

AUTOMATED WEB SERVER INFRASTRUCTURE PROVISIONING AND CONFIGURATION WITH INFRASTRUCTURE AS CODE (IAC) USING TERRAFORM IN MICROSOFT AZURE

PROJEK AKHIR CLOUD COMPUTING AND SECURITY

Untuk memenuhi tugas akhir
mata kuliah Cloud Computing and Security

Disusun oleh:

Mikhael Ryan Toar	215150300111006
Dimas Yudistira Wijaya	215150300111009
Mohammad Faiz M.	215150300111038
Rizaldy Ariobimo K.	215150301111003
Abraar Bimasakti	215150307111005



Teknik Komputer
Departemen Teknik Informatika
Fakultas Ilmu Komputer
Universitas Brawijaya
Malang
2024

Projek dilakukan menggunakan Infrastructure as Code (IaC) platform Terraform, untuk provisioning dan konfigurasi infrastruktur Web Server pada Cloud Provider Microsoft Azure.

Deskripsi Infrastruktur

Template mengandung provisioning dan konfigurasi Web Server yang akan menampilkan pesan “Welcome to Web Server 1/2/3 SEA/JP Region” menggunakan Nginx, yang dijalankan pada beberapa Azure Virtual Machines yang diletakkan di belakang sebuah Public Load Balancer. Infrastruktur didirikan pada dua region yaitu South East Asia dan Japan East sehingga akan membuat layanan Web App yang sama untuk dua region berbeda. Sebuah File Share storage yang telah dibuat menggunakan Azure Storage Account, akan di mount kepada setiap Virtual Machine pada kedua region untuk kebutuhan Web Server jika diperlukan, dan status konektivitas Web Server dengan file share akan ditampilkan pada Web App dengan pesan “File Share Status: Connected”. Selain di deploy pada dua region berbeda, Virtual Machine di deploy ke dalam Availability Zones berbeda di dalam setiap region. Hal ini dilakukan untuk aspek ‘High Availability’ sehingga layanan Web Server akan tetap Available walaupun terdapat kasus datacenter failure. Selain itu, setiap Virtual Machine juga di konfigurasikan kedalam Daily Backup. Dimana, backup setiap Virtual Machine kemudian akan disimpan kedalam Recovery Services Vault pada tiap region sehingga meningkatkan aspek Durability dan Recoverability kepada infrastruktur Web Server.

Konektivitas kepada Web Server dilakukan melalui Public IP Load Balancer melalui protokol HTTP Port 80.

Projek Terbagi kedalam dua bagian: Infrastructure Provisioning dan Initial VM Configuration menggunakan Setup Script.

File/Full template code bisa di cek pada

<https://github.com/Leonheart-Libel/LeonProjekCCS>

A. Provisioning

CCSProjectAzure.tf -> File Utama Terraform untuk provisioning Infrastruktur

Terraform.tfvars -> Letak Environment Variables yang akan digunakan dengan main template file. Terdapat data sensitif mengandung:

- Azure Subscription ID
- Azure Service Principal App ID
- Azure Service Principal Password
- Azure Tenant ID

1. Resource Group
















Resource Group adalah sebuah container logika yang mengandung resources terkait dalam sebuah solusi dalam Azure. Resources yang dapat terkandung dalam sebuah Resource Group termasuk Virtual Machines, Storage Account, Web Apps, Databases, Virtual network, dll. Semua resource dalam azure dimasukkan ke dalam sebuah resource group untuk memudahkan organisasi resource, dan manajemen unified.

Sehingga untuk kepentingan projek, dibutuhkan setidaknya satu resource group untuk provision resources pada setiap region.

Resource group Southeast Asia

```
resource "azurerm_resource_group" "sea_resource_group" {  
  provider = azurerm.sea_provider  
  name     = "sea-webapp-rg"  
  location = "Southeast Asia"  
}
```

Semua resources yang akan dibuat melalui Terraform Template pada resource group SEA
















<input type="checkbox"/> Name ↑↓	Type ↑↓	Location ↑↓
<input type="checkbox"/>  sea-lb-publicip	Public IP address	Southeast Asia
<input type="checkbox"/>  sea-webapp-backup-vault	Recovery Services vault	Southeast Asia
<input type="checkbox"/>  sea-webapp-lb	Load balancer	Southeast Asia
<input type="checkbox"/>  sea-webapp-vnet	Virtual network	Southeast Asia
<input type="checkbox"/>  sea-webserver-1	Virtual machine	Southeast Asia
<input type="checkbox"/>  sea-webserver-1_OsDisk_1_55d5fa82931c4edb9d88cb811ed112ff	Disk	Southeast Asia
<input type="checkbox"/>  sea-webserver-2	Virtual machine	Southeast Asia
<input type="checkbox"/>  sea-webserver-2_OsDisk_1_2ae45ecd57d047b48239934b64a130fa	Disk	Southeast Asia
<input type="checkbox"/>  sea-webserver-3	Virtual machine	Southeast Asia
<input type="checkbox"/>  sea-webserver-3_OsDisk_1_57e88a02e988421c87d3499d7d2475b0	Disk	Southeast Asia
<input type="checkbox"/>  sea-webserver-nic-1	Network Interface	Southeast Asia
<input type="checkbox"/>  sea-webserver-nic-2	Network Interface	Southeast Asia
<input type="checkbox"/>  sea-webserver-nic-3	Network Interface	Southeast Asia
<input type="checkbox"/>  seawebappstoragekeyk3l7i5f	Storage account	Southeast Asia
<input type="checkbox"/>  webapp-nsg	Network security group	Southeast Asia

Resource group Japan east

```
resource "azurerm_resource_group" "jp_resource_group" {  
  provider = azurerm.jp_provider  
  name     = "jp-webapp-rg"
```

```
location = "Japan East"
}
```

Semua resources yang akan dibuat melalui Terraform Template pada resource group JP

<input type="checkbox"/> Name ↑↓	Type ↑↓	Location ↑↓
<input type="checkbox"/>  jp-lb-publicip	Public IP address	Japan East
<input type="checkbox"/>  jp-webapp-backup-vault	Recovery Services vault	Japan East
<input type="checkbox"/>  jp-webapp-lb	Load balancer	Japan East
<input type="checkbox"/>  jp-webapp-vnet	Virtual network	Japan East
<input type="checkbox"/>  jp-webserver-1	Virtual machine	Japan East
<input type="checkbox"/>  jp-webserver-1_OsDisk_1_61c759be5007417dbcb289865a784834	Disk	Japan East
<input type="checkbox"/>  jp-webserver-2	Virtual machine	Japan East
<input type="checkbox"/>  jp-webserver-2_OsDisk_1_ecaa217d2eb041e89c7a2113da051ab3	Disk	Japan East
<input type="checkbox"/>  jp-webserver-3	Virtual machine	Japan East
<input type="checkbox"/>  jp-webserver-3_OsDisk_1_4eec5960c67c4a948fa11fd1093ed0cf	Disk	Japan East
<input type="checkbox"/>  jp-webserver-nic-1	Network Interface	Japan East
<input type="checkbox"/>  jp-webserver-nic-2	Network Interface	Japan East
<input type="checkbox"/>  jp-webserver-nic-3	Network Interface	Japan East
<input type="checkbox"/>  jpwebappstorageyk3l7i5f	Storage account	Japan East
<input type="checkbox"/>  webapp-nsg	Network security group	Japan East

2. Virtual Network

Virtual Network, dalam Azure merupakan building block fundamental untuk segala hal terkait networking. Virtual Network (Vnet) digunakan untuk secara secure connect azure resources kepada satu sama lain, ke internet, atau ke network on-premise. Virtual network memberikan lingkungan networking isolated dengan kendali penuh atas IP Address range, subnets, routing, dan secure policies.

