



This repository has been archived by the owner on Sep 15, 2023. It is now read-only.

MDO2023_INO / INO / GCL2 / KP406287 / lab8i9 / Sprawozdanie.md



qamilciaq1 KP406287

10 months ago



139 lines (71 loc) · 11.4 KB

Preview

Code

Blame

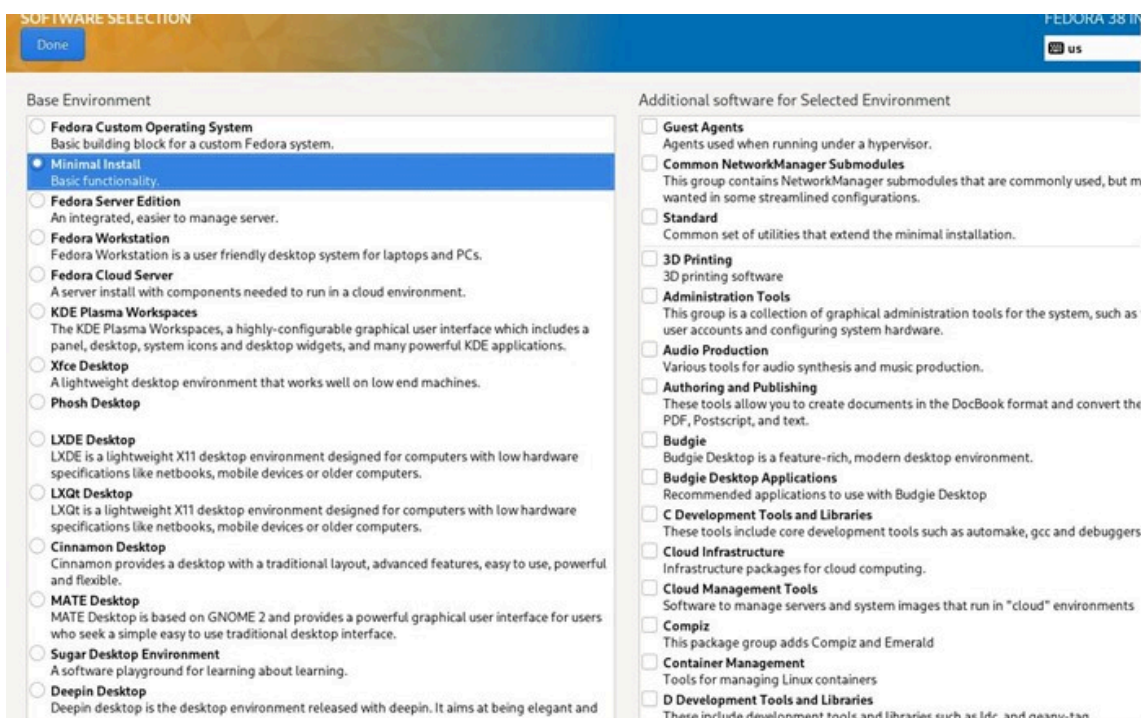
Raw



Ansible

1. Instalacja Ansible na głównej maszynie oraz na maszynie pomocniczej.
- Utworzyłam nową maszynę wirtualną z minimalną wersją oprogramowania Fedora oraz pobrałam Ansible według linku podanego w instrukcji do wykonania sprawozdania.

```
kamipa@fedora:/etc/ansible$ ansible --version
ansible [core 2.14.5]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/kamipa/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /home/kamipa/.local/lib/python3.10/site-packages/ansible
  ansible collection location = /home/kamipa/.ansible/collections:/usr/share/ansible/collections
  executable location = /home/kamipa/.local/bin/ansible
  python version = 3.10.6 (main, Mar 10 2023, 10:55:28) [GCC 11.3.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
```



PlikMaszynaWidokWejścieUrządzeniaPomoc

Done

FEDORA

US

Full name

kamila


User name

kamila

☒ Add administrative privileges to this user account (wheel group membership)

☒ Require a password to use this account

fedora
SERVER



US

Downloading 409 RPMs, 5.46 MiB / 420.01 MiB (1%) done.

fedora
SERVER



Complete!

Fedora is now successfully installed and ready for you to use!
Go ahead and reboot your system to start using it!

