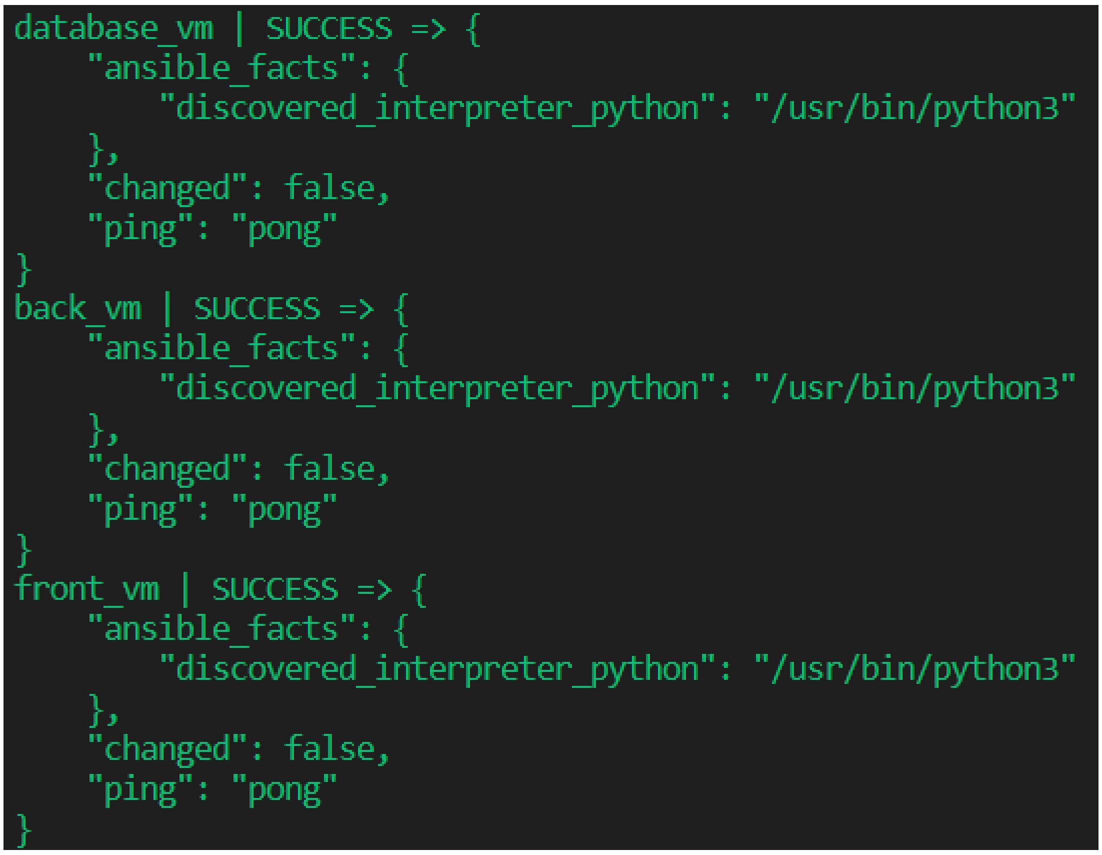
Na początku tworzymy maszyny za pomocą skryptu: `./deploy.sh config.json`. Skrypt oprócz tworzenia maszyn uzupełnia plik inventory.yaml oraz folder /vars.

Następnie sprawdza czy ansible może nawiązać połączenie z maszynami za pomocą komendy: `ansible -i inventory.yaml all -m ping`