Virtual Network Southeast Asia

```
resource "azurerm_virtual_network" "sea_vnet" {
  provider      = azurerm.sea_provider
  name          = "sea-webapp-vnet"
  address_space = ["10.0.0.0/16"]
  location      = azurerm_resource_group.sea_resource_group.location
  resource_group_name = azurerm_resource_group.sea_resource_group.name
}
```

Virtual Network Japan East

```
resource "azurerm_virtual_network" "jp_vnet" {
  provider      = azurerm.jp_provider
  name          = "jp-webapp-vnet"
  address_space = ["10.1.0.0/16"]
  location      = azurerm_resource_group.jp_resource_group.location
}
```

```
resource_group_name = azurerm_resource_group.jp_resource_group.name
}
```

Virtual network yang dibuat pada Azure tidak dapat memiliki address space yang saling conflicting (“10.0.0.0/16” pada SEA, “10.1.0.0/16” pada JP).

a. Subnet

Subnet atau Subnetwork, pada Azure adalah sebuah segment lebih kecil dari sebuah Virtual Network, yang memungkinkan untuk membagi virtual network menjadi segmen terisolasi yang lebih kecil. Subnets menghasilkan organisasi yang lebih baik, enhanced security, dan kontrol yang lebih granular atas komunikasi resource dalam VNet.

Subnet memiliki address space yang merupakan bagian dari Virtual Network.

Subnet SEA

```
resource "azurerm_subnet" "sea_subnet" {
  provider      = azurerm.sea_provider
  name          = "sea-webapp-subnet"
  resource_group_name = azurerm_resource_group.sea_resource_group.name
  virtual_network_name = azurerm_virtual_network.sea_vnet.name
  address_prefixes   = ["10.0.1.0/24"]
}
```

Subnet JP

```
resource "azurerm_subnet" "jp_subnet" {
  provider      = azurerm.jp_provider
  name          = "jp-webapp-subnet"
  resource_group_name = azurerm_resource_group.jp_resource_group.name
  virtual_network_name = azurerm_virtual_network.jp_vnet.name
  address_prefixes   = ["10.1.1.0/24"]
}
```

3. Azure Virtual Machine

Virtual Machine adalah service komputasi utama pada Microsoft Azure yang menyajikan resources komputasi virtualized on-demand, scalable, dan flexible. Membuat Virtual Machine pada Azure terdiri dari komponen lain seperti os_disk, source_image (Operating System), dan juga network interface.

Virtual Machine South East Asia

```
resource "azurerm_linux_virtual_machine" "sea_webservers" {
  count                = 3
  provider            = azurerm.sea_provider
  name                = "sea-webserver-${count.index + 1}"
  resource_group_name = azurerm_resource_group.sea_resource_group.name
  location            = azurerm_resource_group.sea_resource_group.location

  size              = "Standard_B1s"
  admin_username    = "azureuser"

  zone = toString(count.index + 1) # Distribusi VMs sepanjang Availability
  # Zones 1, 2, 3

  network_interface_ids = [
    azurerm_network_interface.sea_nic[count.index].id
  ]

  admin_ssh_key {
    username   = "azureuser"
    public_key = file("C:/Users/rocke/.ssh/id_rsa.pub") # Key SSH Admin
    # untuk Konfigurasi Awal
  }

  os_disk {
    caching              = "ReadWrite"
    storage_account_type = "Standard_LRS"
  }

  source_image_reference {
    publisher = "Canonical"
    offer     = "0001-com-ubuntu-server-jammy"
    sku       = "22_04-lts"
    version   = "latest"
  }
}
```

Virtual Machine Japan East

```
resource "azurerm_linux_virtual_machine" "jp_webservers" {
  count                = 3
  provider             = azurerm.jp_provider
  name                 = "jp-webserver-${count.index + 1}"
  resource_group_name = azurerm_resource_group.jp_resource_group.name
  location             = azurerm_resource_group.jp_resource_group.location
  size                 = "Standard_B1s"
  admin_username       = "azureuser"

  zone = tostring(count.index % 2 + 2) # Distribusi VM melalui
  Availability Zones 2 dan 3

  network_interface_ids = [
    azurerm_network_interface.jp_nic[count.index].id
  ]

  admin_ssh_key {
    username   = "azureuser"
    public_key = file("C:/Users/rocke/.ssh/id_rsa.pub") # Key SSH Admin
    untuk Konfigurasi Awal
  }

  os_disk {
    caching              = "ReadWrite"
    storage_account_type = "Standard_LRS"
  }

  source_image_reference {
    publisher = "Canonical"
    offer     = "0001-com-ubuntu-server-jammy"
    sku       = "22_04-lts"
    version   = "latest"
  }
}
```

Dalam proyek ini, tiga virtual machine dibuat pada setiap region, yang di deploy ke availability zone berbeda.

Spesifikasi

- Size : Standard_ B1s
- Image: Ubuntu Server 22_04-lts

Virtual machines ✨ ...

Default Directory

+ Create ▾ ↗ Switch to classic ⌚ Reservations ▾ ⚙ Manage view ▾ ↻ Refresh ⬇ Export to CSV 🔗 Open query | ...

Filter for any field... Subscription equals all Type equals all + Add filter ▾ More (2)

Showing 1 to 6 of 6 records. No grouping ▾ List view ▾

<input type="checkbox"/> Name ↑↓	Subscription ↑↓	Resource group ↑↓	Location ↑↓	Status ↑↓	Operating system ↑↓
<input type="checkbox"/> jp-webserver-1	Azure subscription 1	jp-webapp-rg	Japan East	Running	Linux
<input type="checkbox"/> jp-webserver-2	Azure subscription 1	jp-webapp-rg	Japan East	Running	Linux
<input type="checkbox"/> jp-webserver-3	Azure subscription 1	jp-webapp-rg	Japan East	Running	Linux
<input type="checkbox"/> sea-webserver-1	Azure subscription 1	sea-webapp-rg	Southeast Asia	Running	Linux
<input type="checkbox"/> sea-webserver-2	Azure subscription 1	sea-webapp-rg	Southeast Asia	Running	Linux
<input type="checkbox"/> sea-webserver-3	Azure subscription 1	sea-webapp-rg	Southeast Asia	Running	Linux

4. Network Interface

Network Interface adalah komponen virtual machine yang menghubungkan virtual machine ke virtual network. Setiap Virtual machine dalam Azure membutuhkan setidaknya satu network interface agar dapat berkomunikasi dengan jaringan.

