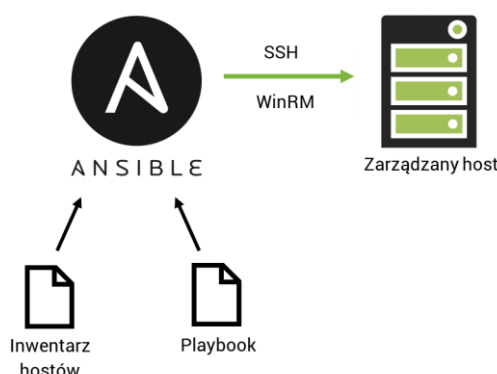


I. Wprowadzenie

Zadania do wykonania znajdują się w rozdziale IV

Ansible to oprogramowanie open-source, które wykorzystuje rozproszoną architekturę bezagentową, co oznacza, że nie wymaga instalacji dodatkowego oprogramowania na zarządzanych maszynach. Jest używane m.in. do zarządzania infrastrukturą, automatyzacji konfiguracji oraz wdrażania aplikacji w ramach praktyk DevOps. Ansible działa na systemach uniksowych i uniksopodobnych oraz może zarządzać maszynami z systemami Windows. Komunikacja z zarządzanymi jednostkami odbywa się zdalnie za pomocą protokołu SSH dla systemów Unix lub WinRM do uruchamiania poleceń PowerShell na systemach Windows.



Ansible został napisany w Pythonie, co pozwala na tworzenie własnych rozszerzeń w tym języku. Narzędzie korzysta również z silnika szablonów Jinja2, umożliwiającego dynamiczne generowanie konfiguracji. Orkiestracja zadań w Ansible odbywa się za pomocą tzw. playbooków, czyli scenariuszy definiowanych w języku YAML. YAML to uniwersalny format do reprezentowania struktur danych, często wykorzystywany do serializacji danych.

II. Ansible inventory – inwentarz hostów

Ansible działa w tym samym czasie na wielu zarządzanych hostach (węzłach/maszynach/serwerach) zdefiniowanych za pomocą pliku/plików inwentarza (ang. *inventory*). Inwentarze pozwalają na grupowanie zarządzanych hostów oraz definiowanie zmiennych wykorzystywanych później w scenariuszach lub komendach wykonywanych z poziomu konsoli.

Domyślną lokalizacją dla `inventory` jest plik `/etc/ansible/hosts`. Można wskazać inny plik inwentarza w wierszu poleceń ustawiając parametr `-i <ścieżka>`.

Przykładowa zawartość podstawowego pliku inwentarza `hosts` może wyglądać następująco:

```
[all:vars]
ansible_user=root
[server1]
192.168.0.1
[server2]
192.168.0.2
```

Istnieją dwie domyślne grupy hostów: `all` oraz `ungrouped`. Grupa `all` zawiera wszystkie hosty, a grupa `ungrouped` – wszystkie hosty, które nie należą do innej grupy niż `all`. Oznacza to, że każdy host zawsze będzie należał do co najmniej dwóch grup (`all` i `ungrouped` lub `all` i inna grupa). Chociaż grupy `all` i `ungrouped` są zawsze obecne, mogą być niejawne i nie pojawiać się podczas listowania nazw wszystkich istniejących grup.

Przypisanie zmiennych do grupy:

```
[nazwa_grupy]
192.168.10.101
192.168.10.102
```

III. Ansible modules – wbudowane funkcje Ansible

Moduły są jednostkami kodu implementującymi funkcje Ansible, które mogą być użyte z poziomu wiersza poleceń (ad hoc) lub w ramach scenariuszy.

Wywołanie konkretnego modułu Ansible ad hoc odbywa się wg następującej składni:

```
ansible <zakres> -m <nazwa_modułu> -a <argumenty_modułu>
```

Przykłady uruchomienia modułów:

Weryfikacja komunikacji pomiędzy maszynami za pomocą modułu `ping`:

```
ansible all -m ping
```

```
osboxes@osboxes:~/Desktop$ ansible all -m ping
192.168.10.102 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
192.168.10.101 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

Skopiowanie pliku o podanej nazwie na wszystkie zarządzane maszyny za pomocą modułu copy:

```
ansible all -m copy -a 'src=plik.txt dest=/home/dev'
```

Wykonanie komendy powłoki za pomocą modułu shell (wyświetlenie informacji o zasobach pamięciowych komendą free):

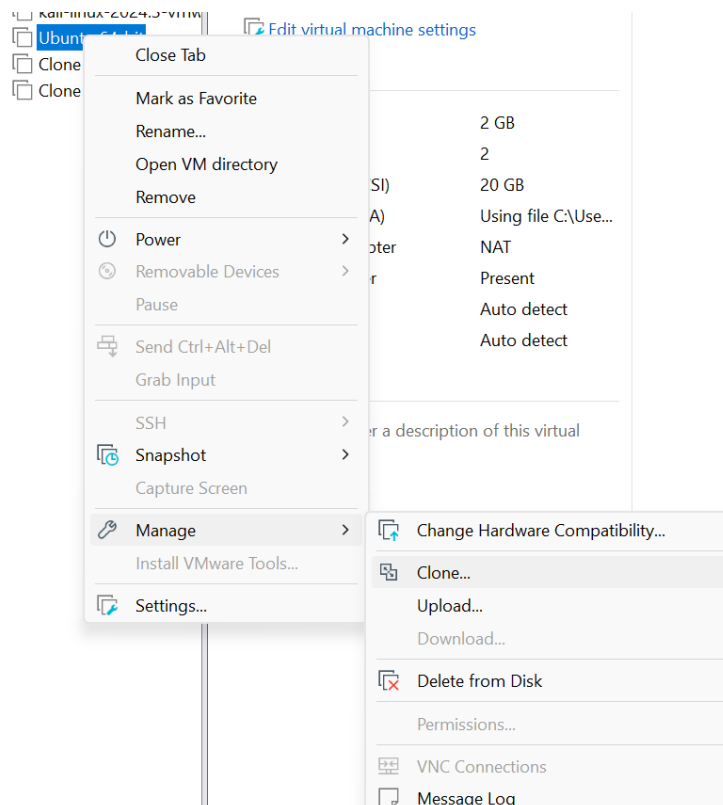
```
ansible all -m shell -a 'free -m'
```

Przykładowe wykonanie polecenia shell:

```
ubuntu@ubuntu2004:~/Desktop$ ansible all -m shell -a 'ls'
192.168.10.101 | CHANGED | rc=0 >>
Desktop
Documents
Downloads
Music
Pictures
Public
PycharmProjects
snap
Templates
Videos
```

IV. Zadania do wykonania

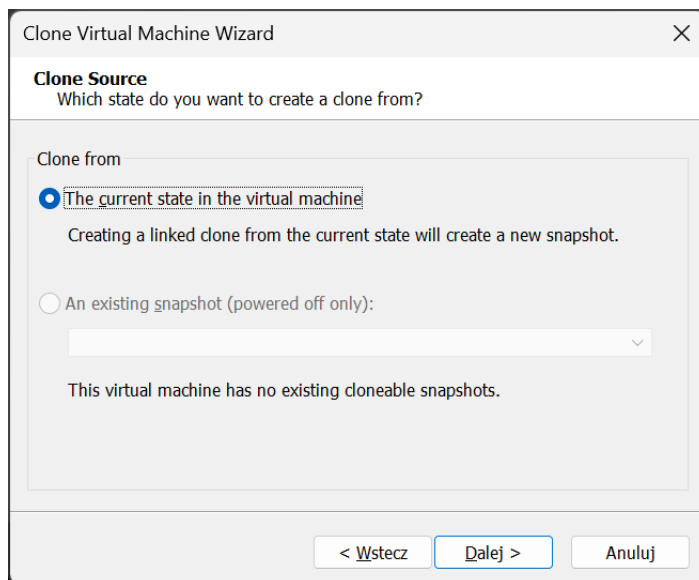
1. Utwórz maszynę wirtualną z systemem Debian.
2. Sklonuj maszynę – naciśnij prawy klawisz myszy na maszynę, a następnie wybierz opcję Manage -> Clone....



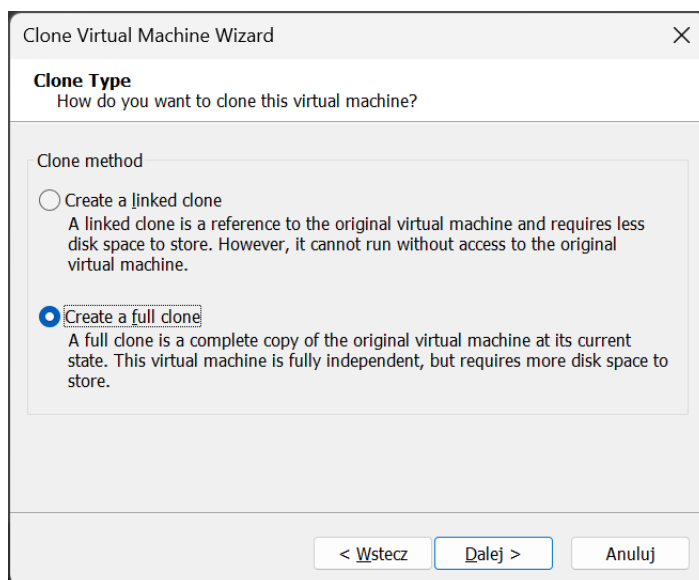
3. Naciśnij przycisk Dalej >.



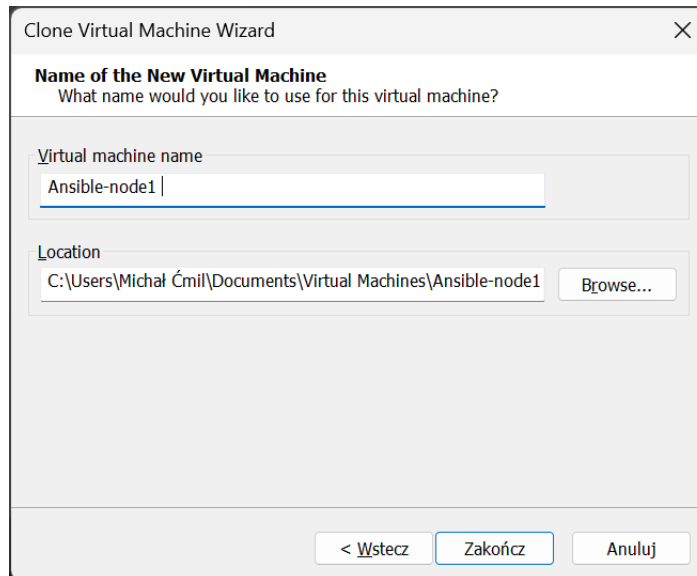
4. Naciśnij przycisk Dalej >.



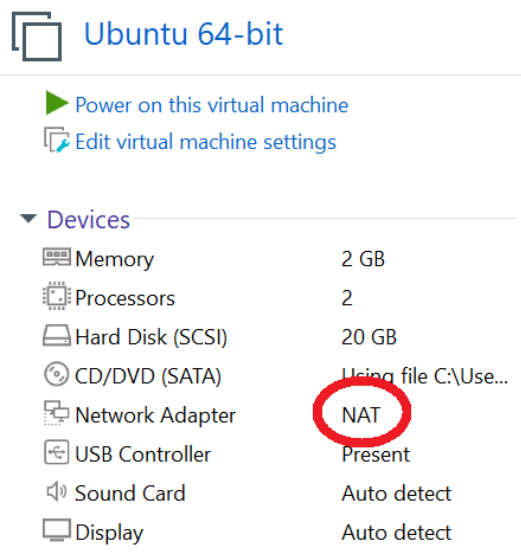
5. Przeczytaj oba opisy metod klonowania, a następnie wybierz Create a full clone. Następnie naciśnij przycisk Dalej >.



6. Nazwij maszynę **Ansible-node1**, a następnie naciśnij przycisk **Zakończ**.

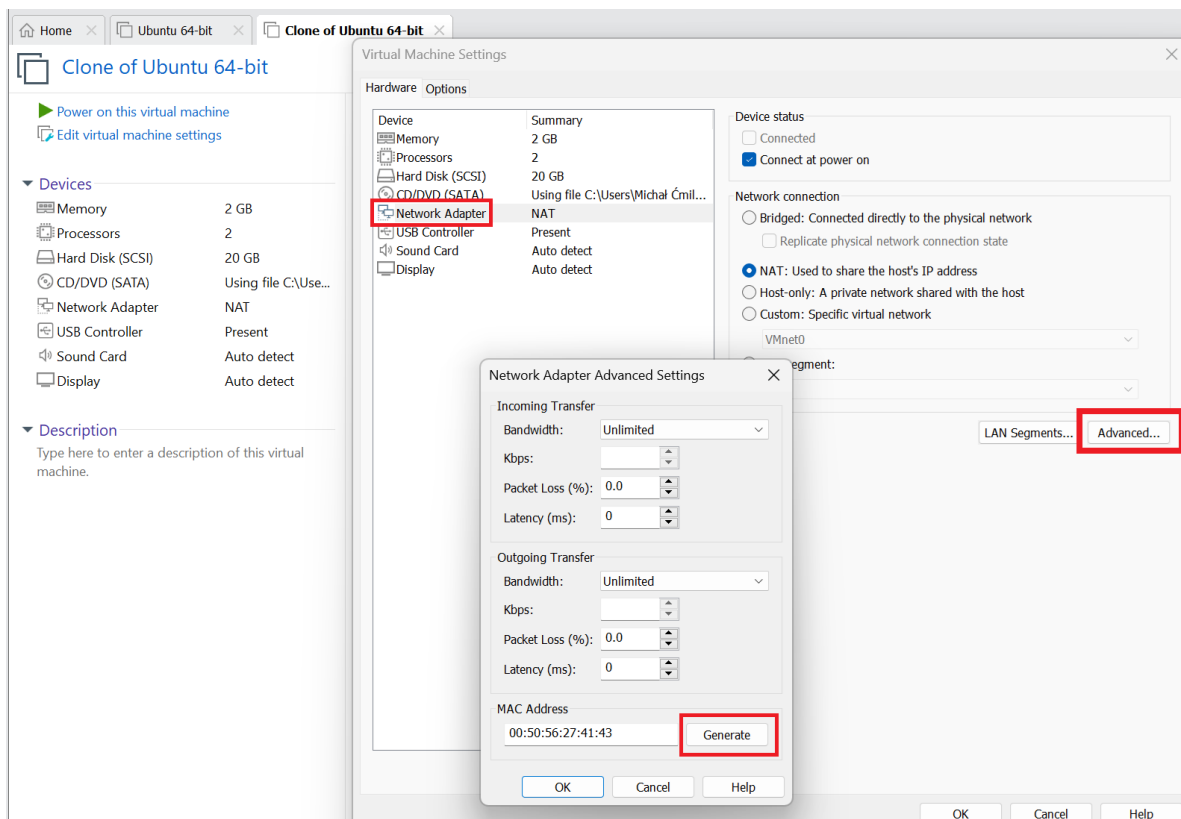


7. Analogicznie stwórz jeszcze jedną kopię, lecz tym razem nazwij ją **Ansible-node2**.
8. Sprawdź ustawienia kart sieciowych maszyn wirtualnych i upewnij się, że są skonfigurowane w trybie NAT.



9. Uruchom wszystkie 3 maszyny wirtualne.
10. Sprawdź adresy IP maszyn, a następnie zweryfikuj połączenie między nimi, sprawdzając, czy mogą się wzajemnie pingować.

Może się zdarzyć, że sklonowane maszyny będą miały te same adresy IP, co maszyna główna. Wynika to z faktu, że VMware sklonował również adres MAC. Taka sytuacja jest niedopuszczalna, ponieważ powoduje konflikt adresów. Aby zmienić adres MAC, należy wyłączyć daną maszynę wirtualną, a następnie przejść do **Network Adapter** → **Advanced...** → **Generate**. Po wygenerowaniu nowego adresu MAC zatwierdź zmiany przyciskiem **OK** i ponownie włącz maszynę.



11. Przejdź teraz do maszyny głównej, która będzie stanowić serwer zarządzający całą infrastrukturą.

12. Zaktualizuj repozytoria oraz zainstaluj wymagane dodatkowe oprogramowanie:

```
sudo apt update
sudo apt install sshpass
```

13. Zainstaluj Ansible (tylko maszyna główna):

```
sudo apt install ansible-core
```

 14. Sprawdź wersję programu (na ocenę 3.0):

```
ansible --version
```

15. Stwórz katalog `ansible` w `etc`:

```
sudo mkdir /etc/ansible
```

16. Sprawdź uprawnienia utworzonego folderu:

```
ls -l /etc/
```

 17. Czy możesz w tej chwili utworzyć nowy plik w katalogu `ansible`? Dlaczego? (na ocenę 3.0)

18. Dodaj odpowiednie uprawnienia:

```
sudo chmod o+w /etc/ansible
```

19. Stwórz plik inwentarza:

```
nano /etc/ansible/hosts
```

20. Dokonaj edycji pliku inwentarza tak, by maszyny `Ansible_node1` i `Ansible_node2` znajdowały się w osobnych grupach oraz trzeciej – wspólnej. Grupy powinny mieć następujące nazwy:

- `gr1_tvoj_indeks`,
- `gr2_tvoj_indeks`,
- `wspolna_tvoj_indeks`.

21. Skonfiguruj w pliku inwentarza domyślne dane do logowania zdalnego:

```
[all:vars]
ansible_connection=ssh
ansible_user=xxx
ansible_ssh_pass=xxx
```

22. Utwórz plik konfiguracyjny `ansible.cfg`:

```
nano /etc/ansible/ansible.cfg
```

23. W `ansible.cfg` dodaj wpis:

```
[defaults]
host_key_checking = false
```

 24. Spróbuj wykonać `ping` wszystkich węzłów z poziomu narzędzia Ansible (na ocenę 3.0):

```
ansible all -m ping
```

 25. Wypisz hosty należące do konkretnej grupy używając komendy (na ocenę 3.5):

```
ansible --list-hosts nazwa_grupy
```

 26. Jak wykonać `ping` jedynie do zdefiniowanej wcześniej grupy hostów? – podaj przykład takiego polecenia. (na ocenę 4.0)

27. Przykładowe wywołanie polecenia `shell` z odpowiednim modułem do tego:

```
ansible nazwa_grupy -m ansible.builtin.shell -a 'echo $PATH'
```

 28. Wykorzystując odpowiednie polecenie, stwórz na obu zarządzanych maszynach plik `bachleda_fare1.hej`, następnie kolejnym poleceniem wylistuj zawartość katalogu i sprawdź, czy plik został poprawnie utworzony. (na ocenę 5.0)