Reboot System

```

Total 3.5 MB/s | 52 MB 00:14
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :
Installing : python3-resolvlib-0.5.5-7.fc38.noarch 1/1
Installing : python3-pyyaml-6.0-6.fc38.x86_64 1/12
Installing : python3-ply-3.11-18.fc38.noarch 2/12
Installing : python3-pycparser-2.20-9.fc38.noarch 3/12
Installing : python3-cffi-1.15.1-4.fc38.x86_64 4/12
Installing : python3-cryptography-37.0.2-8.fc38.x86_64 5/12
Installing : python3-packaging-23.0-1.fc38.noarch 6/12
Installing : python3-markupsafe-2.1.2-1.fc38.x86_64 7/12
Installing : python3-jinja2-3.0.3-7.fc38.noarch 8/12
Installing : python3-argcomplete-2.0.8-6.fc38.noarch 9/12
Installing : ansible-core-2.14.5-1.fc38.noarch 10/12
Installing : ansible-7.5.0-1.fc38.noarch 11/12
Running scriptlet: ansible-7.5.0-1.fc38.noarch 12/12
Verifying : python3-argcomplete-2.0.8-6.fc38.noarch 1/12
Verifying : python3-cffi-1.15.1-4.fc38.x86_64 2/12
Verifying : python3-cryptography-37.0.2-8.fc38.x86_64 3/12
Verifying : python3-jinja2-3.0.3-7.fc38.noarch 4/12
Verifying : python3-markupsafe-2.1.2-1.fc38.x86_64 5/12
Verifying : python3-packaging-23.0-1.fc38.noarch 6/12
Verifying : python3-ply-3.11-18.fc38.noarch 7/12
Verifying : python3-pycparser-2.20-9.fc38.noarch 8/12
Verifying : python3-pyyaml-6.0-6.fc38.x86_64 9/12
Verifying : python3-resolvlib-0.5.5-7.fc38.noarch 10/12
Verifying : ansible-7.5.0-1.fc38.noarch 11/12
Verifying : ansible-core-2.14.5-1.fc38.noarch 12/12

Installed:
ansible-7.5.0-1.fc38.noarch      ansible-core-2.14.5-1.fc38.noarch  python3-argcomplete-2.0.8-6.fc38.noarch  python3-cffi-1.15.1-4.fc38.x86_64
python3-cryptography-37.0.2-8.fc38.x86_64  python3-jinja2-3.0.3-7.fc38.noarch  python3-markupsafe-2.1.2-1.fc38.x86_64  python3-packaging-23.0-1.fc38.noarch
python3-ply-3.11-18.fc38.noarch      python3-pycparser-2.20-9.fc38.noarch  python3-pyyaml-6.0-6.fc38.x86_64      python3-resolvlib-0.5.5-7.fc38.noarch

```

```

[kamila@fedora ~]$ ansible --version
ansible [core 2.14.5]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/kamila/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3.11/site-packages/ansible
  ansible collection location = /home/kamila/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.11.3 (main, Apr 5 2023, 00:00:00) [GCC 13.0.1 20230401 (Red Hat 13.0.1-0)] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = False
[kamila@fedora ~]$

```

- Dokonałam inwentaryzacji systemów- ustawiłam najbardziej przewidywalne nazwy, czyli na maszynie z Fedorą- fedora, na maszynie z Ubuntu- ubuntu. Najpierw ustawiłam nazwę, następnie ją zweryfikowałam za pomocą hostnamectl status.

```

kamipa@Ubuntu:~$ sudo hostnamectl set-hostname ubuntu
kamipa@Ubuntu:~$ hostnamectl status
Static hostname: ubuntu
    Icon name: computer-vm
    Chassis: vm
    Machine ID: 4a4996a0038b47e4b75b0f05c3a750e8
    Boot ID: ff0c6bd58128442b819723ecf2b9c038
    Virtualization: oracle
Operating System: Ubuntu 22.04.2 LTS
    Kernel: Linux 5.19.0-41-generic
    Architecture: x86_64
    Hardware Vendor: innotek GmbH
    Hardware Model: VirtualBox
kamipa@Ubuntu:~$

```

```
[kamila@fedora ~]$ sudo hostnamectl set-hostname fedora
[sudo] password for kamila:
[kamila@fedora ~]$ hostnamectl status
  Static hostname: fedora
          Icon name: computer-vm
          Chassis: vm
          Machine ID: 9bd18a0fc57642daaf1e1646f6ccf5ed
          Boot ID: 92f15e6ebcc44db9890f55c3450e3960
          Virtualization: oracle
          Operating System: Fedora Linux 38 (Thirty Eight)
          CPE OS Name: cpe:/o:fedoraproject:fedora:38
          OS Support End: Tue 2024-05-14
          OS Support Remaining: 11month 3w 1d
          Kernel: Linux 6.2.15-300.fc38.x86_64
          Architecture: x86-64
          Hardware Vendor: innotek GmbH
          Hardware Model: VirtualBox
          Firmware Version: VirtualBox
          Firmware Date: Fri 2006-12-01
```

