

Міністерство освіти і науки України
Карпатський національний університет
імені В.Стефаника

Факультет математики та інформатики
Кафедра інформаційних технологій

Інформатика і програмування

Лабораторна робота № 6

Тема: Впровадження управління мережевим трафіком

Виконав: Федірко А.Р

Група ПЗ-43

Дата: 12 жовтня 2025р.

Викладач: Поварчук Д.Д

Івано-Франківськ – 2025

Лабораторний сценарій

Ваша організація має публічний веб-сайт. Вам потрібно балансувати навантаження вхідних публічних запитів між різними віртуальними машинами. Вам також потрібно надати зображення та відео з різних віртуальних машин. Ви плануєте впровадити балансувальник навантаження Azure та шлюз додатків Azure. Усі ресурси знаходяться в одному регіоні.

Одразу хочу сказати що вирішив трішки ультанути і цього разу буде тільки кілька скріншотів як тераформ працює і два скріна в ажурі, цього разу теж з першого разу не вийшло задеплойти

Отож, скрін команди terraform plan та terraform validate

The image shows a Windows desktop environment. The primary application is Visual Studio Code, which has a dark theme. The Explorer sidebar on the left shows a project structure with folders like '.terraform' and 'terraform.tfvars', and files like 'main.tf'. The main editor area displays the content of 'main.tf', which is a Terraform configuration for an Azure Application Gateway. The code includes resource definitions for 'azurerm_application_gateway' and 'azurerm_routing_rule', along with a comment in Ukrainian about routing rules. The bottom of the editor shows tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', 'PORTS', and 'AZURE'. The 'TERMINAL' tab is active, showing a PowerShell session where the command 'terraform validate' was executed, resulting in a 'Success!' message. The Windows taskbar at the bottom includes the Start button, a search bar, and several pinned application icons. The system tray shows the date and time as 12:10:25 on 12.10.2025.

```
PS C:\Users\fedir\Documents\IPZ_43\Cloud\technologies\lab6> terraform plan -out=lab6template

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# azurerm_application_gateway.appgw will be created
+ resource "azurerm_application_gateway" "appgw" {
  + id                        = (known after apply)
  + location                 = "eastus"
  + name                     = "az104-appgw"
  + private_endpoint_connection = (known after apply)
  + resource_group_name      = "az104-rg6"

  + backend_address_pool {
    + fqdns      = []
    + id         = (known after apply)
    + ip_addresses = (known after apply)
    + name       = "az104-appgwbe"
  }
  + backend_address_pool {
    + fqdns      = []
    + id         = (known after apply)
  }
}
```

Остання (також успішна) спроба задеплоїти всі сервіси

```
Plan: 26 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ appgw_public_ip = (known after apply)
+ lb_public_ip    = (known after apply)

Saved the plan to: lab6template

To perform exactly these actions, run the following command to apply:
    terraform apply "lab6template"
PS C:\Users\Fedir\Documents\TPZ_43\Cloud_tecnologies\lab6> terraform apply "lab6template"
random_password.vm: Creating...
random_password.vm: Creation complete after 0s [id=none]
```