Network interface SEA

```
resource "azurerm_network_interface" "sea_nic" {
  count                = 3
  provider             = azurerm.sea_provider
  name                 = "sea-webserver-nic-${count.index + 1}"
  location             = azurerm_resource_group.sea_resource_group.location
  resource_group_name  = azurerm_resource_group.sea_resource_group.name

  ip_configuration {
    name                  = "internal"
    subnet_id            = azurerm_subnet.sea_subnet.id
    private_ip_address_allocation = "Dynamic"
  }
}
```

Network interface JP

```
resource "azurerm_network_interface" "jp_nic" {
```



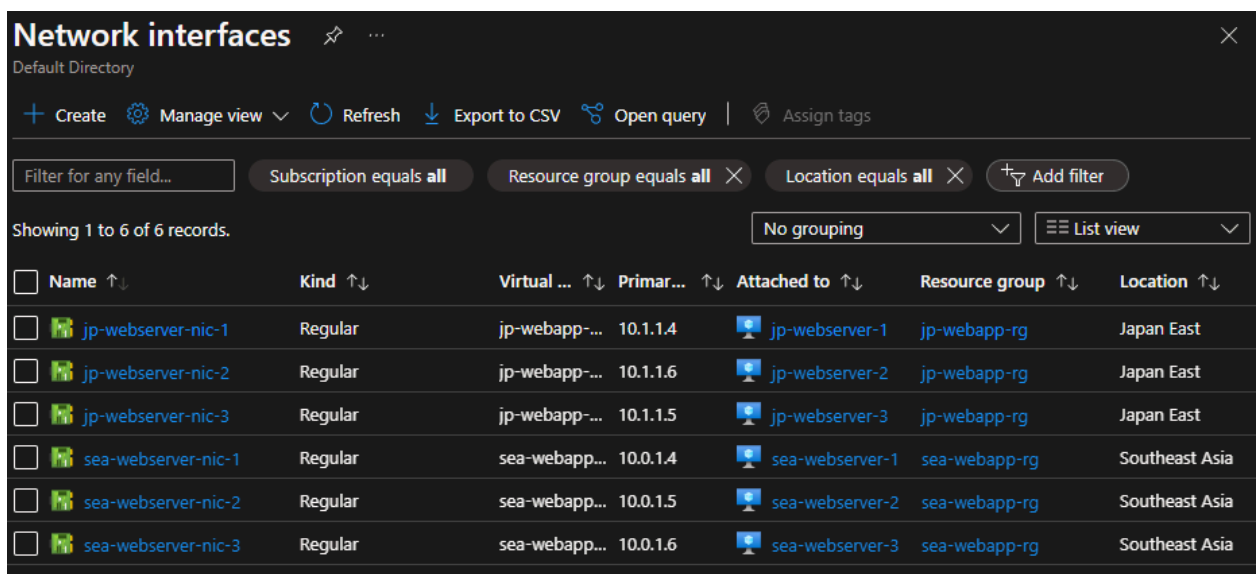
```

count                = 3
provider              = azurerm.jp_provider
name                  = "jp-webserver-nic-${count.index + 1}"
  location              =
azurerm_resource_group.jp_resource_group.location
  resource_group_name  =
azurerm_resource_group.jp_resource_group.name

  ip_configuration {
    name                = "internal"
    subnet_id           = azurerm_subnet.jp_subnet.id
    private_ip_address_allocation = "Dynamic"
  }
}

```

Network Interfaces yang telah dibuat



The screenshot shows the 'Network interfaces' page in the Azure portal. It displays a table with 6 records. The table columns are: Name, Kind, Virtual IP address, Primary IP address, Attached to, Resource group, and Location. The first three interfaces are in the 'jp-webapp-rg' resource group in 'Japan East' location, and the last three are in the 'sea-webapp-rg' resource group in 'Southeast Asia' location.

Name	Kind	Virtual IP address	Primary IP address	Attached to	Resource group	Location
jp-webserver-nic-1	Regular	jp-webapp-...	10.1.1.4	jp-webserver-1	jp-webapp-rg	Japan East
jp-webserver-nic-2	Regular	jp-webapp-...	10.1.1.6	jp-webserver-2	jp-webapp-rg	Japan East
jp-webserver-nic-3	Regular	jp-webapp-...	10.1.1.5	jp-webserver-3	jp-webapp-rg	Japan East
sea-webserver-nic-1	Regular	sea-webapp...	10.0.1.4	sea-webserver-1	sea-webapp-rg	Southeast Asia
sea-webserver-nic-2	Regular	sea-webapp...	10.0.1.5	sea-webserver-2	sea-webapp-rg	Southeast Asia
sea-webserver-nic-3	Regular	sea-webapp...	10.0.1.6	sea-webserver-3	sea-webapp-rg	Southeast Asia

5. Network Security Group

Network Security Group adalah komponen keamanan jaringan yang memungkinkan kendali atas network traffic inbound atau outbound untuk sumber daya Azure seperti Virtual Machines, Virtual Network, atau Network Interfaces. Network Security Group bekerja berdasarkan security rules yang didefinisikan untuk menentukan apakah suatu lalu lintas diizinkan atau ditolak.

Untuk Kepentingan proyek, Security rule yang ditetapkan dalam Network Security Group adalah:

- Allow-HTTP (Inbound)
- Allow-SSH (Inbound)
- Allow-outbound-Internet (Outbound)

Network Security Group SEA

```
resource "azurerm_network_security_group" "webapp_nsg" {
  provider      = azurerm.sea_provider
  name          = "webapp-nsg"
  location      = azurerm_resource_group.sea_resource_group.location
  resource_group_name = azurerm_resource_group.sea_resource_group.name

  # Definisi Security Rule Inbound/Outbound untuk Web Servers (VM)
  security_rule {
    name                = "allow-http"
    priority             = 100
    direction           = "Inbound"
    access              = "Allow"
    protocol             = "Tcp"
    source_port_range    = "*"
    destination_port_range = "80"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }

  security_rule {
    name                = "allow-ssh"
    priority             = 110
    direction           = "Inbound"
    access              = "Allow"
    protocol             = "Tcp"
    source_port_range    = "*"
    destination_port_range = "22"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }

  security_rule {
    name                = "allow-outbound-internet"
    direction           = "Outbound"
    access              = "Allow"
    protocol             = "Any"
    source_port_range    = "*"
    destination_port_range = "*"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }
}
```

```

priority          = 120
direction         = "Outbound"
access           = "Allow"
protocol         = "*"
source_port_range = "*"
destination_port_range = "*"
source_address_prefix = "*"
destination_address_prefix = "Internet"
}
}

```

Network Security Group JP

```

resource "azurerm_network_security_group" "webapp_nsg_jp" {
  provider      = azurerm.jp_provider
  name          = "webapp-nsg"
  location      = azurerm_resource_group.jp_resource_group.location
  resource_group_name = azurerm_resource_group.jp_resource_group.name

  # Definisi Security Rule Inbound/Outbound untuk Web Servers (VM)
  # pada region JP
  security_rule {
    name          = "allow-http"
    priority      = 100
    direction     = "Inbound"
    access        = "Allow"
    protocol      = "Tcp"
    source_port_range = "*"
    destination_port_range = "80"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }

  security_rule {
    name          = "allow-ssh"
    priority      = 110
    direction     = "Inbound"
  }
}