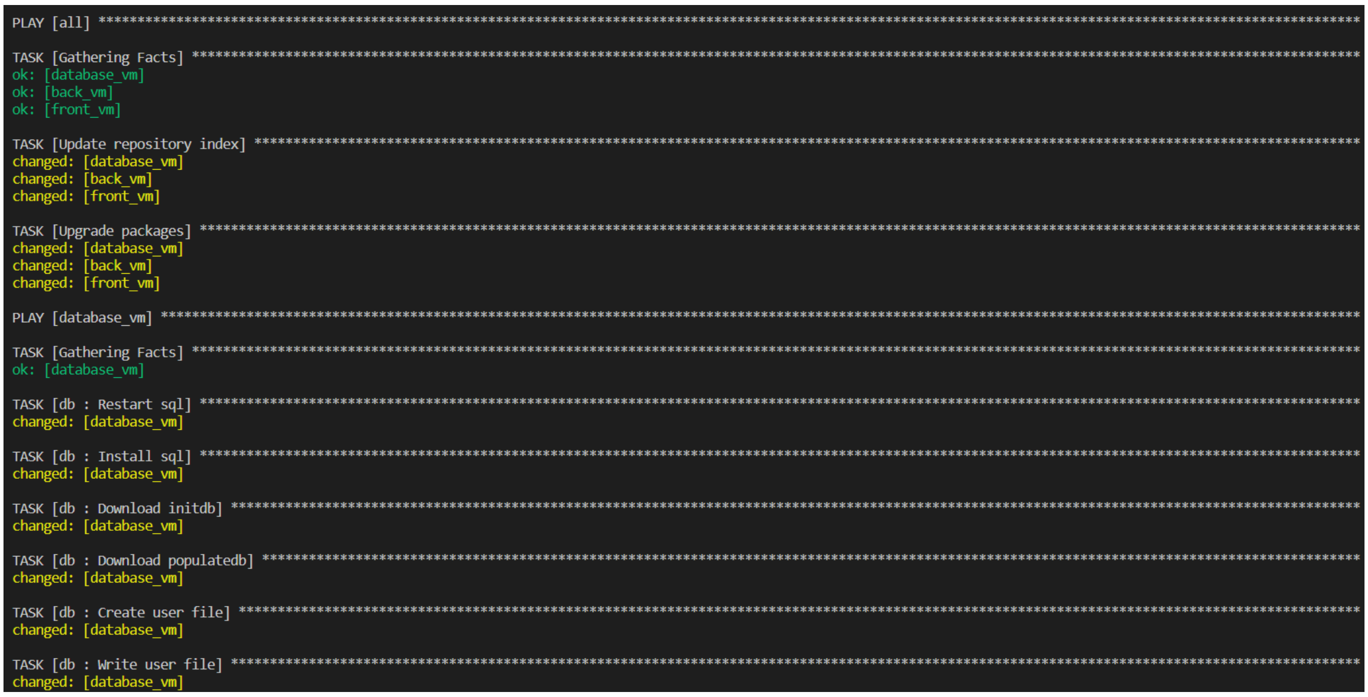


```
database_vm | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
back_vm | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
front_vm | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

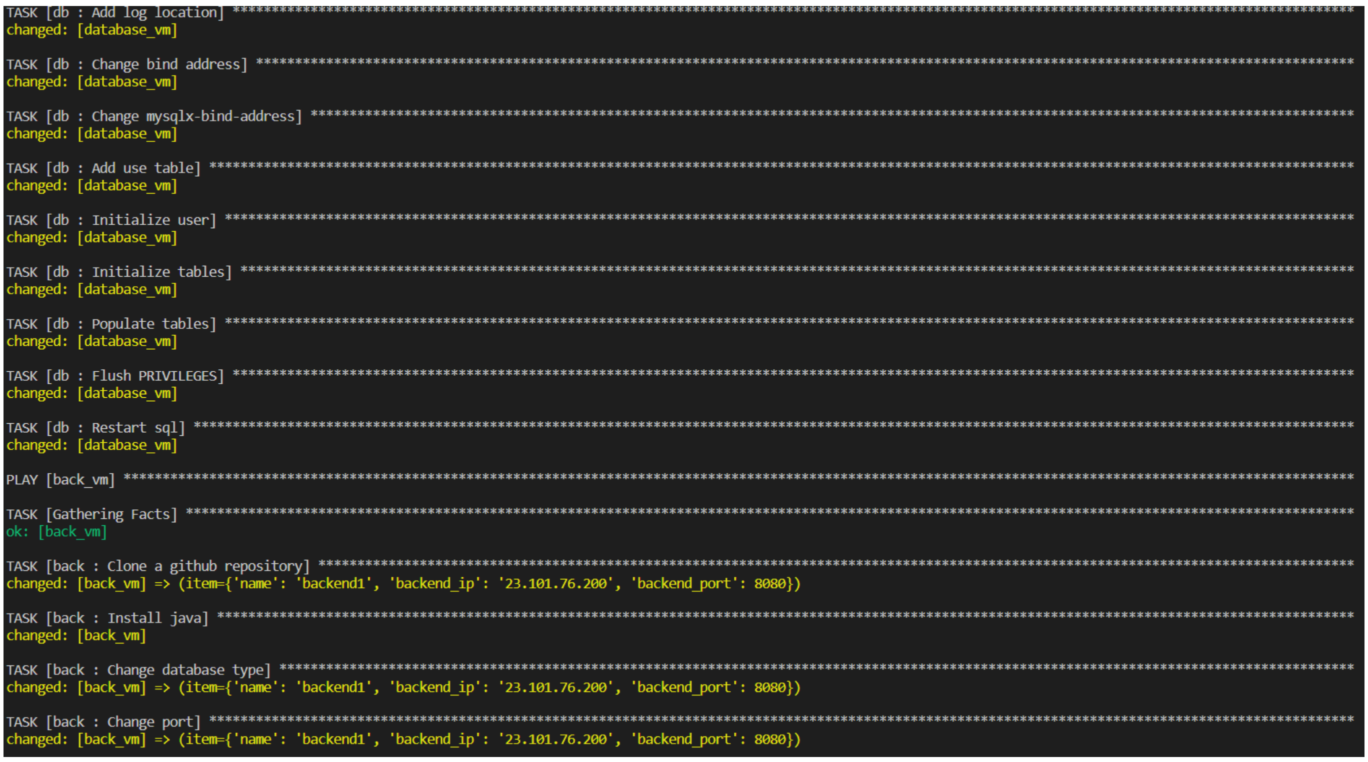