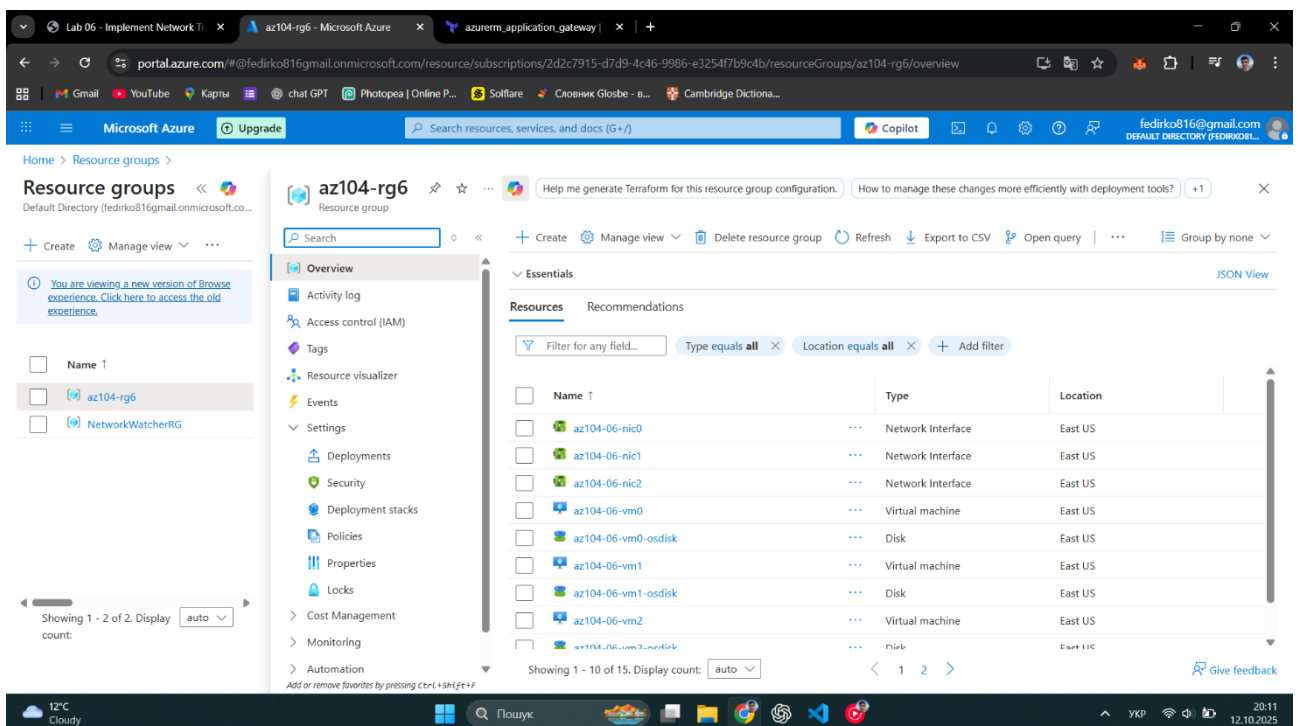
```
412 host = "127.0.0.1"
413 path = "/"

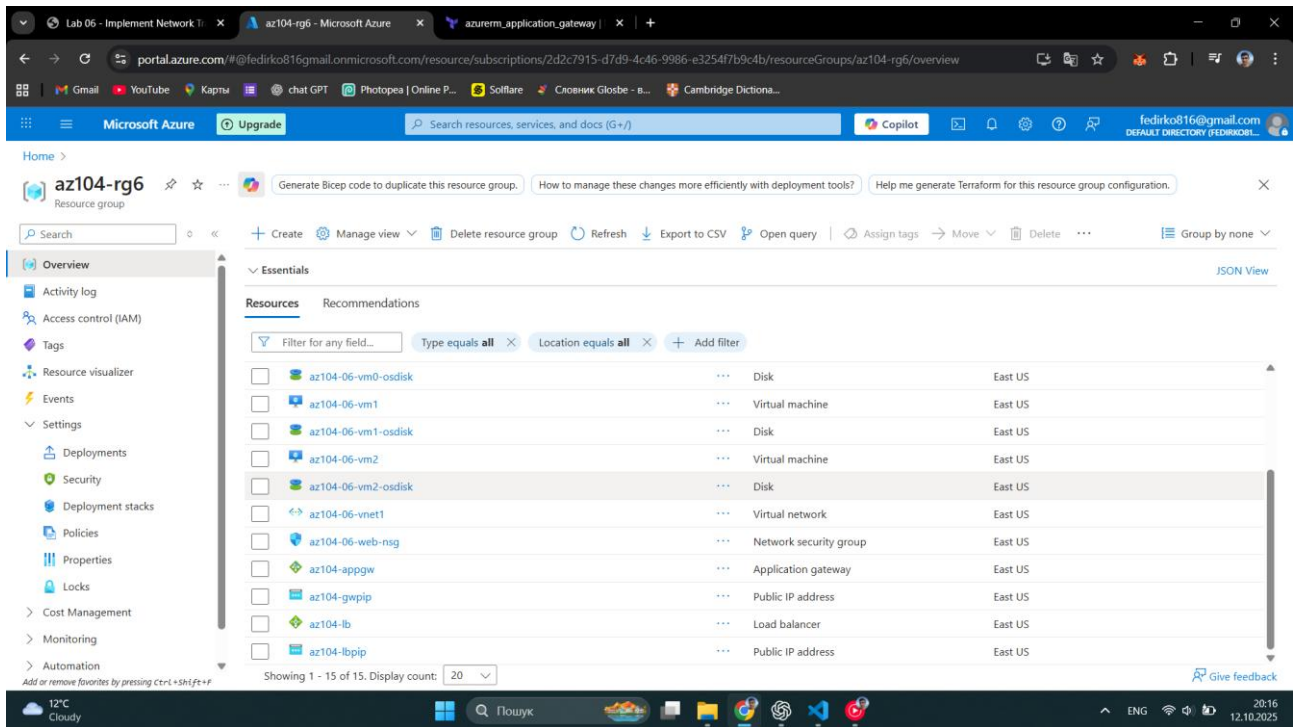
azurerm_application_gateway.appgw: Still creating... [02m20s elapsed]
azurerm_application_gateway.appgw: Still creating... [02m30s elapsed]
azurerm_application_gateway.appgw: Still creating... [02m40s elapsed]
azurerm_application_gateway.appgw: Still creating... [02m50s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m00s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m10s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m20s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m30s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m40s elapsed]
azurerm_application_gateway.appgw: Still creating... [03m50s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m00s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m10s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m20s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m30s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m40s elapsed]
azurerm_application_gateway.appgw: Still creating... [04m50s elapsed]