- Aby umożliwić wywoływanie komputerów za pomocą nazw zamiast adresów IP, wprowadziłam nazwy DNS dla maszyn wirtualnych, korzystając z systemd-resolved. W tym celu dokonałam edycji dwóch plików /etc/hosts oraz /etc/systemd/resolved.conf. Musiałam użyć polecenia "sudo" w celu uzyskania uprawnień administratora, inaczej pojawiał się błąd związany z brakiem uprawnień do edycji pliku.

```
GNU nano 6.2 /etc/hosts
127.0.0.1    localhost
127.0.1.1    Ubuntu.myguest.virtualbox.org  Ubuntu

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters

192.168.1.106 fedora
192.168.1.127 ubuntu
```

```
#Domains=
#DNSSEC=no
#DNSOverTLS=no
#MulticastDNS=no
#LLMNR=no
#Cache=no-negative
#CacheFromLocalhost=no
#DNSStubListener=yes
#DNSStubListenerExtra=
#ReadEtcHosts=yes
#ResolveUnicastSingleLabel=no
DNS=192.168.1.106
Domains=fedora
DNS=192.168.1.127
Domains=ubuntu
```

- Zrestartowałam usługę systemd-resolved przy użyciu polecenia sudo systemctl restart systemd-resolved.
- Zweryfikowałam łączność za pomocą komendy ping i nazwy hosta. Weryfikacja łączności przebiegła pomyślnie.


```
kamipa@Ubuntu:~$ ping fedora
PING fedora (192.168.1.93) 56(84) bytes of data.
64 bytes from fedora (192.168.1.93): icmp_seq=1 ttl=64 time=3.19 ms
64 bytes from fedora (192.168.1.93): icmp_seq=2 ttl=64 time=1.73 ms
64 bytes from fedora (192.168.1.93): icmp_seq=3 ttl=64 time=2.60 ms
64 bytes from fedora (192.168.1.93): icmp_seq=4 ttl=64 time=1.34 ms
64 bytes from fedora (192.168.1.93): icmp_seq=5 ttl=64 time=1.76 ms
64 bytes from fedora (192.168.1.93): icmp_seq=6 ttl=64 time=2.62 ms
64 bytes from fedora (192.168.1.93): icmp_seq=7 ttl=64 time=3.75 ms
64 bytes from fedora (192.168.1.93): icmp_seq=8 ttl=64 time=2.65 ms
64 bytes from fedora (192.168.1.93): icmp_seq=9 ttl=64 time=2.75 ms
64 bytes from fedora (192.168.1.93): icmp_seq=10 ttl=64 time=4.19 ms
64 bytes from fedora (192.168.1.93): icmp_seq=11 ttl=64 time=3.06 ms
```

```
[kamila@kamila1 ~]$ ping ubuntu
PING ubuntu (192.168.1.120) 56(84) bytes of data.
64 bytes from ubuntu (192.168.1.120): icmp_seq=1 ttl=64 time=4.08 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=2 ttl=64 time=1.38 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=3 ttl=64 time=1.56 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=4 ttl=64 time=2.96 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=5 ttl=64 time=0.964 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=6 ttl=64 time=1.28 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=7 ttl=64 time=3.08 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=8 ttl=64 time=1.17 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=9 ttl=64 time=3.32 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=10 ttl=64 time=2.84 ms
64 bytes from ubuntu (192.168.1.120): icmp_seq=11 ttl=64 time=1.36 ms
```

- W celu uzyskania łączności między maszynami wygenerowałam klucz i wymieniłam go pomiędzy maszynami.

Próba zalogowania przy użyciu ssh powiodła się.

```
kamipa@Ubuntu:~$ ssh-copy-id -i ~/.ssh/id_ed25519 kamila@192.168.1.95
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/kamipa/.ssh/id_ed25519.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'kamila@192.168.1.95'"
and check to make sure that only the key(s) you wanted were added.

kamipa@Ubuntu:~$ ssh kamila@192.168.1.95
Web console: https://kamila1:9090/ or https://192.168.1.94:9090/

Last login: Fri May 19 08:43:59 2023 from 192.168.1.120
```

- Dodałam nazwy moich maszyn do pliku /etc/ansible/hosts.

Następnie wyświetliłam listę dostępnych hostów.

```
kamipa@fedora:/etc/ansible$ ansible all --list-hosts
hosts (2):
  fedora
  ubuntu
```

- Stworzyłam plik inventory.yaml do użytku za pomocą Ansible. Określiłam w nim orchestrators (hosty) oraz endpoints (czyli miejsce docelowe wykonujące zleconą pracę).