```

```

    access                = "Allow"
    protocol               = "Tcp"
    source_port_range      = "*"
    destination_port_range = "22"
    source_address_prefix  = "*"
    destination_address_prefix = "*"
  }

  security_rule {
    name                = "allow-outbound-internet"
    priority             = 120
    direction           = "Outbound"
    access              = "Allow"
    protocol             = "*"
    source_port_range    = "*"
    destination_port_range = "*"
    source_address_prefix = "*"
    destination_address_prefix = "Internet"
  }
}

```

Tampilan Security Rules pada Network Security Group

Priority ↑↓	Name ↑↓	Port ↑↓	Protocol ↑↓	Source ↑↓	Destination ↑↓	Action ↑↓
Inbound Security Rules						
100	allow-http	80	Tcp	Any	Any	✓ Allow
110	allow-ssh	22	Tcp	Any	Any	✓ Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	✓ Allow
65500	DenyAllInBound	Any	Any	Any	Any	✗ Deny
Outbound Security Rules						
120	allow-outbound-internet	Any	Any	Any	Internet	✓ Allow
65000	AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowInternetOutBound	Any	Any	Any	Internet	✓ Allow
65500	DenyAllOutBound	Any	Any	Any	Any	✗ Deny

6. Load Balancer

Load Balancer adalah layanan yang mendistribusikan traffic jaringan dari internet ke sumber daya Azure, seperti Virtual Machines, yang ada di belakangnya (backend pool). Load Balancer memiliki alamat IP Publik yang dapat diakses dari luar jaringan Azure. Load balancer memastikan aplikasi tetap tersedia dengan mendistribusikan lalu lintas ke sumber daya yang aktif dan sehat. Sehingga jika salah satu backend gagal, load balancer akan mengarahkan traffic ke backend (VM) lain yang masih aktif.

Load Balancer SEA

```
resource "azurerm_lb" "sea_load_balancer" {
  provider      = azurerm.sea_provider
  name          = "sea-webapp-lb"
  location      = azurerm_resource_group.sea_resource_group.location
  resource_group_name = azurerm_resource_group.sea_resource_group.name
  sku           = "Standard"

  frontend_ip_configuration {
    name          = "sea-lb-frontend"
    public_ip_address_id = azurerm_public_ip.sea_lb_public_ip.id
  }
}
```

Load Balancer JP

```
resource "azurerm_lb" "jp_load_balancer" {
  provider      = azurerm.jp_provider
  name          = "jp-webapp-lb"
  location      = azurerm_resource_group.jp_resource_group.location
  resource_group_name = azurerm_resource_group.jp_resource_group.name
  sku           = "Standard"

  frontend_ip_configuration {
    name          = "jp-lb-frontend"
    public_ip_address_id = azurerm_public_ip.jp_lb_public_ip.id
  }
}
```

Load Balancer membutuhkan komponen-komponen lain seperti yang disebutkan dibawah ini, agar dapat bekerja.

a. Frontend IP Address (Public IP)

Public IP Address adalah alamat IP unik yang memungkinkan Load Balancer menerima traffic dari internet. Alamat ini digunakan sebagai antarmuka publik untuk mengarahkan traffic ke backend resources seperti Virtual Machines pada

Azure. Public IP akan digunakan sebagai Frontend IP Address dari load balancer yang akan dikonfigurasi.

Frontend IP Address SEA

```
# Public IP untuk Load Balancer pada region Southeast Asia
resource "azurerm_public_ip" "sea_lb_public_ip" {
  provider      = azurerm.sea_provider
  name          = "sea-lb-publicip"
  location      = azurerm_resource_group.sea_resource_group.location
  resource_group_name = azurerm_resource_group.sea_resource_group.name
  allocation_method = "Static"
  sku           = "Standard"
}
```

```
frontend_ip_configuration {
  name = "sea-lb-frontend"
  public_ip_address_id = azurerm_public_ip.sea_lb_public_ip.id
}
```

Frontend IP Address JP

```
# Public IP untuk Load Balancer region Japan East
resource "azurerm_public_ip" "jp_lb_public_ip" {
  provider      = azurerm.jp_provider
  name          = "jp-lb-publicip"
  location      = azurerm_resource_group.jp_resource_group.location
  resource_group_name = azurerm_resource_group.jp_resource_group.name
  allocation_method = "Static"
  sku           = "Standard"
}
```

```
frontend_ip_configuration {
  name = "jp-lb-frontend"
}
```

```

                                public_ip_address_id      =
azurerm_public_ip.jp_lb_public_ip.id
    }

```

b. Backend Address Pool

Backend Address Pool adalah kumpulan resources di mana traffic jaringan yang diterima load balancer akan didistribusikan. Resources yang diletakkan dalam backend pool dapat berupa Virtual Machine, Virtual Machine Scale Sets, atau layanan lain yang berada di dalam virtual network. Jika salah satu resources pada backend pool tidak aktif, load balancer secara otomatis mendistribusikan traffic ke sumber daya lain yang sehat/running.

Backend Pool SEA

```

# Load Balancer Backend Address Pool
resource "azurerm_lb_backend_address_pool" "sea_backend_pool"
{
    provider          = azurerm.sea_provider
    loadbalancer_id   = azurerm_lb.sea_load_balancer.id
    name              = "sea-backend-pool"
}

```

```

# Asosiasi VM's dengan Backend Pool Load balancer pada
Southeast Asia
resource
"azurerm_network_interface_backend_address_pool_association"
"sea_backend_pool_assoc" {
    count              = 3
    provider           = azurerm.sea_provider
    network_interface_id =
azurerm_network_interface.sea_nic[count.index].id
    ip_configuration_name = "internal"
    backend_address_pool_id =
azurerm_lb_backend_address_pool.sea_backend_pool.id
}

```

Backend...	Resourc...	IP address	Network interface	Availabil...	Rules co...	Resource St...	Admin s...
<div> <div>sea-backend-pool (3)</div> <div></div> </div>							
sea-backend-pool	sea-webserver-2	10.0.1.5	sea-webserver-nic-2	2	1	Running	None
sea-backend-pool	sea-webserver-1	10.0.1.4	sea-webserver-nic-1	1	1	Running	None
sea-backend-pool	sea-webserver-3	10.0.1.6	sea-webserver-nic-3	3	1	Running	None

Backend Pool JP

```
# Load Balancer Backend Address Pool
resource "azurerm_lb_backend_address_pool" "jp_backend_pool" {
  provider           = azurerm.jp_provider
  loadbalancer_id    = azurerm_lb.jp_load_balancer.id
  name               = "jp-backend-pool"
}
```

```
# Asosiasi VM dengan Backend Pool Load Balancer pada Japan East
resource "azurerm_network_interface_backend_address_pool_association" "jp_backend_pool_assoc" {
  count               = 3
  provider            = azurerm.jp_provider
  network_interface_id = azurerm_network_interface.jp_nic[count.index].id
  ip_configuration_name = "internal"
  backend_address_pool_id = azurerm_lb_backend_address_pool.jp_backend_pool.id
}
```

Backend pool	Resource Name	IP address	Networ...	Availabi...	Rules co...	Resourc...	Admin s...
<div> <div>jp-backend-pool (3)</div> <div></div> </div>							
jp-backend-pool	jp-webserver-2	10.1.1.6	jp-webserver-nic-2	3	1	Running	None
jp-backend-pool	jp-webserver-3	10.1.1.5	jp-webserver-nic-3	2	1	Running	None
jp-backend-pool	jp-webserver-1	10.1.1.4	jp-webserver-nic-1	2	1	Running	None

c. Health Probe

Health Probe adalah komponen load balancer yang memiliki mekanisme untuk memantau status backend pool untuk memastikan bahwa hanya resources yang sehat yang akan menerima traffic. Jika sebuah resource gagal atau tidak responsif, load balancer akan berhenti mengirimkan lalu lintas ke sumber daya tersebut sampai dinyatakan sehat kembali.