azurerm_application_gateway.appgw: Creation complete after 4m54s [id=/subscriptions/2d2c7915-d7d9-4c46-9986-e3254f7b9c4b/resourceGroups/az104-rg6/providers/Microsoft.Network/applicationsGateways/az104-appgw]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

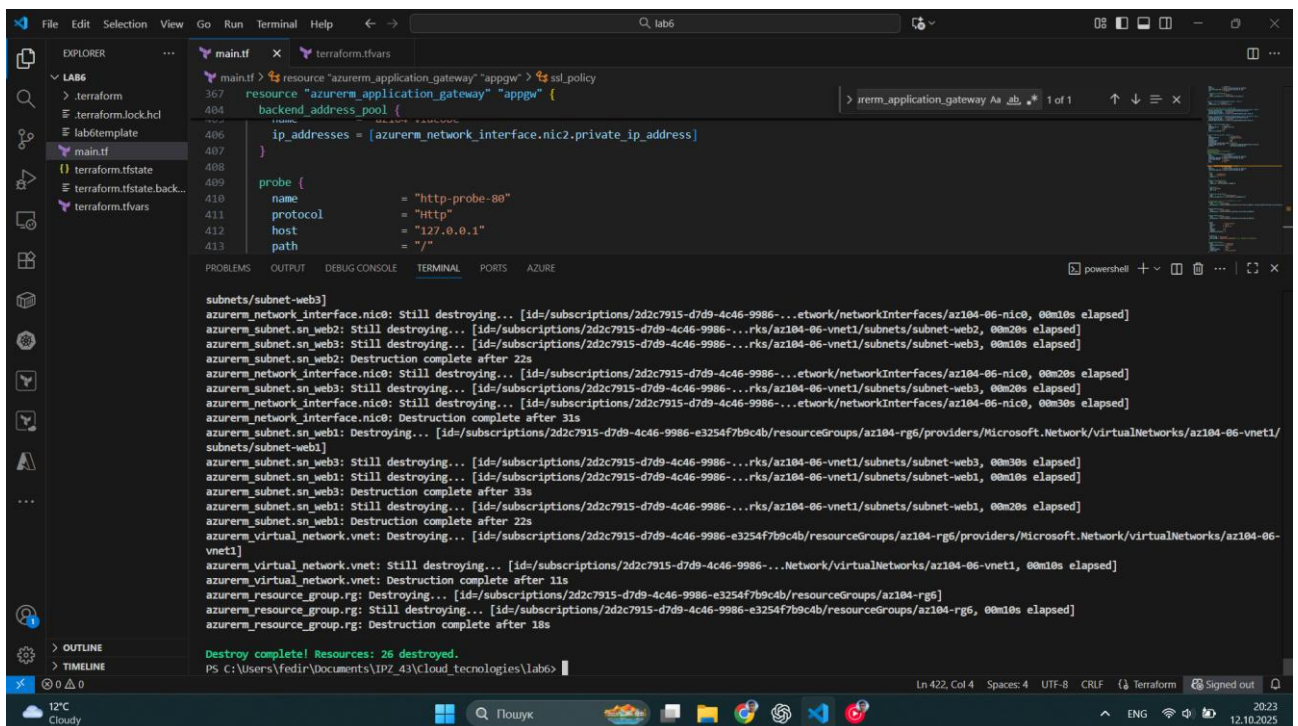
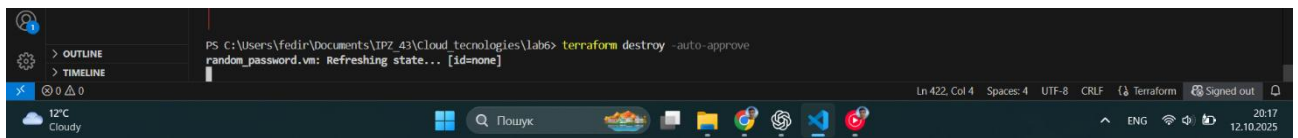
Outputs:
appgw_public_ip = "74.235.203.159"
lb_public_ip    = "20.169.172.30"
PS C:\Users\Fedir\Documents\TPZ_43\Cloud_tecnologies\lab6>
```

Ну і скріншот-докази що все запустилось





Ну і видалення)



Висновок:

У ході лабораторної роботи було налаштовано балансування мережевого трафіку за допомогою Azure Load Balancer та Azure Application Gateway.

Load Balancer забезпечує розподіл запитів на рівні L4 (TCP) між кількома віртуальними машинами, що підвищує відмовостійкість та ефективність обробки запитів.

Application Gateway працює на рівні L7 (HTTP/HTTPS), що дозволяє виконувати маршрутизацію на основі шляхів (/image/, /video/) та забезпечує додаткові можливості, такі як SSL-термінація і Web Application Firewall (WAF).

Отримано практичні навички конфігурації публічного IP, створення пулів бекендів, правил маршрутизації та перевірки роботи балансувальників. У результаті реалізовано сценарій, де вебзапити рівномірно розподіляються між кількома ВМ, а контент різного типу обробляється окремими бекендами.