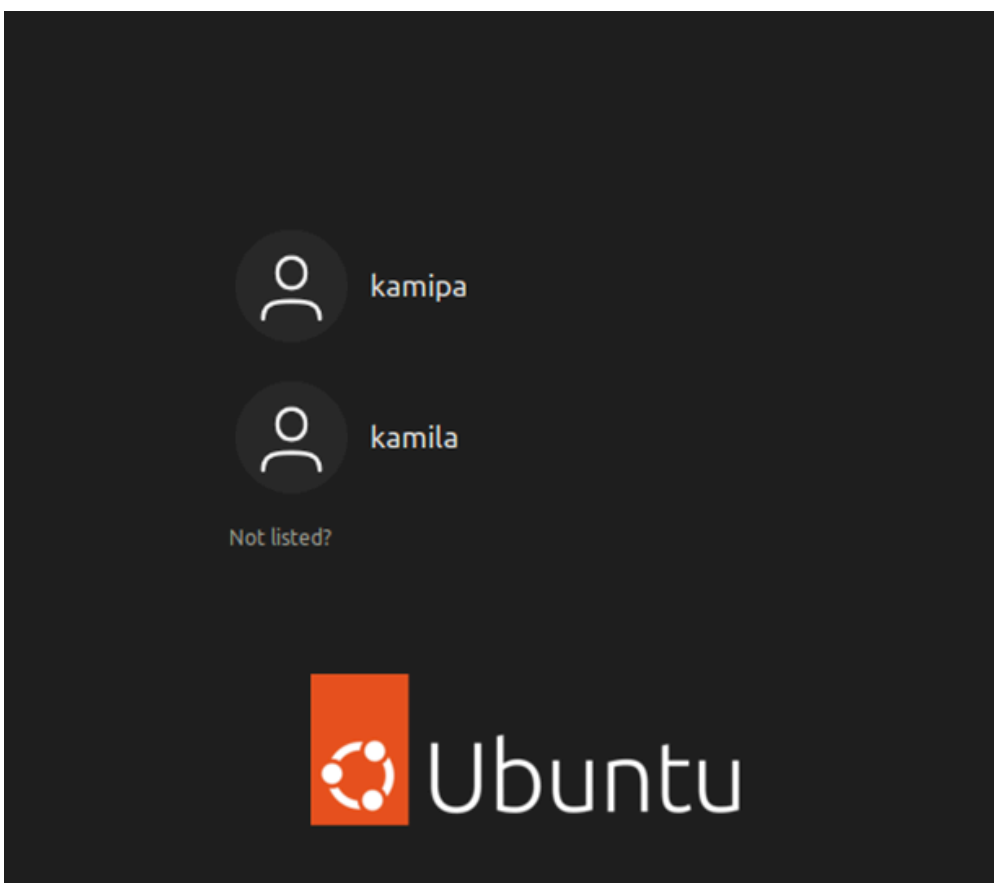
```
GNU nano 6.2 inventory.yaml
orchestrators:
  hosts:
    central-node:
      ansible_host: ubuntu

endpoints:
  hosts:
    node1:
      ansible_host: fedora
```

Podczas wykonywania laboratorium zawiesił mi się komputer, więc po uruchomieniu musiałam od początku wykonać wszystkie poprzednie kroki, ze względu na to, że zmienił się u mnie adres ip (dynamiczne przydzielanie).

Tym razem, aby uniknąć tego typu sytuacji kolejnym razem ustawiłam adres ip na stały na ubuntu oraz fedorze. W skrócie, na ubuntu weszłam w sekcję "Ustawienia sieciowe", następnie w oknie konfiguracji interfejsu sieciowego wybrałam kartę "IPv4", wprowadziłam adres IP, maskę podsieci i bramę domyślną. Aby zrobić to samo na fedorze z minimalną instalacją wpisałam polecenie **sudo vi /etc/sysconfig/network-scripts/ifcfg-enp0s3**. Następnie w pliku konfiguracyjnym interfejsu znalazłam linie dotyczące protokołu IPv4. Zmieniłam BOOTPROTO na none, aby wyłączyć dynamiczne przydzielanie adresu IP przez DHCP. Dodałam linie dla adresu IP, maski podsieci, bramy domyślnej i serwerów DNS. Po zatwierdzeniu zmian wpisałam **sudo systemctl restart network**, aby ponownie uruchomić usługę sieciową i załadować nową konfigurację. Opisany przeze mnie proces to dodatkowo wykonany krok pomocniczy, aby ułatwić sobie pracę.