Aby uruchomić poszególne konfiguracje należy mieć zainstalowany ansible, a następnie wywołać komendę: `ansible-playbook playbookN.yaml -i inventory.yaml` ,gdzie N to numer konfiguracji

1

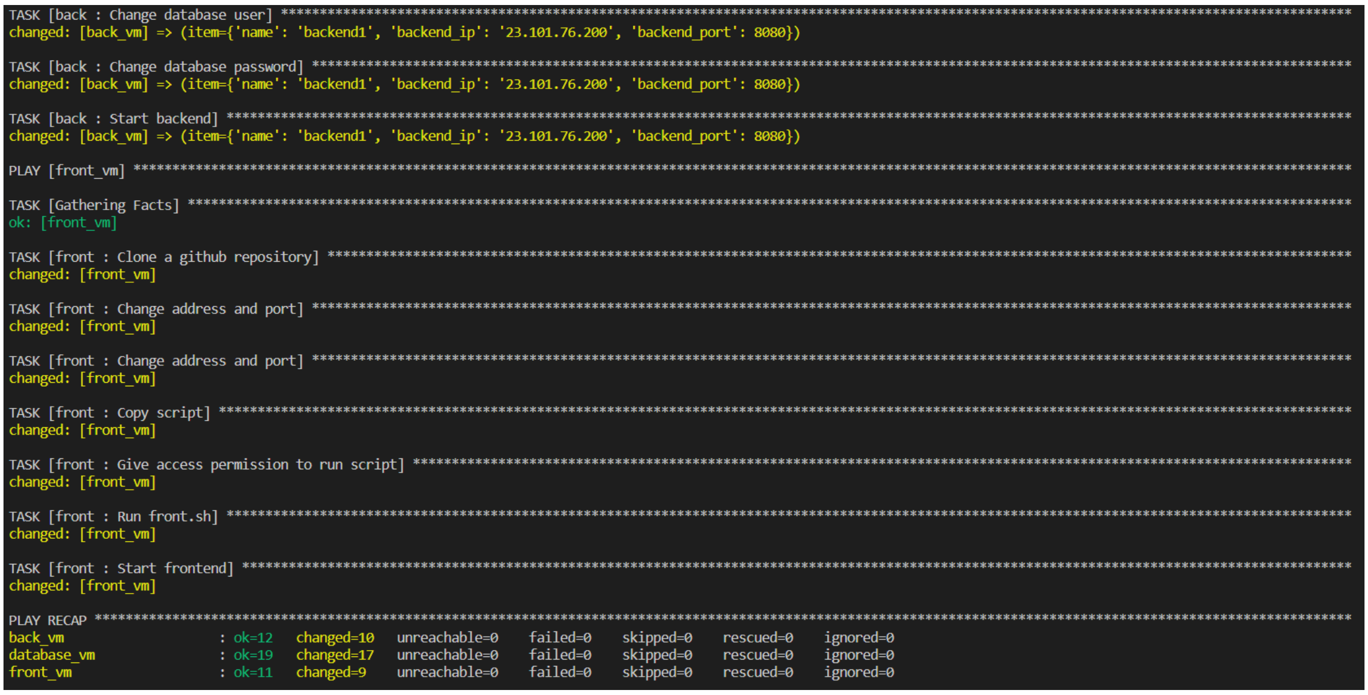
```
ansible-playbook playbook1.yaml -i inventory.yaml
```



