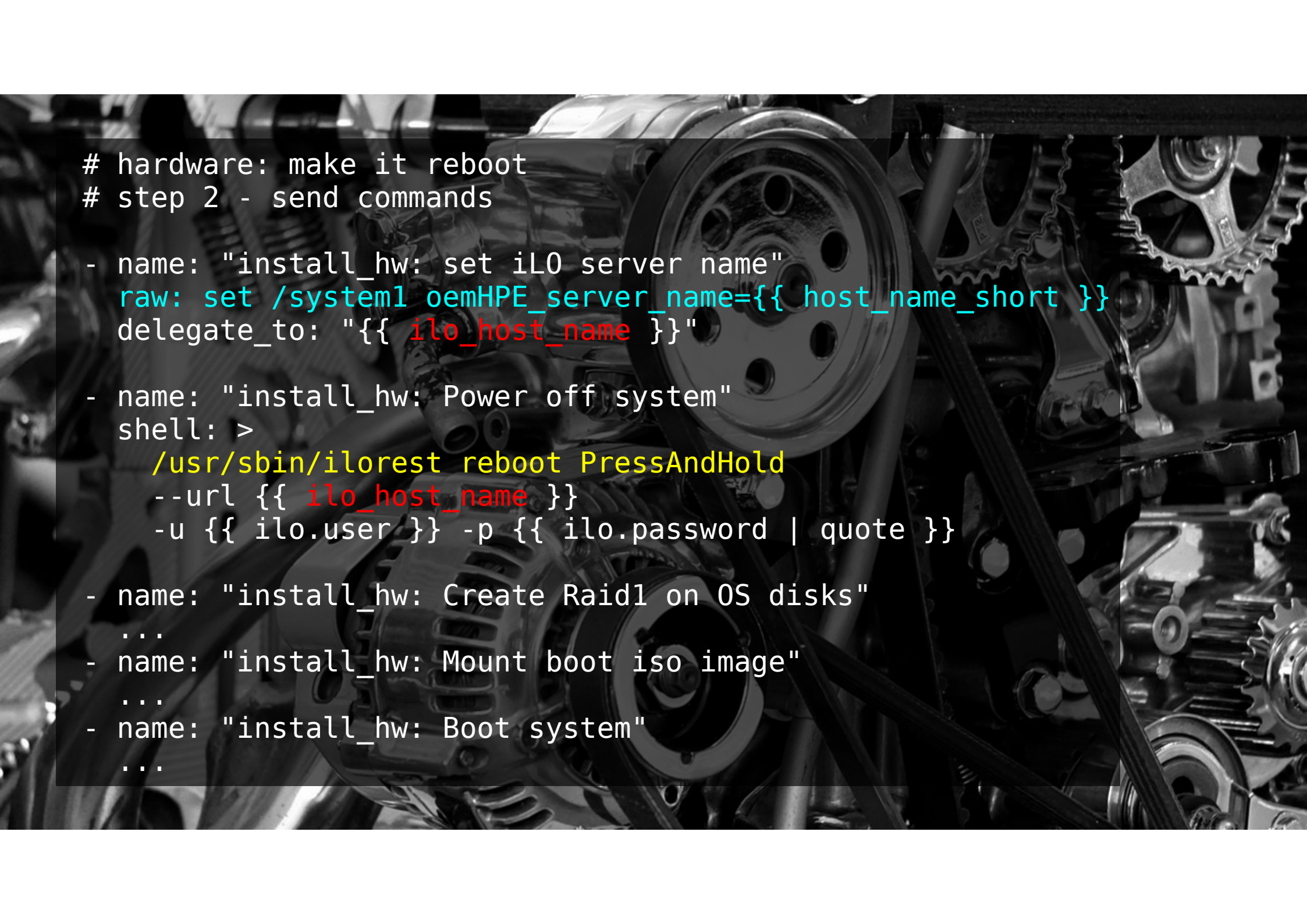

```
# virtual machine: create from scratch

- name: "install_vm: Create VM using govc"
  environment:
    GOVC_USERNAME: "{{ vcenter.user }}"
    GOVC_PASSWORD: "{{ vcenter.password }}"
  command: >
    govc vm.create
    -k=true -debug -u '{{ vcenter.host | quote }}' -on=false
    -m={{ vm_memory | quote }}
    -c={{ vm_cpus | quote }}
    ...
    -iso={{ bootiso_name | quote }}
    -iso-datastore={{ vm_isostore | quote }}
    {{ host_name_short | quote }}

# for additional disks
- name: "install_vm: Query govc for information about the new VM"
  command: govc vm.info ...
- name: "install_vm: Add data disk volume using govc"
  command: govc vm.disk.create ...
```

```
# hardware: make it reboot
# step 1 - include the ilo in the inventory
- name: "install_hw: add {{ ilo_host_name }} to inventory"
  add_host:
    hostname: "{{ ilo_host_name }}"
    ansible_host: "{{ ilo_host_name }}"
    ansible_port: 22
    ansible_user: "{{ ilo.user }}"
    ansible_ssh_pass: "{{ ilo.password }}"
    no_log: True
```