Tę samą pracę po restarcie komputera (czyli wszystkie poprzednie kroki wykonałam raz jeszcze na swoim 2 koncie- kamila)



```
kamila@ubuntu:/etc/ansible$ ansible-inventory -i inventory.yaml --list
{
  "_meta": {
    "hostvars": {
      "central-node": {
        "ansible_host": "ubuntu"
      },
      "node1": {
        "ansible_host": "fedora"
      }
    }
  },
  "all": {
    "children": [
      "ungrouped",
      "orchestrators",
      "endpoints"
    ]
  },
  "endpoints": {
    "hosts": [
      "node1"
    ]
  },
  "orchestrators": {
    "hosts": [
      "central-node"
    ]
  }
}
```

- Uruchomienie pingowania dla wszystkich dostępnych hostów oraz endpoints przez Ansible.

Polecenie `ansible --inventory inventory.yaml all -m ping` jest używane w narzędziu Ansible do przeprowadzenia testu ping na wszystkich hostach zdefiniowanych w pliku inwentarza (inventory.yaml).

Polecenie uruchamia test ping na wszystkich hostach zdefiniowanych w pliku inwentarza, aby sprawdzić, czy są one dostępne i odpowiadają na żądania. Jest to jedno z podstawowych poleceń Ansible, które można używać do weryfikacji łączności i dostępności hostów w infrastrukturze.

```
kamila@ubuntu:/etc/ansible$ ansible --inventory inventory.yaml all -m ping
central-node | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: kamila@ubuntu: Permission denied (publickey,password).",
  "unreachable": true
}
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

```
kamila@ubuntu:/etc/ansible$ ansible --inventory inventory.yaml endpoints -m ping
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

- Skopiowanie pliku inwentaryzacji na maszynę endpoint, następnie wykonanie tej samej komendy w celu sprawdzenia czy plik został zmieniony.


```
kamila@ubuntu:/etc/ansible$ ansible --inventory inventory.yaml endpoints -m copy -a "src=inventory.yaml dest=/home/kamila/inventory.yaml"
node1 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": true,
  "checksum": "e36cd2f411f6f704729aa448fd8dde8e6acf6e10",
  "dest": "/home/kamila/inventory.yaml",
  "gid": 1000,
  "group": "kamila",
  "md5sum": "cf22a59e1c04f11360e659826f9983dd",
  "mode": "0644",
  "owner": "kamila",
  "secontext": "unconfined_u:object_r:user_home_t:s0",
  "size": 128,
  "src": "/home/kamila/.ansible/tmp/ansible-tmp-1684595154.4116447-15073-133240462334613/source",
  "state": "file",
  "uid": 1000
}
```

Plik został poprawnie skopiowany na mojego endpointa:

```
[kamila@fedora ~]$ ls
inventory.yaml
[kamila@fedora ~]$ cat inventory.yaml
orchestrators:
  hosts:
    central-node:
      ansible_host: ubuntu

endpoints:
  hosts:
    node1:
      ansible_host: fedora
```

Wpisałam ponownie tę samą komendę oraz uzyskałam status SUCCESS zamiast CHANGED. Sukces oznacza, że operacja zakończyła się pomyślnie, natomiast zmiana (CHANGED) oznacza, że operacja wprowadziła jakieś zmiany na docelowym systemie. W moim przypadku, podczas pierwszego kopiowania pliku, nastąpiła zmiana, a w drugim podejściu, gdy plik już istniał, nie było zmian, co skutkowało wynikiem SUCCESS.

```
kamila@ubuntu:/etc/ansible$ ansible --inventory inventory.yaml endpoints -m copy -a "src=inventory.yaml dest=/home/kamila/inventory.yaml"
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "checksum": "e36cd2f411f6f704729aa448fd8dde8e6acf6e10",
  "dest": "/home/kamila/inventory.yaml",
  "gid": 1000,
  "group": "kamila",
  "mode": "0644",
  "owner": "kamila",
  "path": "/home/kamila/inventory.yaml",
  "secontext": "unconfined_u:object_r:user_home_t:s0",
  "size": 128,
  "state": "file",
  "uid": 1000
}
```

- Próba wykonania powyższej komendy przy wyłączonym internecie oraz usłudze ssh


```
[kamila@fedora ~]$ systemctl stop sshd
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to stop 'sshd.service'.
Authenticating as: kamila
Password:
==== AUTHENTICATION COMPLETE ====
[kamila@fedora ~]$ systemctl status sshd
■ sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Drop-In: /usr/lib/systemd/system/service.d
           └─10-timeout-abort.conf
   Active: inactive (dead) since Sat 2023-05-20 07:53:56 EDT; 9s ago
   Duration: 14h 4min 41.440s
   Docs: man:sshd(8)
         man:sshd_config(5)
   Process: 650 ExecStart=/usr/sbin/sshd -D $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 650 (code=exited, status=0/SUCCESS)
   CPU: 7.387s
```

```
kamila@ubuntu:/etc/ansible$ ansible endpoints -m ping -i inventory.yaml
node1 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: ssh: connect to host fedora
port 22: Connection refused",
  "unreachable": true
}
```