Health probe akan mengirimkan request (TCP, HTTP, atau HTTPS) ke sebuah endpoint yang ditentukan pada backend untuk memeriksa apakah VM akan merespons dengan benar atau tidak. Interval pengujian menggunakan request tersebut dapat dikonfigurasi, namun biasanya dilakukan setiap 5 hingga 30 detik.

Health Probe SEA

```
# Health Probe untuk Load Balancer Southeast Asia
resource "azurerm_lb_probe" "sea_health_probe" {
  provider            = azurerm.sea_provider
  loadbalancer_id     = azurerm_lb.sea_load_balancer.id
  name                = "sea-health-probe"
  port                = 80
  protocol             = "Http"
  request_path        = "/"
  interval_in_seconds = 15
  number_of_probes    = 3
}
```

Health Probe JP

```
# Health Probe untuk Load Balancer Japan East
resource "azurerm_lb_probe" "jp_health_probe" {
  provider           = azurerm.jp_provider
  loadbalancer_id    = azurerm_lb.jp_load_balancer.id
  name               = "jp-health-probe"
  port               = 80
  protocol            = "Http"
  request_path        = "/"
  interval_in_seconds = 15
  number_of_probes    = 3
}
```

Resourc...	IP Address	State	Reason
jp-webserver-2	10.1.1.6	✓ Up	The backend instance is responding to health probe successfully.
jp-webserver-3	10.1.1.5	✓ Up	The backend instance is responding to health probe successfully.
jp-webserver-1	10.1.1.4	✓ Up	The backend instance is responding to health probe successfully.

d. Load Balancing Rule

Load Balancing Rule, adalah konfigurasi yang menentukan bagaimana traffic yang diterima oleh Frontend IP Address harus didistribusikan ke resources yang ada di Backend Pool. Peraturan ini mengatur berbagai aspek dari cara traffic diarahkan, seperti port, protokol, dan jenis distribusi beban.

Load Balancing rule SEA

```
# Load Balancing Rule untuk Load Balancer Southeast Asia
resource "azurerm_lb_rule" "sea_lb_rule" {
  provider           = azurerm.sea_provider
  loadbalancer_id    = azurerm_lb.sea_load_balancer.id
  name               = "sea-lb-rule"
  protocol            = "Tcp"
  frontend_port       = 80
}
```

```

        backend_port = 80
        frontend_ip_configuration_name = "sea-lb-frontend"
        probe_id =
azurerm_lb_probe.sea_health_probe.id
        backend_address_pool_ids =
[azurerm_lb_backend_address_pool.sea_backend_pool.id]
    }

```

Load Balancing rule JP

```

# Load Balancing Rule untuk Load Balancer Japan East
resource "azurerm_lb_rule" "jp_lb_rule" {
    provider = azurerm.jp_provider
    loadbalancer_id =
azurerm_lb.jp_load_balancer.id
    name = "jp-lb-rule"
    protocol = "Tcp"
    frontend_port = 80
    backend_port = 80
    frontend_ip_configuration_name = "jp-lb-frontend"
    probe_id =
azurerm_lb_probe.jp_health_probe.id
    backend_address_pool_ids =
[azurerm_lb_backend_address_pool.jp_backend_pool.id]
}

```

Kedua Load Balancing Rule diatas akan mendengar traffic melalui port 80 HTTP, dan mengarahkannya ke backend melalui port 80 HTTP.

7. Storage Account

Storage account adalah layanan dalam Azure yang menyediakan akses ke berbagai jenis layanan penyimpanan data yang aman dan skalabel pada cloud. Storage Account dibutuhkan untuk membuat penyimpanan Blob Storage, File Storage, Table Storage, Queue Storage, dan Disk Storage. Untuk mengakses layanan penyimpanan apapun dalam sebuah storage account, dibutuhkan access keys, atau shared access signature (SAS) sebagai fitur keamanan.

Storage Account SEA

```

resource "azurerm_storage_account" "sea_storage" {
    provider = azurerm.sea_provider

```

```

    name
    "seawebappstorage${random_string.storage_suffix.result}"
    resource_group_name
    azurerm_resource_group.sea_resource_group.name
    location
    azurerm_resource_group.sea_resource_group.location
    account_tier          = "Standard"
    account_replication_type = "ZRS"
}

```

Storage Account JP

```

resource "azurerm_storage_account" "jp_storage" {
  provider          = azurerm.jp_provider
  name
  "jpwebappstorage${random_string.storage_suffix.result}"
  resource_group_name
  azurerm_resource_group.jp_resource_group.name
  location
  azurerm_resource_group.jp_resource_group.location
  account_tier      = "Standard"
  account_replication_type = "ZRS"
}

```

Storage Account menggunakan replikasi tipe “ZRS” atau Zone Redundant Storage, sehingga jika terjadi datacenter failure, storage akan tetap accessible.

8. File Share Storage

Azure File Storage adalah layanan penyimpanan yang menyediakan file shares yang dapat diakses melalui protokol SMB (Server Message Block) atau NFS (Network File System). Layanan ini memungkinkan sebuah file untuk diakses dan dibagi oleh beberapa Virtual Machines, aplikasi, atau pengguna, seperti berbagi file pada server tradisional.

File Storage SEA

```

resource "azurerm_storage_share" "sea_fileshare" {
  provider          = azurerm.sea_provider
  name              = "sea-webapp-share"
  storage_account_name = azurerm_storage_account.sea_storage.name
  quota             = 50
}

```

File Storage JP

```
resource "azurerm_storage_share" "jp_fileshare" {  
  provider      = azurerm.jp_provider  
  name          = "jp-webapp-share"  
  storage_account_name = azurerm_storage_account.jp_storage.name  
  quota         = 50  
}
```

Quota = 50, adalah definisi ukuran file share yang akan dibuat. Sehingga file share yang akan dibuat dan digunakan akan memiliki size 50 GB.