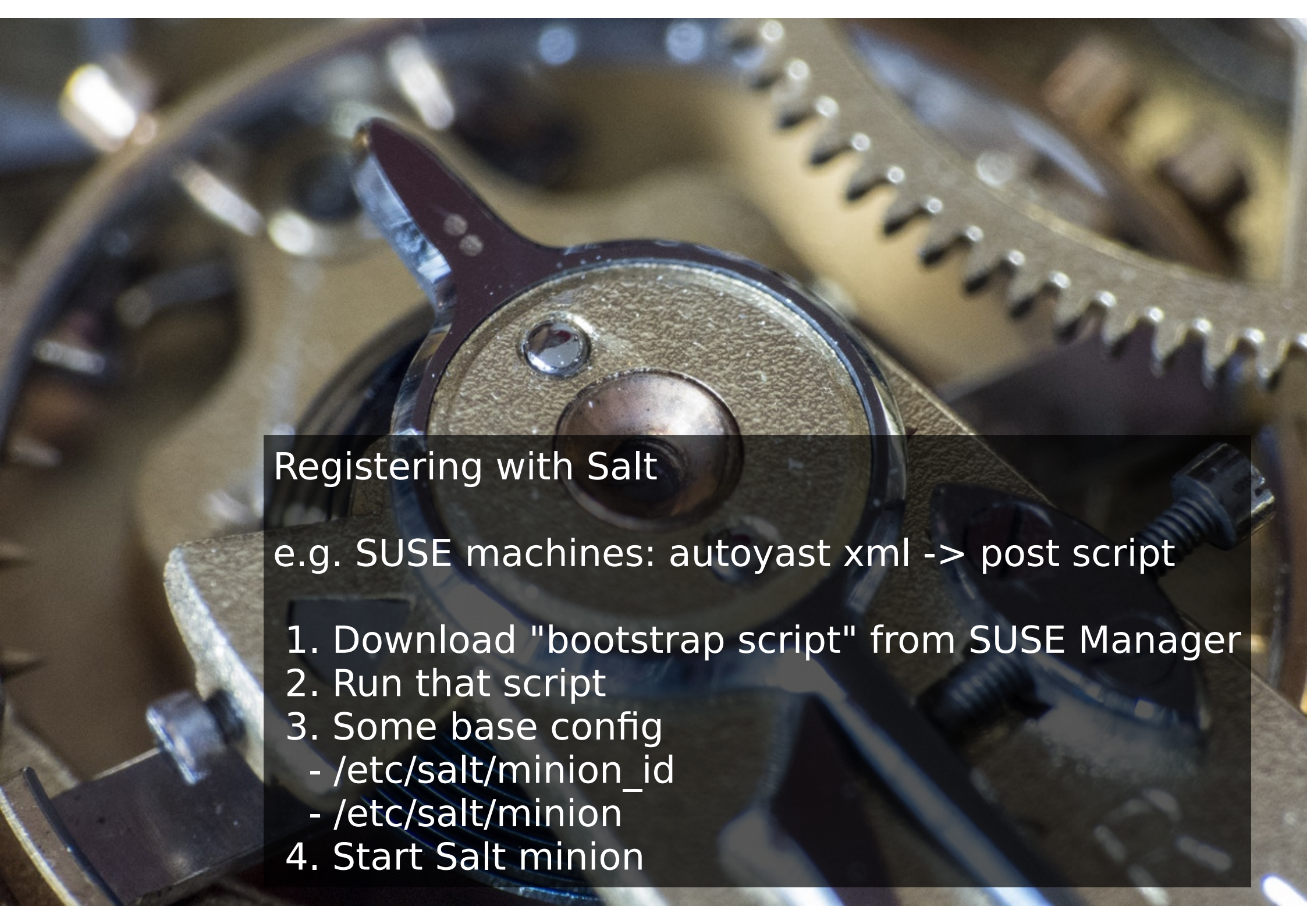



```
# hardware: make it reboot
# step 2 - send commands

- name: "install_hw: set ilo server name"
  raw: set /system1 oemHPE_server_name={{ host_name_short }}
  delegate_to: "{{ ilo_host_name }}"

- name: "install_hw: Power off system"
  shell: >
    /usr/sbin/iloorest reboot PressAndHold
    --url {{ ilo_host_name }}
    -u {{ ilo.user }} -p {{ ilo.password | quote }}

- name: "install_hw: Create Raid1 on OS disks"
  ...
- name: "install_hw: Mount boot iso image"
  ...
- name: "install_hw: Boot system"
  ...
```