- Stworzyłam plik playbook.yaml zawierający kroki budujące kontenera.

```
- name: Deploy application in Docker container
  hosts: endpoints
  become: yes
  become_method: sudo
  tasks:
    - name: Install docker
      become: yes
      pip:
        name:
          - websocket-client==0.59.0
          - requests==2.28.2
          - urllib3==1.26.15
          - docker
          - docker-compose
        state: present
    - name: First test
      shell: docker ps
    - name: Build application
      docker_image:
        source: build
        build:
          path: /home/kamila/dockerfiles
          name: app
    - name: Run container
      docker_container:
        name: app_container
        image: app
```

```
dockerfiles inventory.yaml playbook.yaml playbook.yaml.save
[kamila@fedora ~]$ ansible-playbook -i inventory.yaml playbook.yaml -K
BECOME password:

PLAY [Deploy application in Docker container]
TASK [Gathering Facts]
ok: [node1]
TASK [Install docker]
ok: [node1]
TASK [First test]
changed: [node1]
TASK [Build application]
changed: [node1]
TASK [Run container]
changed: [node1]
PLAY RECAP
node1 : ok=5 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

- Wersja z obrazem z DockerHub:

```

- name: Deploy application in Docker container 2
  hosts: endpoints
  become: yes
  become_method: sudo
  tasks:
    - name: Install docker
      become: yes
      pip:
        name:
          - websocket-client==0.59.0
          - requests==2.28.2
          - urllib3==1.26.15
          - docker
          - docker-compose
        state: present
    - name: First test
      shell: docker ps
    - name: Build application
      docker_image:
        source: pull
        name: kamila/moja_appka
    - name: Volume
      docker_volume:
        name: volume
    - name: Run docker container
      docker_container:
        name: app
        image: kamila/moja_appka
        interactive: true
        ports:
          - "8081:8081"
        volumes:
          - /volume

```

```

(kamila@fedora ~)$ ansible-playbook -i inventory.yml playbook2.yml -K
BECOME password:

PLAY [Deploy application in Docker container 2] *****

TASK [Gathering Facts] *****
ok: [node1]

TASK [Install docker] *****
ok: [node1]

TASK [First test] *****
changed: [node1]

TASK [Build application] *****
changed: [node1]

TASK [Volume] *****
changed: [node1]

TASK [Run docker container] *****
changed: [node1]

PLAY RECAP *****
node1 : ok=6  changed=4  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

```

- Zatrzymanie oraz usunięcie

```

- name: Deploy application in Docker container 2
  hosts: endpoints
  become: yes
  become_method: sudo
  tasks:
    - name: Install docker
      become: yes
      pip:
        name:
          - websocket-client==0.59.0
          - requests==2.28.2
          - urllib3==1.26.15
          - docker
          - docker-compose
        state: present
    - name: First test
      shell: docker ps
    - name: Build application
      docker_image:
        source: pull
        name: kamila/moja_appka
    - name: Volume
      docker_volume:
        name: volume
    - name: Run docker container
      docker_container:
        name: app
        image: kamila/moja_appka
        interactive: true
        ports:
          - "8081:8081"
        volumes:
          - /volume
    - name: Stop
      docker_container:
        name: app
        state: stopped
    - name: Remove
      docker_container:
        name: app
        state: absent

```

```

[kamila@fedora ~]$ ansible-playbook -i inventory.yaml playbook2.yaml -K
BECOME password:

PLAY [Deploy application in Docker container 2] *****

TASK [Gathering Facts] *****
ok: [node1]

TASK [Install docker] *****
ok: [node1]

TASK [First test] *****
changed: [node1]

TASK [Build application] *****
ok: [node1]

TASK [Volume] *****
ok: [node1]

TASK [Run docker container] *****
ok: [node1]

TASK [Stop] *****
changed: [node1]

TASK [Remove] *****
changed: [node1]

PLAY RECAP *****
node1                : ok=0    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