```
PLAY [all] *********************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [database_vm]
ok: [back_vm]
ok: [front_vm]

TASK [Update repository index] *************************************************************************************************
changed: [database_vm]
changed: [back_vm]
changed: [front_vm]

TASK [Upgrade packages] ********************************************************************************************************
changed: [database_vm]
changed: [back_vm]
changed: [front_vm]

PLAY [database_vm] *************************************************************************************************************

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

TASK [db : Restart sql] ********************************************************************************************************
changed: [database_vm]

TASK [db : Install sql] ********************************************************************************************************
changed: [database_vm]

TASK [db : Download initdb] ****************************************************************************************************
changed: [database_vm]

TASK [db : Download populatedb] ************************************************************************************************
changed: [database_vm]

TASK [db : Create user file] ***************************************************************************************************
changed: [database_vm]

TASK [db : Write user file] ****************************************************************************************************
changed: [database_vm]
```



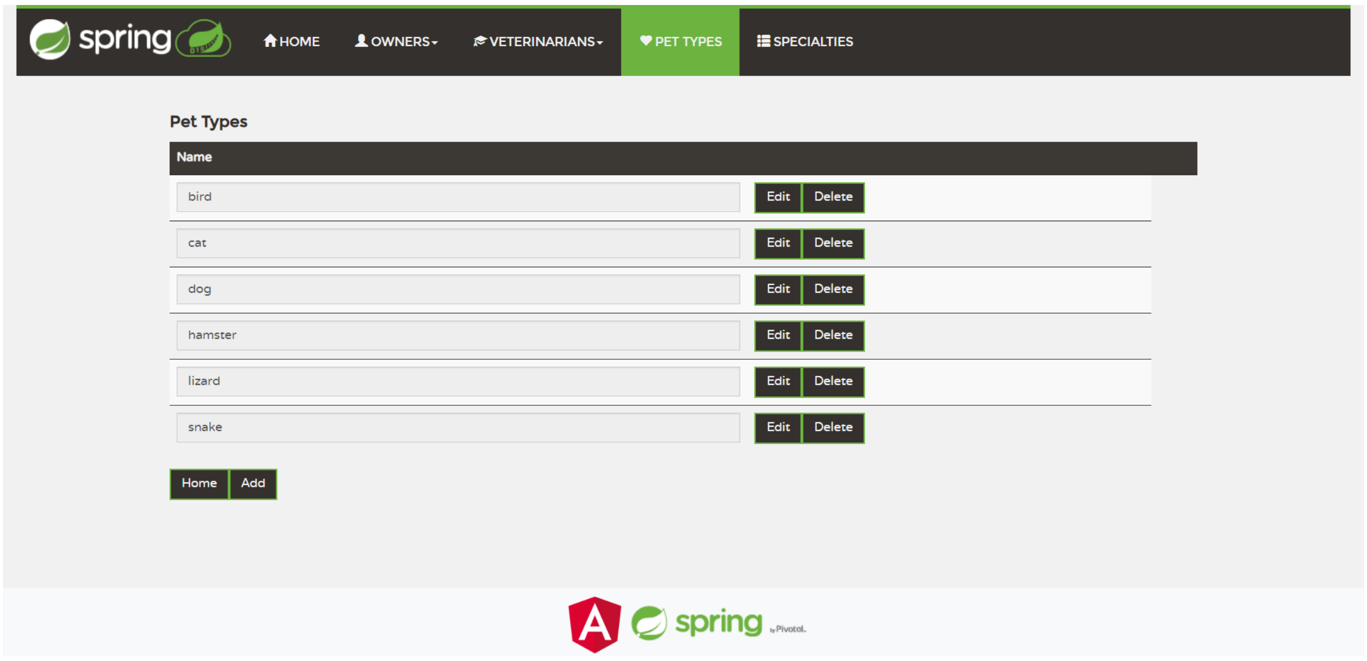
```
TASK [db : Add log location] *************************************************************************************************************************
changed: [database_vm]

TASK [db : Change bind address] **********************************************************************************************************************
changed: [database_vm]

TASK [db : Change mysqlx-bind-address] ***************************************************************************************************************
changed: [database_vm]

TASK [db : Add use table] ****************************************************************************************************************************
changed: [database_vm]

TASK [db : Initialize user] **************************************************************************************************************************
changed: [database_vm]

TASK [db : Initialize tables] ************************************************************************************************************************
changed: [database_vm]

TASK [db : Populate tables] **************************************************************************************************************************
changed: [database_vm]

TASK [db : Flush PRIVILEGES] *************************************************************************************************************************
changed: [database_vm]

TASK [db : Restart sql] ******************************************************************************************************************************
changed: [database_vm]

PLAY [back_vm] ***************************************************************************************************************************************

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

TASK [back : Clone a github repository] **************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})

TASK [back : Install java] ***************************************************************************************************************************
changed: [back_vm]

TASK [back : Change database type] *******************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})

TASK [back : Change port] ****************************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})
```



```
TASK [back : Change database user] *******************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})

TASK [back : Change database password] ***************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})

TASK [back : Start backend] **************************************************************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '23.101.76.200', 'backend_port': 8080})

PLAY [front_vm] **************************************************************************************************************************************

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

TASK [front : Clone a github repository] *************************************************************************************************************
changed: [front_vm]

TASK [front : Change address and port] ***************************************************************************************************************
changed: [front_vm]

TASK [front : Change address and port] ***************************************************************************************************************
changed: [front_vm]

TASK [front : Copy script] ***************************************************************************************************************************
changed: [front_vm]

TASK [front : Give access permission to run script] **************************************************************************************************
changed: [front_vm]

TASK [front : Run front.sh] **************************************************************************************************************************
changed: [front_vm]

TASK [front : Start frontend] ************************************************************************************************************************
changed: [front_vm]