Registering with Salt

e.g. SUSE machines: autoyast xml -> post script

1. Download "bootstrap script" from SUSE Manager
2. Run that script
3. Some base config
 - /etc/salt/minion_id
 - /etc/salt/minion
4. Start Salt minion

💻

Keys

?

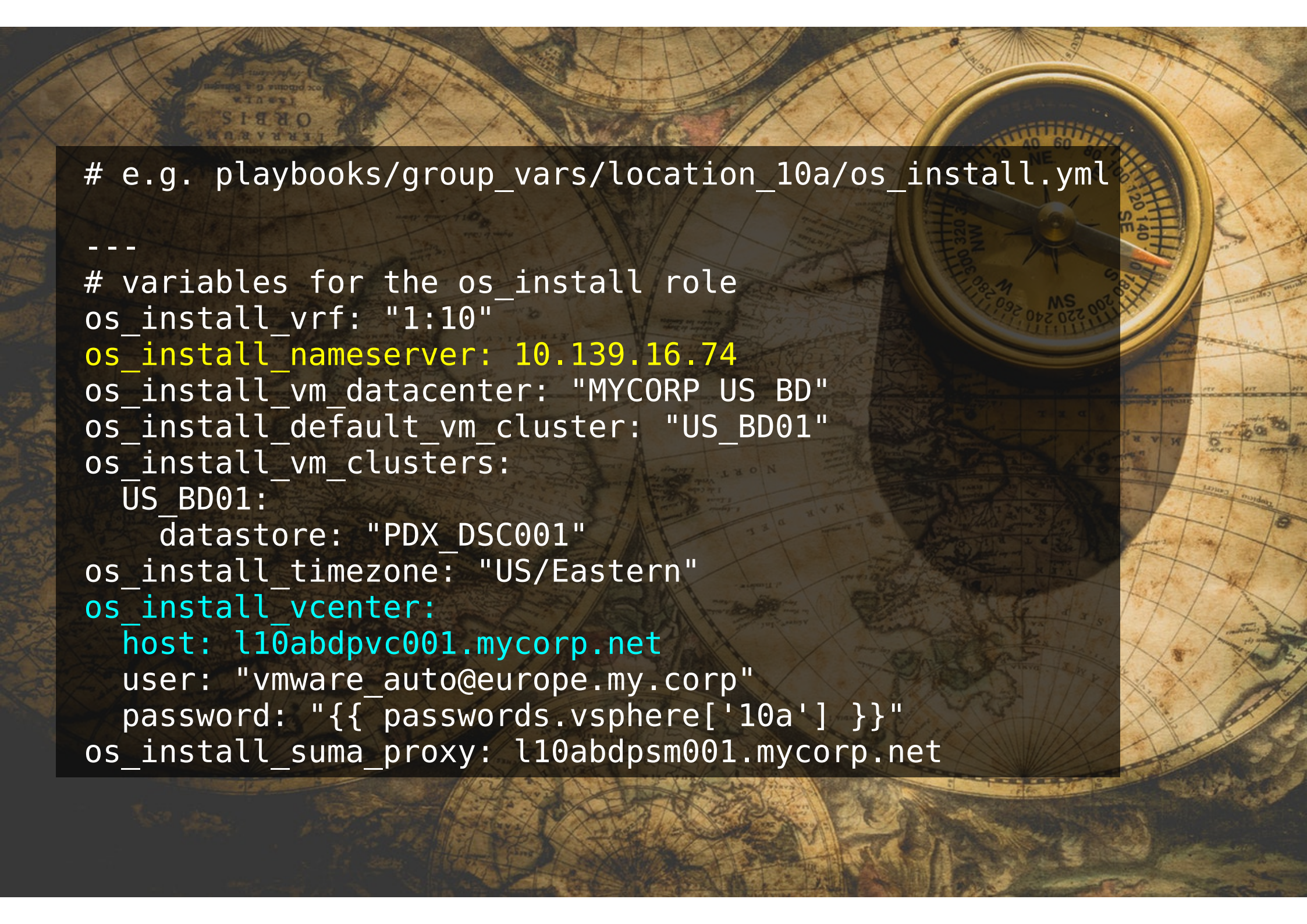
🔄

Refresh

Items 1 - 25 of 514

25 items per page

Name	Fingerprint	State	Actions
101abdpaux001.	ff:6d:f1:01:c5:da:0c:75:8c:e8:02:2b:bd:88:25:99:81:fa:e3:a8:5a:bc:b7:a2:e6:e8:ae:ae:93:87:1b:dd	accepted	🗑️
101abdpaux002.	f6:bf:23:ec:75:64:1a:f5:55:5e:19:d0:84:18:53:54:64:73:4a:ba:a3:06:94:52:39:20:20:25:2d:61:35:45	accepted	🗑️
101abdpaux003.	1b:82:0f:87:20:04:9c:53:3d:d6:65:dd:ee:0a:4d:c1:84:38:40:78:ab:9b:f1:12:2c:d6:88:43:db:b3:6d:a2	accepted	🗑️
101abdpaux004.	33:0b:06:12:38:62:24:8e:56:b5:6d:cf:a4:fc:f7:69:64:ad:ae:16:f4:59:c7:25:42:11:a0:91:ec:5a:fb:43	accepted	🗑️

The background of the image is a detailed, aged map of the world, likely from the 17th or 18th century. It features a compass rose in the upper right corner, with a needle pointing towards the top right. The map is covered in a grid of latitude and longitude lines, and various geographical features are labeled in Latin. The overall tone is sepia and historical.

```
# e.g. playbooks/group_vars/location_10a/os_install.yml
```

```
---
```

```
# variables for the os_install role
```

```
os_install_vrf: "1:10"
```

```