Fedora

- Aby przeprowadzić instalację nienadzorowaną, musiałam skopiować plik odpowiedzi o nazwie /root/anaconda-ks.cfg. Ten plik był potrzebny do skonfigurowania instalacji. Jednak, z powodu ograniczeń dotyczących uprawnień, nie mogłam bezpośrednio skopiować pliku z folderu /root. W celu rozwiązania tego problemu, skorzystałam z wcześniej utworzonej maszyny. Z tej maszyny mogłam uzyskać dostęp do pliku odpowiedzi. Następnie skopiowałam ten plik z maszyny do docelowej lokalizacji na innym urządzeniu, aby umożliwić instalację nienadzorowaną. W rezultacie, mogłam pomyślnie skopiować plik odpowiedzi /root/anaconda-ks.cfg.

```

[kamila@fedora ~]$ sudo cp /root/anaconda-ks.cfg .
[sudo] password for kamila:
[kamila@fedora ~]$ ls
anaconda-ks.cfg  dockerfiles  inventory.yaml  playbook2.yaml  playbook.yaml  playbook.yaml.save
[kamila@fedora ~]$

```

- Nadałam uprawnienia plikowi anaconda-ks.cfg.

```

[kamila@fedora ~]$ ls
anaconda-ks.cfg  dockerfiles  inventory.yaml  playbook2.yaml  playbook.yaml  playbook.yaml.save
[kamila@fedora ~]$ sudo chmod 777 anaconda-ks.cfg
[kamila@fedora ~]$ ls -la
total 44
drwx-----. 6 kamila kamila 4096 May 21 13:09 .
drwxr-xr-x. 3 root root 20 May 16 12:18 ..
-rw-rw-rw-. 1 root root 762 May 21 13:09 anaconda-ks.cfg
drwxr-xr-x. 4 kamila kamila 27 May 21 07:59 .ansible
-rw-----. 1 kamila kamila 606 May 20 06:44 .bash_history
-rw-r--r--. 1 kamila kamila 18 Feb 5 19:00 .bash_logout
-rw-r--r--. 1 kamila kamila 141 Feb 5 19:00 .bash_profile
-rw-r--r--. 1 kamila kamila 492 Feb 5 19:00 .bashrc
drwxr-xr-x. 2 kamila kamila 24 May 21 07:51 dockerfiles
-rw-r--r--. 1 kamila kamila 128 May 20 07:28 inventory.yaml
-rw-----. 1 kamila kamila 20 May 20 07:57 .lessht
drwxr-xr-x. 3 kamila kamila 19 May 21 08:59 .npm
-rw-r--r--. 1 root root 932 May 21 12:43 playbook2.yaml
-rw-r--r--. 1 root root 634 May 21 09:18 playbook.yaml
-rw-r--r--. 1 root root 695 May 20 18:57 playbook.yaml.save
drwx-----. 2 kamila kamila 143 May 20 07:18 .ssh

```

```

GNU nano 7.2 /root/anaconda-ks.cfg
# Generated by Anaconda 30.23.4
# Generated by pykickstart v3.47
# version=F20
# Use graphical install
graphical

# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
lang en_US.UTF-8

#packages
@minimal-environment

end

# Run the Setup Agent on first boot
firstboot --enable

# Generated using Blivet version 3.7.1
ignoredisk --only-use=sda
autopart
# Partition clearing information
clearpart --none --initlabel

# System timezone
timezone America/New_York --utc

# Root password
rootpw --iscrypted $y$j9T$ibEEMu5KCB2wCAd8enLhZEL$WjJJj8F9x/1QU8b1GJTKr0trzkmt/RLkEufk1P8aLw4
user --groups=wheel --name=kamila --password=$y$j9T$7pHfWchBCKDM.KpSquZscQZ$KCPeaf3NqJfjvwuFZJ1F8sCcoEr8FCYpM40R0Y3de/ --iscrypted --gecos="kamila"

```

- Skopiowałam plik na drugą maszynę.

```

kamila@ubuntu:~$ scp fedora:anaconda-ks.cfg .
anaconda-ks.cfg 100% 762 209.3KB/s 00:00

```

- Utworzyłam nową maszynę wirtualną z systemem Fedora.