PLAY RECAP *******************************************************************************************************************************************
back_vm                    : ok=12   changed=10   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
database_vm                : ok=19   changed=17   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
front_vm                   : ok=11   changed=9    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Po zakończeniu skryptu można przejść do stron frontendu oraz backendu. Adresy ip znajdują się w pliku inventory.yaml

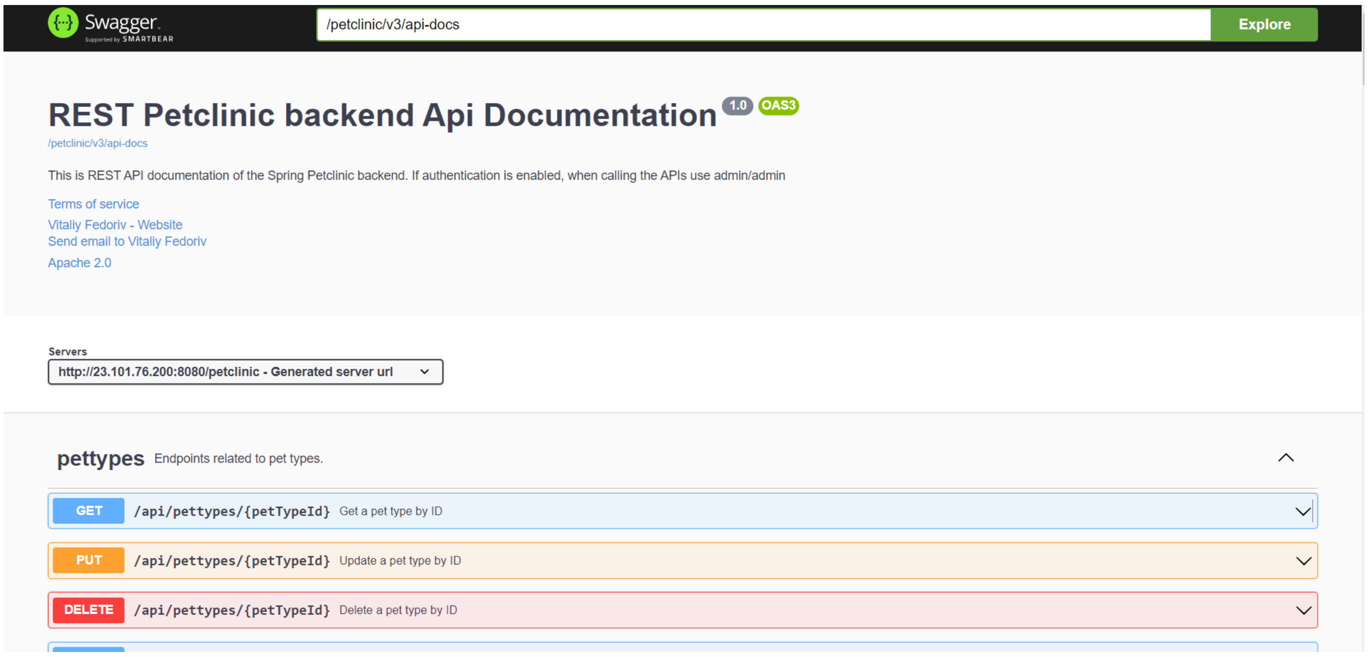
Adres frontendu: {front_ip}:8080



Pet Types

Name		
bird	Edit	Delete
cat	Edit	Delete
dog	Edit	Delete
hamster	Edit	Delete
lizard	Edit	Delete
snake	Edit	Delete

Adres backendu: {back_ip}:8080/petclinic



REST Petclinic backend Api Documentation 1.0 OAS3

/petclinic/v3/api-docs

This is REST API documentation of the Spring Petclinic backend. If authentication is enabled, when calling the APIs use admin/admin

Terms of service
Vitaliy Fedoriv - Website
Send email to Vitaliy Fedoriv
Apache 2.0