9. Mounting Script (Bash)

```
locals {  
  sea_mount_script = <<-SCRIPT  
#!/bin/bash  
# Mount Azure File Share  
mkdir -p /mnt/sea-webapp-share  
mount -t cifs  
//${azurerm_storage_account.sea_storage.name}.file.core.windows.net/  
sea-webapp-share /mnt/sea-webapp-share \  
-o  
vers=3.0,username=${azurerm_storage_account.sea_storage.name},passwo  
rd=${azurerm_storage_account.sea_storage.primary_access_key},dir_mod  
e=0777,file_mode=0777,serverino  
echo  
"//${azurerm_storage_account.sea_storage.name}.file.core.windows.net  
/sea-webapp-share /mnt/sea-webapp-share cifs  
nofail,vers=3.0,username=${azurerm_storage_account.sea_storage.name}  
,password=${azurerm_storage_account.sea_storage.primary_access_key},  
dir_mode=0777,file_mode=0777,serverino 0 0" >> /etc/fstab  
SCRIPT  
  
  jp_mount_script = <<-SCRIPT  
#!/bin/bash  
# Mount Azure File Share  
mkdir -p /mnt/jp-webapp-share  
mount -t cifs  
//${azurerm_storage_account.jp_storage.name}.file.core.windows.net/j  
p-webapp-share /mnt/jp-webapp-share \  

```

```

-o
vers=3.0,username=${azurerm_storage_account.jp_storage.name},password=${azurerm_storage_account.jp_storage.primary_access_key},dir_mode=0777,file_mode=0777,serverino
echo
"//${azurerm_storage_account.jp_storage.name}.file.core.windows.net/jp-webapp-share /mnt/jp-webapp-share cifs nofail,vers=3.0,username=${azurerm_storage_account.jp_storage.name},password=${azurerm_storage_account.jp_storage.primary_access_key},dir_mode=0777,file_mode=0777,serverino 0 0" >> /etc/fstab
SCRIPT
}

```

Bash Script diatas digunakan untuk mounting Azure File Share dari Storage Account menggunakan protokol Common Internet File System (CIFS).

10. Template File -> vm-setup-script.sh

```

# Konfigurasi Template File untuk menjalankan VM Setup Script Saat Infrastruktur dideploy

                                custom_data                                =
base64encode(templatefile("${path.module}/vm-setup-script.sh", {
    STORAGE_ACCOUNT_NAME = azurerm_storage_account.jp_storage.name
                                STORAGE_ACCOUNT_KEY                                =
azurerm_storage_account.jp_storage.primary_access_key
    FILE_SHARE_NAME      = azurerm_storage_share.jp_fileshare.name
    REGION                = "JP"
    SERVER_NUMBER         = count.index + 1
})

```

Potongan kode diatas adalah sebuah konfigurasi custom_data yang digunakan untuk pass startup script pada deployment Virtual Machine. base64encode() adalah fungsi untuk encode isi script ke format base64. Fungsi templatefile() membaca template file bernama vm-setup-script.sh dan memungkinkan variable untuk di interpolasi pada script. Aksi yang akan dilakukan oleh fungsi diatas didefinisikan didalam file vm-setup-script.sh.

11. Virtual Machine Extension

```

# Instalasi Extension cifs Pada VM's SEA
resource "azurerm_virtual_machine_extension" "sea_cifs_install" {

```

```

count                = 3
provider             = azurerm.sea_provider
name                 = "install-cifs-${count.index}"
                    virtual_machine_id =
azurerm_linux_virtual_machine.sea_webserver[id]
publisher            = "Microsoft.Azure.Extensions"
type                 = "CustomScript"
type_handler_version = "2.1"

settings = <<SETTINGS
{
                                "commandToExecute":
"#!/bin/bash\n\nMAX_RETRIES=5\nRETRY_DELAY=30\n\nfor i in $(seq 1
$MAX_RETRIES); do\n    apt-get update -y && apt-get install -y
cifs-utils && break\n    echo \"Attempt $i failed. Waiting
$RETRY_DELAY seconds...\"\n    sleep $RETRY_DELAY\ndone\n\nif [ $i -gt
$MAX_RETRIES ]; then\n    echo \"Failed to install cifs-utils after
$MAX_RETRIES attempts\"\n    exit 1\nfi"
}
SETTINGS
}

```

```

# Instalasi Extension cifs Pada VM's JP
resource "azurerm_virtual_machine_extension" "jp_cifs_install" {
count                = 3
provider             = azurerm.jp_provider
name                 = "install-cifs-${count.index}"
                    virtual_machine_id =
azurerm_linux_virtual_machine.jp_webserver[id]
publisher            = "Microsoft.Azure.Extensions"
type                 = "CustomScript"
type_handler_version = "2.1"

settings = <<SETTINGS
{
                                "commandToExecute":
"#!/bin/bash\n\nMAX_RETRIES=5\nRETRY_DELAY=30\n\nfor i in $(seq 1
$MAX_RETRIES); do\n    apt-get update -y && apt-get install -y
cifs-utils && break\n    echo \"Attempt $i failed. Waiting

```

```

$RETRY_DELAY seconds...\n    sleep $RETRY_DELAY\ndone\n\nif [ $i -gt
$MAX_RETRIES ]; then\n    echo \"Failed to install cifs-utils after
$MAX_RETRIES attempts\"\n    exit 1\nfi"
    }
    SETTINGS
}

```

Block kode diatas mendefinisikan Extension Azure Virtual Machine untuk menginstall utilities Common Internet File System (CIFS) pada Virtual Machines.

12. Virtual Machine Backup

VM Backup SEA

```

resource "azurerm_backup_protected_vm" "sea_vm_backups" {
  count                = 3
  provider             = azurerm.sea_provider
  resource_group_name =
    azurerm_resource_group.sea_resource_group.name
  recovery_vault_name =
    azurerm_recovery_services_vault.backup_vault_sea.name
  source_vm_id        =
    azurerm_linux_virtual_machine.sea_webserver[count.index].id
  backup_policy_id     =
    azurerm_backup_policy_vm.daily_backup_policy_sea.id
}

```

Tampilan VM's yang sudah dibackup pada region SEA

Backup Items (Azure Virtual Machine) ...					
sea-webapp-backup-vault					
Refresh + Add Filter Feedback					
Try our new Business Continuity Center for the at scale BCDR management of your resources protected across Azure Backup and Site Recovery. →					
All data fetched from the service.					
<input type="text" value="Filter items ..."/>					
Name ↑↓	Resource Group ↑↓	Backup Pre-Check	Last Backup Status	Latest restore p... ↑↓	Details
sea-webserver-1	sea-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details
sea-webserver-2	sea-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details
sea-webserver-3	sea-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details

VM Backup JP


```
resource "azurerm_backup_protected_vm" "jp_vm_backups" {
  count                = 3
  provider             = azurerm.jp_provider
  resource_group_name =
azurerm_resource_group.jp_resource_group.name
  recovery_vault_name =
azurerm_recovery_services_vault.backup_vault_jp.name
  source_vm_id        =
azurerm_linux_virtual_machine.jp_webserver[count.index].id
  backup_policy_id    =
azurerm_backup_policy_vm.daily_backup_policy_jp.id
}
```

Tampilan VM's yang sudah di backup pada region JP

Backup Items (Azure Virtual Machine) ...

jp-webapp-backup-vault

Refresh

+ Add

Filter

Feedback

Try our new Business Continuity Center for the at scale BCDR management of your resources protected across Azure Backup and Site Recovery. →

✓ All data fetched from the service.