os_install_nameserver: 10.139.16.74
```

```
os_install_vm_datacenter: "MYCORP US BD"
```

```
os_install_default_vm_cluster: "US_BD01"
```

```
os_install_vm_clusters:
```

```
  US_BD01:
```

```
    datastore: "PDX_DSC001"
```

```
os_install_timezone: "US/Eastern"
```

```
os_install_vcenter:
```

```
  host: l10abdpvc001.mycorp.net
```

```
  user: "vmware_auto@europe.my.corp"
```

```
  password: "{{ passwords.vsphere['10a'] }}"
```

```
os_install_suma_proxy: l10abdpvm001.mycorp.net
```


Defining machines

```
# hardware
l19abdpbl004:
  machine_type: WORKER
  os: SLES12-SP2
  net: prodhdp
```

```
# virtual
l19abdpprx002:
  machine_type: VM
  os: SLES12-SP3
  net: shared
  vm_memory: 1024
  vm_cpus: 1
  vm_disksize: 40GB
```



A scenic landscape photograph of a mountain valley. In the foreground, there is a calm lake reflecting the surrounding scenery. The middle ground features a lush green valley with scattered small houses and trees. In the background, majestic, rugged mountains rise against a clear blue sky with a few wispy clouds. The overall scene is peaceful and picturesque.

Photo credits

(all photos from Pexels)

- [1] Photo by Manuel Geissinger
- [4] Photo by Skitterphoto
- [5] Photo by kelvin carris
- [6] Photo by panumas nikhomkhai
- [7] Photo by Min An
- [12] Photo by Mike
- [14] Photo by Digital Buggu
- [15] Photo by Ylanite Koppens
- [16] Photo by Christina Morillo