

3-Tier Architecture

Introduction:

3-Tier Architecture is a widely used cloud application design pattern that separates an application into three logical layers: Presentation Tier, Application Tier, and Data Tier. This architecture improves scalability, security, and maintainability by isolating user interaction, business logic, and data storage into independent layers.

1. Create Resource Groups:

A Resource Group is a logical container in Azure that groups and manages all related resources for an application.

Resource Group: sai3-tierRG

The screenshot shows the Microsoft Azure Resource Manager console. The main heading is 'Resource Manager | Resource groups'. Below this, there's a search bar and a list of resource groups. The resource groups listed are:

Name	Subscription	Location
NetworkWatcherRG	Azure subscription 1	Australia Southeast
RGAJAY1	Azure subscription 1	Australia East
sai3-tierRG	Azure subscription 1	Korea Central

The 'sai3-tierRG' resource group is highlighted. At the bottom, it says 'Showing 1 - 3 of 3. Display count: 50'. There is also a 'Give feedback' link.

2. Create Virtual Networks and Subnets

Virtual Networks:

An Azure Virtual Network is a private network that securely connects Azure resources and controls traffic between them.

Home > Network foundation

Network foundation | Virtual networks

Preview

Search

Overview

Virtual network

Virtual Network overview

Virtual networks

NAT gateways

Public IP addresses

Network interfaces

Network security groups

Application security groups

Bastions

Create Manage view Refresh Export to CSV Open query Assign tags Group by none

You are viewing a new version of Browse experience. Click here to access the old experience.

Filter for any field... Subscription equals all Resource Group equals all Location equals all Add filter

Name	Resource Group	Location	Subscription
DB_vnet02	sai3-tierRG	Central India	Azure subscription 1
sai_VNET01	sai3-tierRG	Korea Central	Azure subscription 1
VNETAJAY1	RGAJAY1	Australia East	Azure subscription 1

Showing 1 - 3 of 3. Display count: 50

Give feedback

Add or remove favorites by pressing Ctrl+Shift+F

Subnets:

A Subnet divides an Azure Virtual Network into smaller, manageable networks for better security and organization.

Home > Network foundation | Virtual networks >

sai_VNET01 | Subnets

Virtual network

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Settings

Address space

Connected devices

Subnets

Bastion

Subnet Refresh Manage users Delete Export to CSV

Create subnets to segment the virtual network address space into smaller ranges for use by your applications. When you deploy resources into a subnet, Azure assigns the resource an IP address from the subnet.

Search subnets

Name	IPv4	IPv6	Available IPs	Delegated to	Security group	Route table
websubnet	10.0.0/24	-	250	-	-	-
appsubnet	10.0.1/24	-	250	-	-	-