Filter items ...

Name ↑↓	Resource Group ↑↓	Backup Pre-Check	Last Backup Status	Latest restor... ↑↓	Details
jp-webserver-1	jp-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details
jp-webserver-2	jp-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details
jp-webserver-3	jp-webapp-rg	✓ Passed	⚠ Warning (Initial backup pending)		View details

Blok Code diatas akan mengkonfigurasi backup untuk 3 Virtual machines yang ada, dan setiap VM akan di backup secara konfigurasi yang ditetapkan pada backup policy. Backup akan disimpan pada recovery services vault yang ada di masing-masing region.

a. Backup Policy

Backup Policy dalam Azure Backup adalah konfigurasi yang mengatur cara, kapan, dan seberapa lama data cadangan (backup) disimpan dalam Recovery Services Vault. Kebijakan ini dirancang untuk mengotomatisasi proses pencadangan dan memastikan data terlindungi sesuai dengan kebutuhan organisasi atau proyek.

Backup Policy SEA

```
resource "azurerm_backup_policy_vm" "daily_backup_policy_sea"
{
```

```

provider          = azurerm.sea_provider
name              = "sea-daily-backup-policy"
                  resource_group_name          =
azurerm_resource_group.sea_resource_group.name
                  recovery_vault_name          =
azurerm_recovery_services_vault.backup_vault_sea.name

  backup {
    frequency = "Daily"
    time      = "23:00"
  }

  retention_daily {
    count = 7
  }
}

```

Backup Policy JP

```

resource "azurerm_backup_policy_vm" "daily_backup_policy_jp" {
  provider          = azurerm.jp_provider
  name              = "jp-daily-backup-policy"
                  resource_group_name          =
azurerm_resource_group.jp_resource_group.name
                  recovery_vault_name          =
azurerm_recovery_services_vault.backup_vault_jp.name

  backup {
    frequency = "Daily"
    time      = "23:00"
  }

  retention_daily {
    count = 7
  }
}

```

Backup policy diatas akan mengonfigurasi backup yang akan dilakukan dengan frekuensi daily dan pada waktu 23:00 setiap hari. Retention policy count

= 7, berarti setiap backup akan disimpan selama 7 hari dan setelah 7 hari, backup dengan umur tertua akan di delete secara otomatis.

13. Recovery Services Vault

Recovery Services Vault adalah layanan penyimpanan berbasis cloud yang dirancang untuk mengelola dan menyimpan data cadangan (backup) serta konfigurasi pemulihan (recovery) pada Azure.

RSV SEA

```
resource "azurerm_recovery_services_vault" "backup_vault_sea" {
  provider      = azurerm.sea_provider
  name          = "sea-webapp-backup-vault"
  location      = azurerm_resource_group.sea_resource_group.location
  resource_group_name = azurerm_resource_group.sea_resource_group.name
  sku           = "Standard"
}
```

RSV JP

```
resource "azurerm_recovery_services_vault" "backup_vault_jp" {
  provider      = azurerm.jp_provider
  name          = "jp-webapp-backup-vault"
  location      = azurerm_resource_group.jp_resource_group.location
  resource_group_name = azurerm_resource_group.jp_resource_group.name
  sku           = "Standard"
}
```

B. Initial Configuration

vm-setup-script.sh -> File Konfigurasi Setup VM yang akan dijalankan dengan Terraform.

File mengandung bash script yang akan dijalankan di dalam Virtual Machine Linux yang akan mengkonfigurasi Web App menggunakan NGINX, termasuk konfigurasi page HTML yang akan ditampilkan.

C. Deployment

Untuk melakukan Deployment dari Terraform Template yang telah dibuat pada terraform, dibutuhkan tiga langkah:

1. Terraform Validate

Command 'terraform validate' akan mengecek syntax dan konsistensi internal dari file konfigurasi template Terraform. Selain itu command ini juga memverifikasi bahwa konfigurasi valid secara syntactically dan konsisten secara internal. Error seperti konfigurasi resource yang salah, error syntax, argument required yang hilang, dan tipe argumen invalid akan dideteksi.

```
PS C:\Users\rocke\OneDrive\Desktop\VSCODE Project\TerraformProject> terraform validate
Success! The configuration is valid.
```

2. Terraform Plan

Command ini akan membuat rencana eksekusi dan menampilkan perubahan apa yang akan dibuat kepada infrastruktur. Command akan membandingkan current state dengan desired state pada konfigurasi dengan cara berhubung dengan cloud provider untuk memeriksa current state dari infrastructure. Command ini akan memberi preview detail dari:

- Resource yang akan dibuat
- Resource yang akan dimodifikasi
- Resource yang akan dirusak

Sehingga memungkinkan user untuk mereview dan konfirmasi perubahan sebelum applying/deploying.

3. Terraform Apply

Adalah command terraform yang akan menerapkan perubahan yang telah didefinisikan dalam terraform template menggunakan execution plan yang telah ditetapkan oleh command terraform plan.

```
PS C:\Users\rocke\OneDrive\Desktop\VSCODE Project\TerraformProject> terraform apply "deploy"
azure_rm_resource_group.jp_resource_group: Creating...
azure_rm_resource_group.sea_resource_group: Creating...
azure_rm_resource_group.jp_resource_group: Still creating... [10s elapsed]
azure_rm_resource_group.sea_resource_group: Still creating... [10s elapsed]
azure_rm_resource_group.jp_resource_group: Creation complete after 11s [id=/subscriptions/773262d6-6b19-4668-95ac-006e06cbda65/resourceGroups/jp-webapp-rg]
azure_rm_public_ip.jp_lb_public_ip: Creating...
azure_rm_virtual_network.jp_vnet: Creating...
```