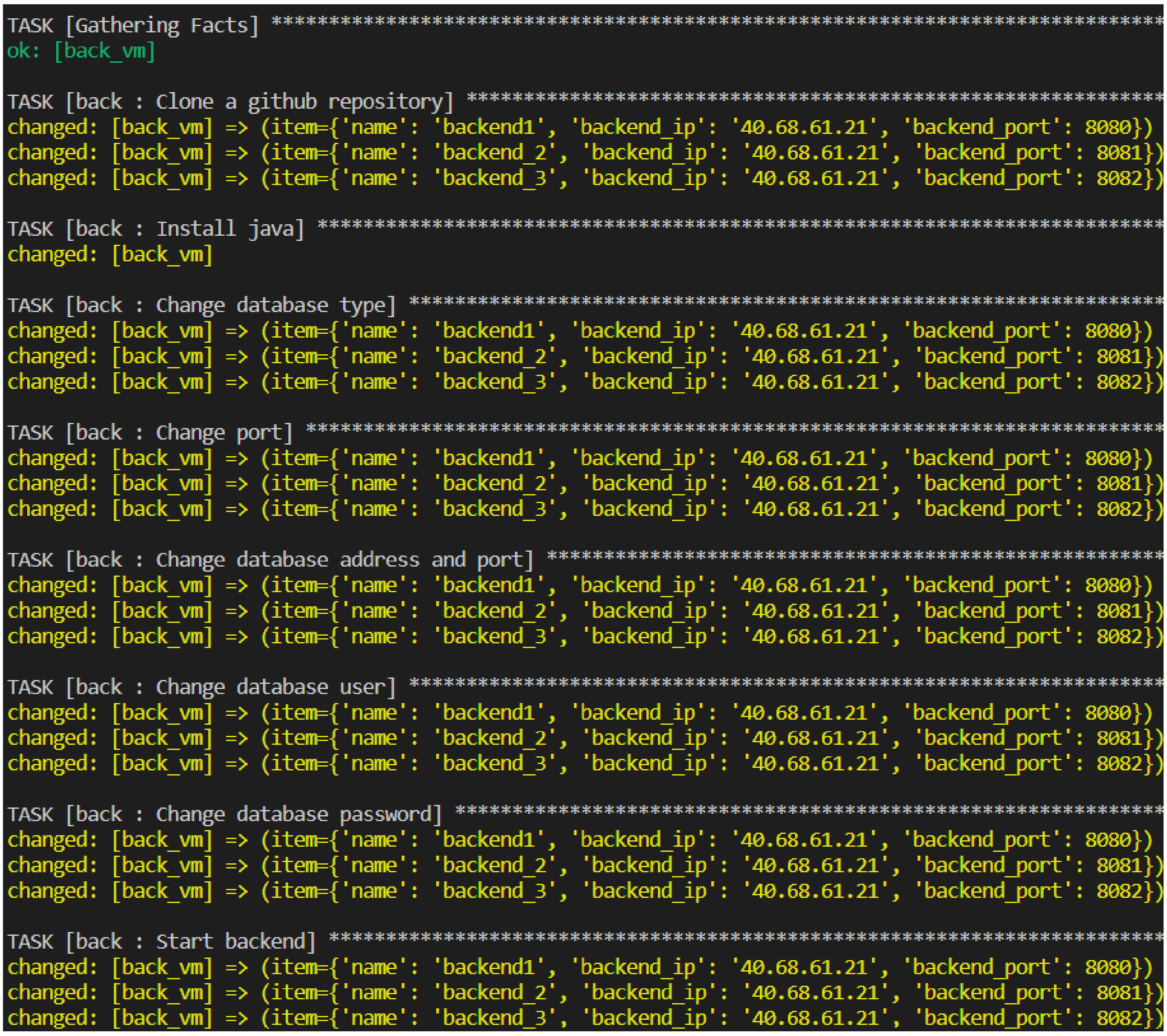
Servers
http://23.101.76.200:8080/petclinic - Generated server url

pettypes Endpoints related to pet types.

GET `/api/pettypes/{petTypeId}` Get a pet type by ID
PUT `/api/pettypes/{petTypeId}` Update a pet type by ID
DELETE `/api/pettypes/{petTypeId}` Delete a pet type by ID