- Edycja pliku konfiguracyjnego, który wrzuciłam na repozytorium github oraz rozpoczęcie instalacji.

```
OK | Finished dracut-initqueue.service - dracut initqueue hook.
OK | Reached target remote-fs-pre.target - Preparation for Remote File Systems.
OK | Reached target remote-cryptsetup.target - Remote Encrypted Volumes.
OK | Reached target remote-fs.target - Remote File Systems.
    Starting dracut-pre-mount.service - dracut pre-mount hook...
OK | Finished dracut-pre-mount.service - dracut pre-mount hook.
OK | Reached target initrd-root-fs.target - Initrd Root File System.
    Starting initrd-parse-etc.service - Mountpoints Configured in the Real Root...
    Stopping multipathd.service - Device-Mapper Multipath Device Controller...
OK | Finished initrd-parse-etc.service - Mountpoints Configured in the Real Root.
OK | Stopped multipathd.service - Device-Mapper Multipath Device Controller.
OK | Reached target initrd-fs.target - Initrd File Systems.
OK | Reached target initrd.target - Initrd Default Target.
    Starting dracut-mount.service - dracut mount hook...
30.531657] dracut-mount[1364]: mount: /sysroot: special device LiveOS_rootfs does not exist.
30.534823] dracut-mount[1364]: dmesg(1) may have more information after failed mount system call
OK | Finished dracut-mount.service - dracut mount hook.
    Starting dracut-pre-pivot.service - dracut pre-pivot and cleanup hook...
OK | Finished dracut-pre-pivot.service - dracut pre-pivot and cleanup hook.
    Starting initrd-cleanup.service - Cleaning Up and Shutting Down Daemons...
OK | Finished initrd-cleanup.service - Cleaning Up and Shutting Down Daemons.
OK | Stopped target network.target - Network.
OK | Stopped target remote-cryptsetup.target - Remote Encrypted Volumes.
OK | Stopped target timers.target - Timer Units.
OK | Closed dbus.socket - D-Bus System Message Bus Socket.
OK | Stopped dracut-pre-pivot.service - dracut pre-pivot and cleanup hook.
OK | Stopped target initrd.target - Initrd Default Target.
OK | Stopped target basic.target - Basic System.
OK | Stopped target initrd-root-device.target - Initrd Root Device.
OK | Stopped target initrd-usr-fs.target - Initrd /usr File System.
OK | Stopped target paths.target - Path Units.
OK | Stopped target remote-fs.target - Remote File Systems.
OK | Stopped target remote-fs-pre.target - Preparation for Remote File Systems.
OK | Stopped target slices.target - Slice Units.
OK | Stopped target sockets.target - Socket Units.
OK | Stopped target sysinit.target - System Initialization.
OK | Stopped target local-fs.target - Local File Systems.
OK | Stopped target local-fs-pre.target - Preparation for Local File Systems.
OK | Stopped target swap.target - Swaps.
OK | Closed iscsid.socket - Open-iSCSI iscsid Socket.
OK | Closed iscsiui.socket - Open-iSCSI iscsiui Socket.
OK | Closed multipathd.socket - multipathd control socket.
OK | Stopped dracut-mount.service - dracut mount hook.
OK | Stopped dracut-pre-mount.service - dracut pre-mount hook.
OK | Stopped target cryptsetup.target - Local Encrypted Volumes.
OK | Stopped dracut-initqueue.service - dracut initqueue hook.
    Starting plymouth-switch-root.service - Plymouth switch root service...
OK | Stopped systemd-sysctl.service - Apply Kernel Variables.
OK | Stopped systemd-modules-load.service - Load Kernel Modules.
```

- Aby potwierdzić, czy instalacja przebiegła pomyślnie na nowo utworzonej maszynie sprawdzam działające kontenery.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
139b671db351	app	"docker-entrypoint.s..."	4 hours ago	Exited (0) 3 hours ago	0.0.0.0:8081->8081/tcp, :::8081->8081/tcp

Wnioski:

Ansible okazał się być przydatnym narzędziem, które znacząco ułatwia operacje między dwoma komputerami, zwłaszcza w przypadku pracy w systemie bez interfejsu graficznego. Chociaż proste operacje, takie jak przesyłanie plików, mogą wydawać się wygodniejsze przy użyciu SCP, to Ansible pokazuje swoje ogromne możliwości w bardziej zaawansowanych krokach operacji między maszynami. Dzięki Ansible możemy zdefiniować zestaw instrukcji i skryptów w pliku konfiguracyjnym, który automatycznie wykonuje wszystkie niezbędne operacje. Na przykład, zamiast ręcznie instalować Dockera, pobierać obrazy i uruchamiać je na wielu maszynach, możemy skonfigurować zadania w Ansible, które wykonają te czynności za nas. Możemy zdefiniować rolę Ansible, która obejmuje wszystkie niezbędne kroki, takie jak instalacja Dockera, pobieranie obrazów i uruchamianie ich na odpowiednich maszynach. Następnie, za pomocą jednego polecenia, możemy uruchomić tę rolę na wszystkich docelowych maszynach, co automatyzuje cały proces. Potencjał Ansible leży w jego zdolności do automatyzacji skomplikowanych operacji między komputerami. Jest to nie tylko efektywne, ale również eliminuje błędy ludzkie, które mogą wystąpić podczas ręcznego wykonywania wielu kroków. Ansible umożliwia replikację i skalowanie zadań na wiele maszyn jednocześnie, co czyni go niezwykle przydatnym narzędziem w środowiskach, w których konieczne jest zarządzanie infrastrukturą wielu serwerów.