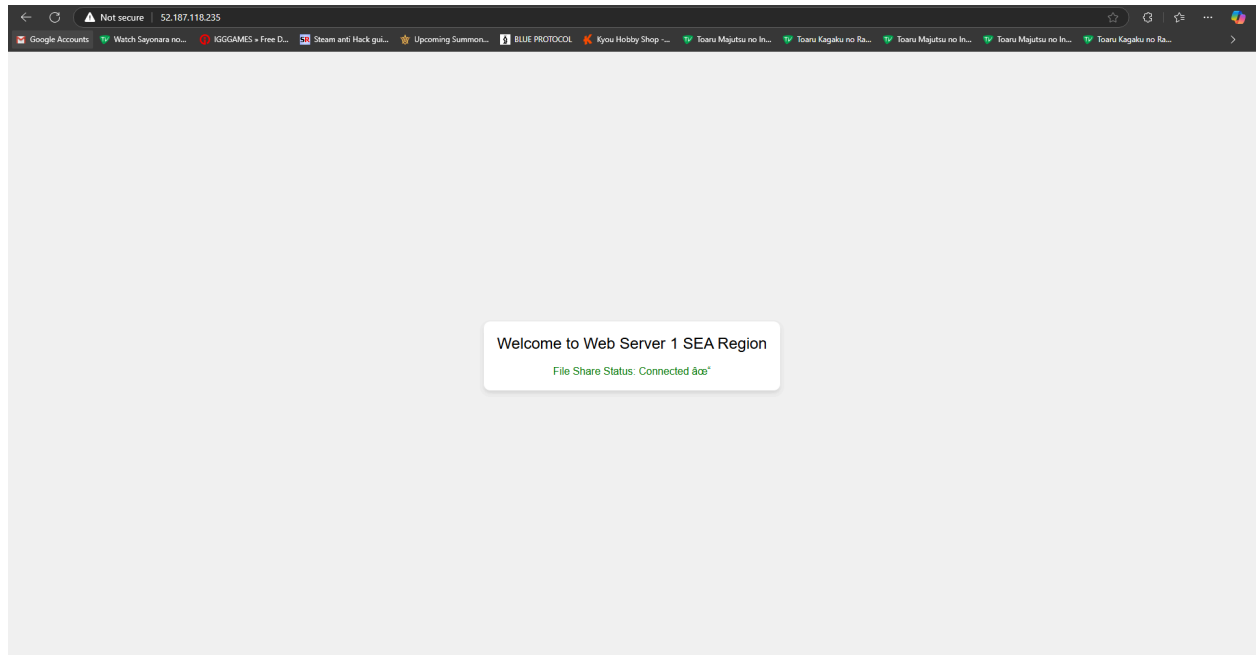
```
Apply complete! Resources: 62 added, 0 changed, 0 destroyed.
```

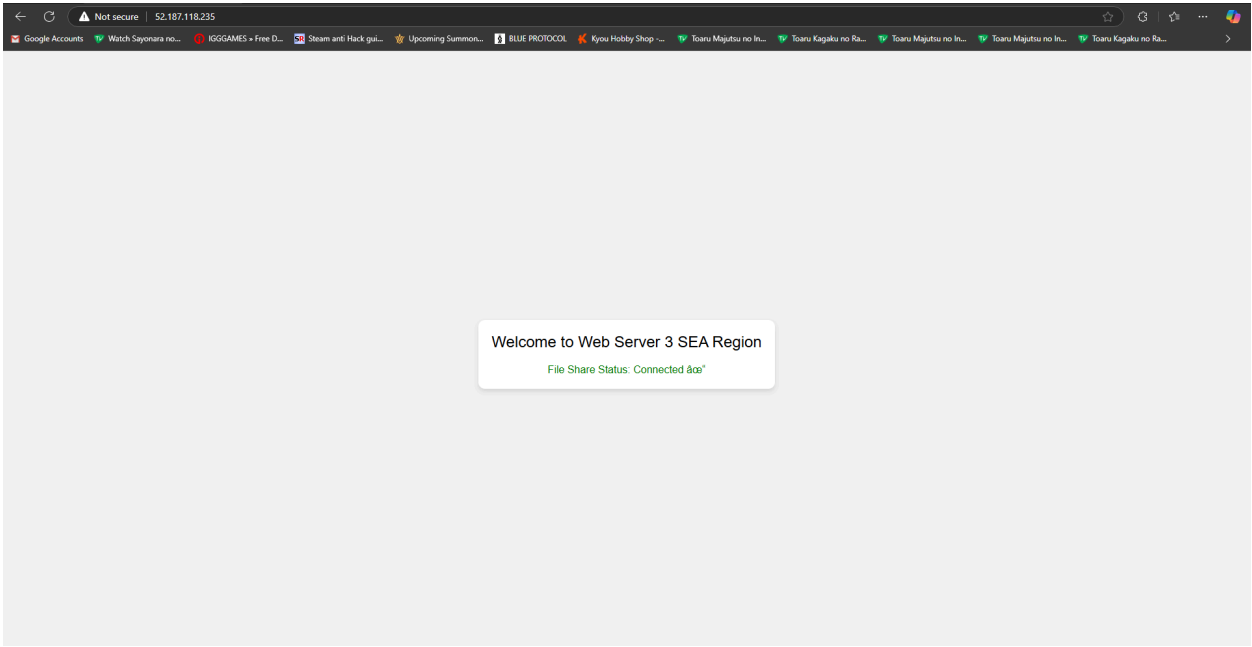
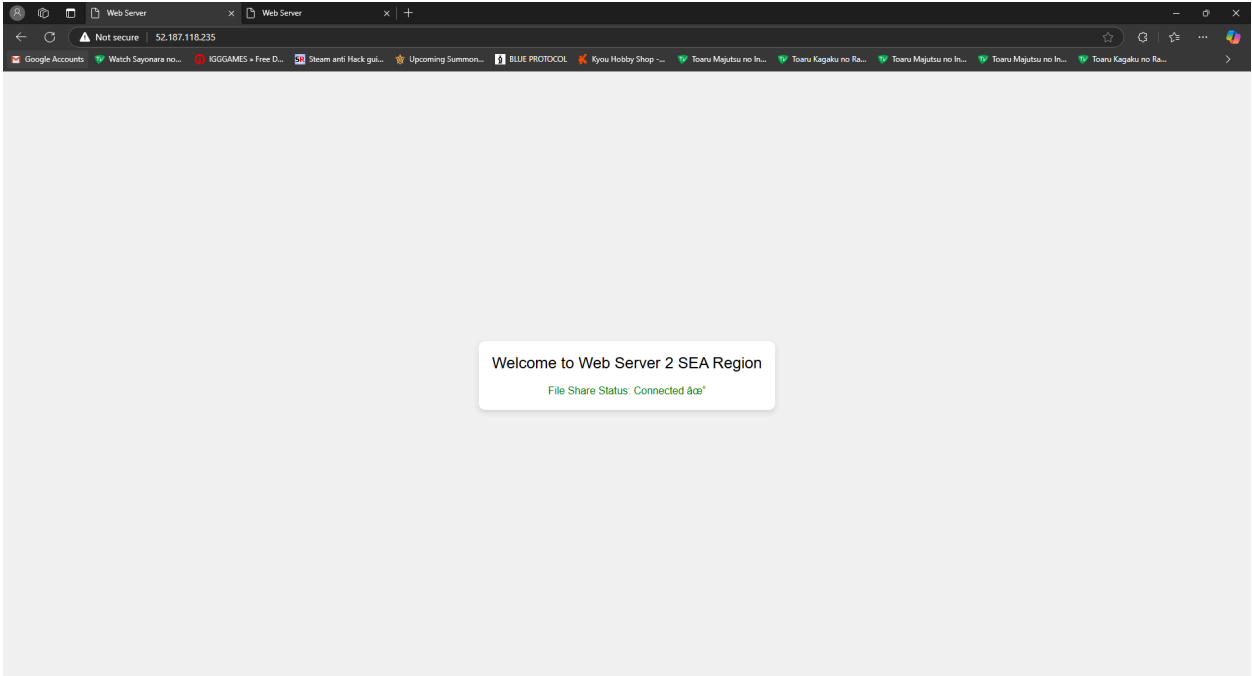
Outputs:

```
jp_load_balancer_ip = "52.155.124.227"  
sea_load_balancer_ip = "52.187.118.235"
```

D. Output

Tampilan Web Server pada region Southeast Asia





Tampilan Web Server pada region Japan East