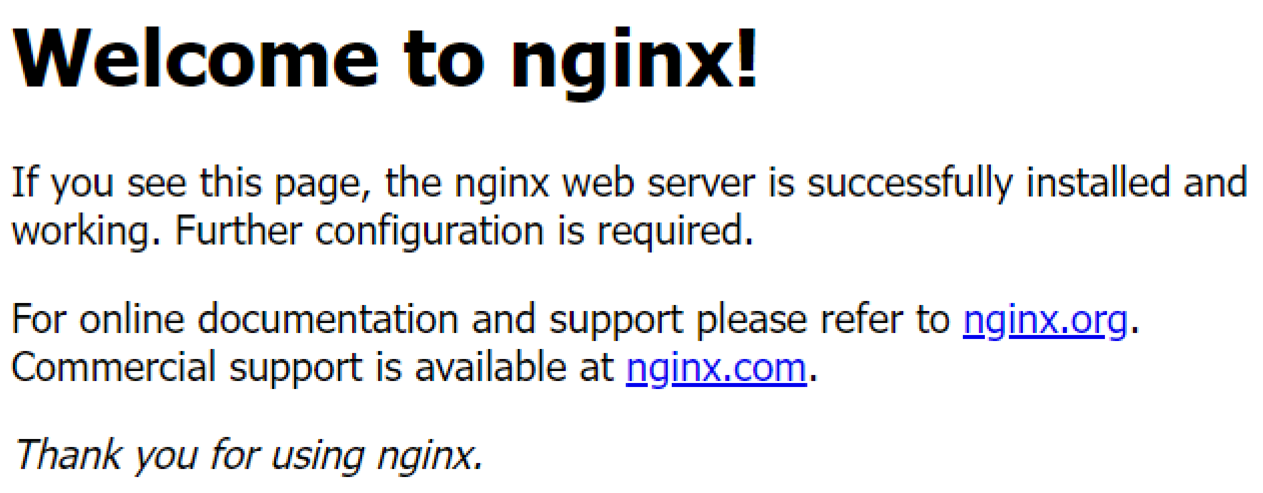
2

`ansible-playbook playbook1.yaml -i inventory.yaml` W 2 konfiguracji właczamy wykonywanie skryptu dla backendu w pętli na maszynie backend_VM.



```
TASK [Gathering Facts] ******************************************************************************
ok: [back_vm]

TASK [back : Clone a github repository] *************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Install java] **************************************************************************
changed: [back_vm]

TASK [back : Change database type] ******************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Change port] ***************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Change database address and port] ******************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Change database user] ******************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Change database password] **************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})

TASK [back : Start backend] *************************************************************************
changed: [back_vm] => (item={'name': 'backend1', 'backend_ip': '40.68.61.21', 'backend_port': 8080})
changed: [back_vm] => (item={'name': 'backend_2', 'backend_ip': '40.68.61.21', 'backend_port': 8081})
changed: [back_vm] => (item={'name': 'backend_3', 'backend_ip': '40.68.61.21', 'backend_port': 8082})
```

Uzyskujemy połącznie przez nginx.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

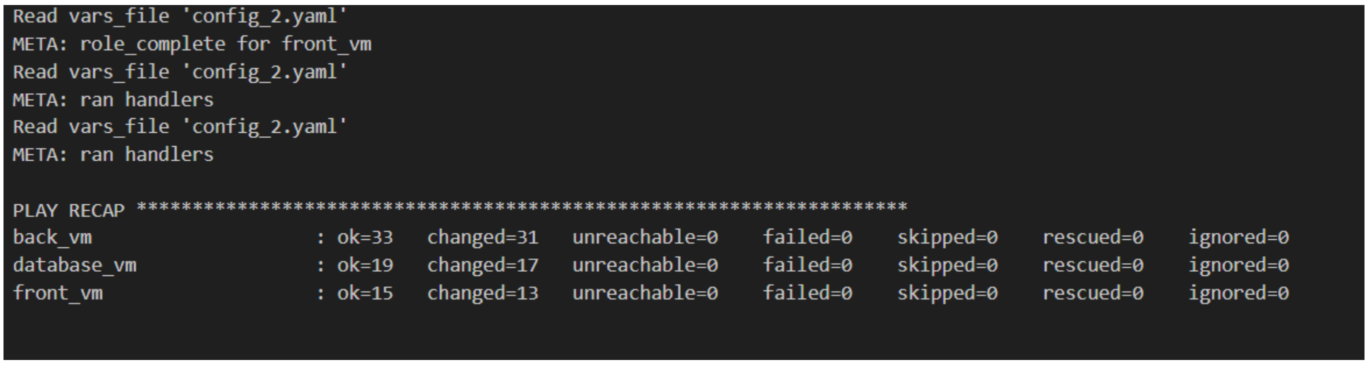
Thank you for using nginx.

5

```
ansible-playbook playbook5.yaml -i inventory.yaml
```

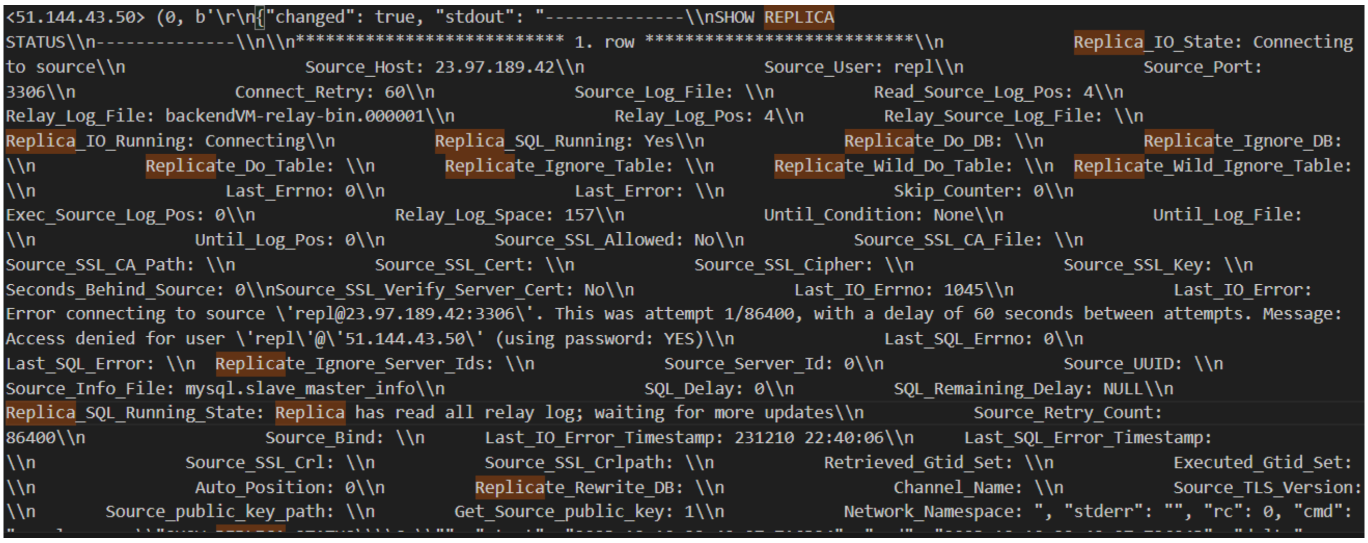
Deploy-ujemy replikę bazy danych na maszynie wirtualnej backend, nginx jest po stronie frontendu, i dostaje jako argument dwa porty jako dwie instancje backendu do obsługi.