Showing 2 subnets

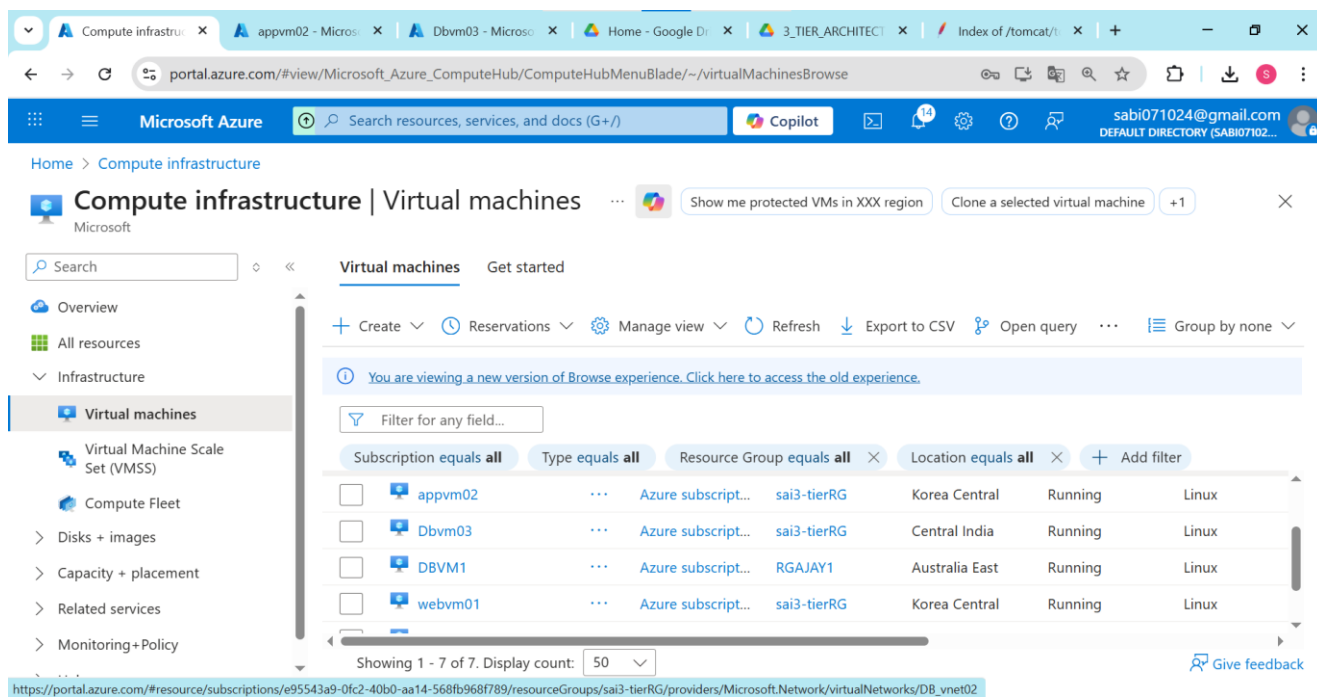
Give feedback

Add or remove favorites by pressing Ctrl+Shift+F

3. Create Virtual Machines

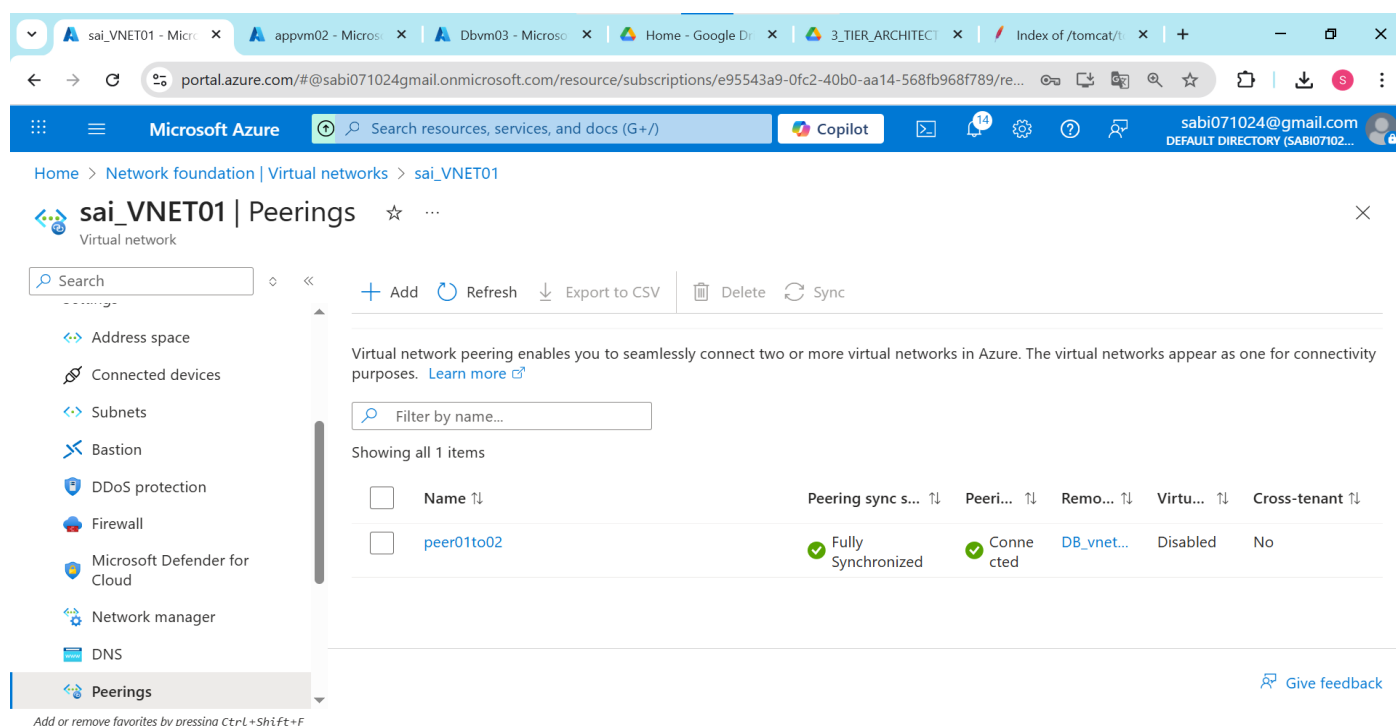
An Azure Virtual Machine is a scalable cloud server used to host applications and services.