Konfiguracja zakończyła się pomyślnie, wszystkie taski były zakończone sukcesem



```
Read vars_file 'config_2.yaml'
META: role_complete for front_vm
Read vars_file 'config_2.yaml'
META: ran handlers
Read vars_file 'config_2.yaml'
META: ran handlers

PLAY RECAP *********************************************************************
back_vm                    : ok=33   changed=31   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
database_vm                : ok=19   changed=17   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
front_vm                   : ok=15   changed=13   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

W logach po włączeniu playbooka, możemy zobaczyć "Replica_SQL_Running_State: Replica has read all relay log; waiting for more updates" co oznacza że replika jest poprawnie skonfigurowana, i podłączyła się do głównego serwera mysql.



```
<51.144.43.50> (0, b'\r\n{"changed": true, "stdout": "--------------\\nSHOW REPLICA
STATUS\\n--------------\\n\\n*************************** 1. row ***************************\\n           Replica_IO_State: Connecting
to source\\n                  Source_Host: 23.97.189.42\\n                  Source_User: repl\\n                  Source_Port:
3306\\n              Connect_Retry: 60\\n              Source_Log_File: \\n          Read_Source_Log_Pos: 4\\n
Relay_Log_File: backendVM-relay-bin.000001\\n             Relay_Log_Pos: 4\\n        Relay_Source_Log_File: \\n
Replica_IO_Running: Connecting\\n            Replica_SQL_Running: Yes\\n             Replicate_Do_DB: \\n          Replicate_Ignore_DB:
\\n          Replicate_Do_Table: \\n       Replicate_Ignore_Table: \\n      Replicate_Wild_Do_Table: \\n  Replicate_Wild_Ignore_Table:
\\n                  Last_Errno: 0\\n                   Last_Error: \\n                 Skip_Counter: 0\\n
Exec_Source_Log_Pos: 0\\n             Relay_Log_Space: 157\\n              Until_Condition: None\\n               Until_Log_File:
\\n               Until_Log_Pos: 0\\n           Source_SSL_Allowed: No\\n           Source_SSL_CA_File: \\n
Source_SSL_CA_Path: \\n             Source_SSL_Cert: \\n           Source_SSL_Cipher: \\n               Source_SSL_Key: \\n
Seconds_Behind_Source: 0\\nSource_SSL_Verify_Server_Cert: No\\n               Last_IO_Errno: 1045\\n               Last_IO_Error:
Error connecting to source \'repl@23.97.189.42:3306\'. This was attempt 1/86400, with a delay of 60 seconds between attempts. Message:
Access denied for user \'repl\'@\'51.144.43.50\' (using password: YES)\\n              Last_SQL_Errno: 0\\n
Last_SQL_Error: \\n  Replicate_Ignore_Server_Ids: \\n            Source_Server_Id: 0\\n                  Source_UUID: \\n
Source_Info_File: mysql.slave_master_info\\n                    SQL_Delay: 0\\n          SQL_Remaining_Delay: NULL\\n
Replica_SQL_Running_State: Replica has read all relay log; waiting for more updates\\n           Source_Retry_Count:
86400\\n                  Source_Bind: \\n      Last_IO_Error_Timestamp: 231210 22:40:06\\n     Last_SQL_Error_Timestamp:
\\n              Source_SSL_Crl: \\n          Source_SSL_Crlpath: \\n          Retrieved_Gtid_Set: \\n            Executed_Gtid_Set:
\\n               Auto_Position: 0\\n         Replicate_Rewrite_DB: \\n                Channel_Name: \\n           Source_TLS_Version:
\\n       Source_public_key_path: \\n        Get_Source_public_key: 1\\n            Network_Namespace: ", "stderr": "", "rc": 0, "cmd":
```