- Web Virtual Machine
- App Virtual Machine
- Db Virtual Machine



VNet Peering:

VNet Peering connects Azure virtual networks privately, enabling secure communication between them.



4. Create Network Security Groups (NSGs)

A **Network Security Group (NSG)** is a security feature in Azure that controls inbound and outbound network traffic to Azure resources. It uses rules to allow or deny traffic based on source, destination, port, and protocol, helping protect virtual machines and subnets from unauthorized access.

webvm01 | Network settings

Virtual machine

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Connect

Networking

Network settings

Load balancing

Application security

Add or remove favorites by pressing Ctrl+Shift+F

Source == all Destination == all Protocol == all Action == all Port == all

Name	Port	Protocol	Source	Destination	Action
port rules (6)					
SSH	22	TCP	27.7.43.206	Any	Allow
allow	80	Any	Any	10.0.0.4	Allow
allow_	8080	Any	Any	Any	Allow
AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalancer	Any	Allow
DenyAllInBound	Any	Any	Any	Any	Deny
Inbound port rules (3)					

appvm02 | Network settings

Virtual machine

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Connect

Networking

Network settings

Load balancing

Application security

Add or remove favorites by pressing Ctrl+Shift+F

Source == all Destination == all Protocol == all Action == all Port == all

Name	Port	Protocol	Source	Destination	Action
Inbound port rules (4)					
Allow	8080	Any	10.0.0.4	10.0.1.4	Allow
AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalancer	Any	Allow
DenyAllInBound	Any	Any	Any	Any	Deny
Inbound port rules (3)					

Microsoft Azure portal showing the Network settings for a virtual machine named Dbvm03. The interface includes a sidebar with navigation options and a main area displaying a table of port rules.

Name	Port	Protocol	Source	Destination	Action
allow	3306	Any	10.0.1.4	10.1.2.4	Allow
deny	Any	Any	10.0.0.4	10.1.2.4	Deny
AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalancer	Any	Allow
DenyAllInBound	Any	Any	Any	Any	Deny

5. Connect Web to App Server:

Using MobaXterm, the web server securely connects to the application server in Linux, enabling seamless request processing in a 3-tier architecture.

MobaXterm interface showing a terminal session. The terminal window displays the following output:

```
root@webvm01:~# telnet 10.0.1.4 8080
Trying 10.0.1.4...
Connected to 10.0.1.4.
Escape character is '^['.
```

The interface also includes a sidebar with file explorer and a bottom status bar with system metrics.

6. Connect App to Database Server:

Using MobaXterm, the application server securely connects to the database server in Linux, enabling backend services to access stored data.

