Celebal Technologies

Summer Internship – Assignment - 6 Submission
Department: Cloud Infrastructure & Security
Internship Duration: 19th May 2025 – 20th July 2025

Assignment Title

Three Tier Architecture

Submitted by: Sanchit Mathur
Course: 5-Year Integrated M.Tech (CSE – Cyber Security)
Current Week: 6/12
Assignment: Week 6

Submitted to:
Celebal Technologies – CSI Team
Portal: [CSI Dashboard – Cloud Infra & Security]

Table of Contents

Objective

Set up a secure 3-tier architecture in Azure with strict network isolation:

- **Web Tier**: Internet-facing, accesses App tier only
- **App Tier**: Accesses Web and DB tiers, no internet access
- **DB Tier**: Completely isolated from all tiers
- Deploy 6 VMs total (2 per tier: 1 Linux with Apache, 1 Windows with IIS)

Network Design

Architecture Overview

Internet

↓

[Web Tier] ← NSG → [App Tier] ← NSG → [DB Tier]

10.0.1.0/24 10.0.2.0/24 10.0.3.0/24

CIDR Allocation

- **Virtual Network**: 10.0.0.0/16
- **Web Subnet**: 10.0.1.0/24
- **App Subnet**: 10.0.2.0/24
- **DB Subnet**: 10.0.3.0/24

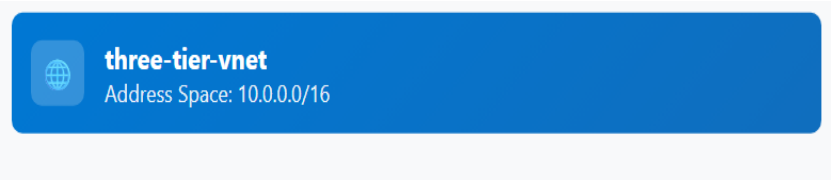

Figure 1 - Network diagram showing VNet with three subnets

Implementation Steps

Step 1: Create Virtual Network and Subnets

1. Navigate to **Virtual Networks** → **Create**
2. Configure VNet:
    - Resource Group: rg-3tier-architecture

- Name: vnet-3tier
- Address Space: 10.0.0.0/16

3. Create three subnets:

Table 1 - three subnets

Subnet Name	Address Range	Purpose
subnet-web	10.0.1.0/24	Web Tier
subnet-app	10.0.2.0/24	App Tier
subnet-db	10.0.3.0/24	DB Tier

Figure 2 -VNet creation with subnets configured

Step 2: Create Network Security Groups

Create three NSGs with the following rules:

Web Tier NSG (nsg-web)

Inbound Rules:

Table 2 - Web Tier Inbound Rules

Priority	Name	Port	Source	Destination	Action
100	Allow-HTTP	80	Any	Any	Allow
110	Allow-HTTPS	443	Any	Any	Allow
120	Allow-SSH	22	Any	Any	Allow
130	Allow-RDP	3389	Any	Any	Allow

Outbound Rules:

Table 3 - Web Tier outbound Rule

Priority	Name	Port	Source	Destination	Action
100	Allow-to-App	Any	Any	10.0.2.0/24	Allow
110	Allow-Internet	Any	Any	Internet	Allow

App Tier NSG (nsg-app)

Inbound Rules:

Table 4 - App Tier Inbound Rule

Priority	Name	Port	Source	Destination	Action
100	Allow-HTTP-from-Web	80	10.0.1.0/24	Any	Allow
110	Allow-HTTPS-from-Web	443	10.0.1.0/24	Any	Allow
120	Allow-SSH-from-Web	22	10.0.1.0/24	Any	Allow
130	Allow-RDP-from-Web	3389	10.0.1.0/24	Any	Allow

Outbound Rules:

Table 5 - App Tier outbound Rule

Priority	Name	Port	Source	Destination	Action
100	Allow-to-DB	Any	Any	10.0.3.0/24	Allow
110	Allow-to-Web	Any	Any	10.0.1.0/24	Allow
4000	Deny-Internet	Any	Any	Internet	Deny

DB Tier NSG (nsg-db)

Inbound Rules:

Table 6 - DB Tier Inbound Rule

Priority	Name	Port	Source	Destination	Action
100	Allow-MySQL-from-App	3306	10.0.2.0/24	Any	Allow
110	Allow-SQL-from-App	1433	10.0.2.0/24	Any	Allow
120	Allow-SSH-from-App	22	10.0.2.0/24	Any	Allow
130	Allow-RDP-from-App	3389	10.0.2.0/24	Any	Allow

Outbound Rules:

Table 7 - DB Tier Outbound Rules

Priority	Name	Port	Source	Destination	Action
4000	Deny-All	Any	Any	Any	Deny

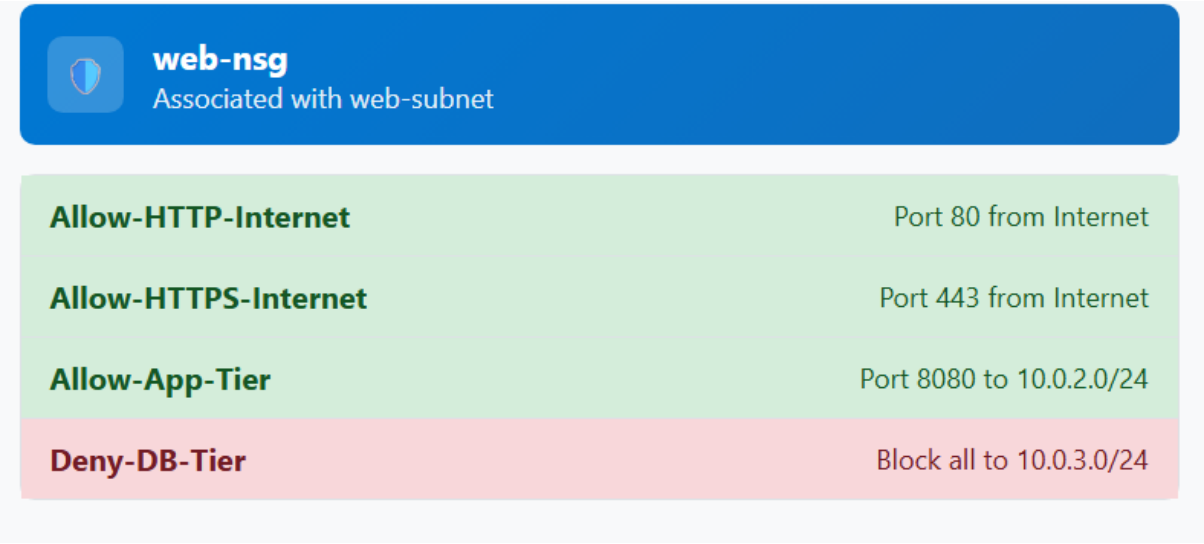

Figure 3 -Web Tier NSGe

app-nsg
Associated with app-subnet

Allow-Web-Tier	Port 8080 from 10.0.1.0/24
Allow-DB-Access	Port 3306,1433 to 10.0.3.0/24
Deny-Internet	Block Internet access

Figure 4 - App Tier NSG

db-nsg
Associated with db-subnet

Allow-App-MySQL	Port 3306 from 10.0.2.0/24
Allow-App-MSSQL	Port 1433 from 10.0.2.0/24
Deny-All-Outbound	Block all outbound traffic

Figure 5 - DB Tier NSG

Step 3: Associate NSGs to Subnets

Associate each NSG to its corresponding subnet:

- nsg-web → subnet-web
- nsg-app → subnet-app
- nsg-db → subnet-db

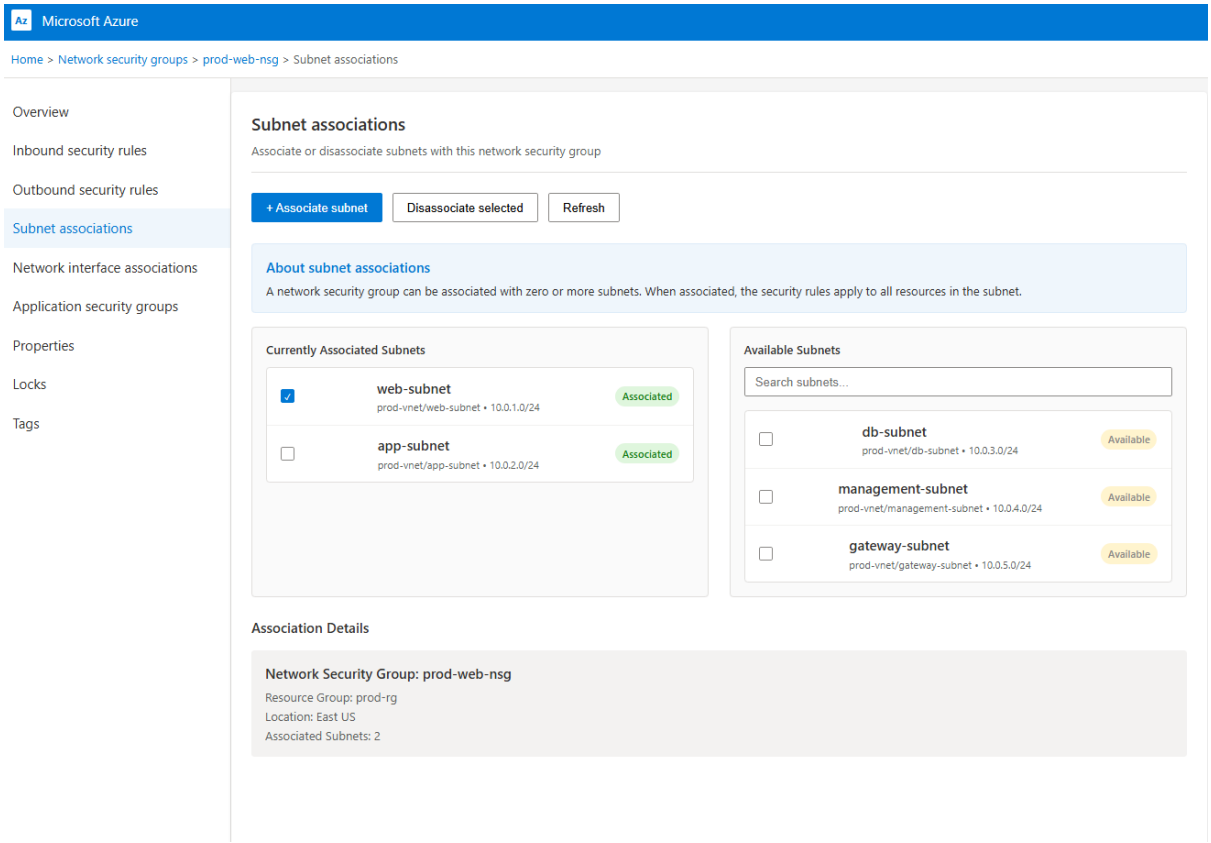

Figure 6 -NSG subnet association screen

Step 4: Deploy Virtual Machines

Deploy 6 VMs across the three tiers:

VM Configuration Table

VM Name	OS	Tier	Subnet	Public IP	Web Server
vm-web-linux	Ubuntu 20.04	Web	subnet-web	Yes	Apache
vm-web-windows	Windows Server 2022	Web	subnet-web	Yes	IIS
vm-app-linux	Ubuntu 20.04	App	subnet-app	No	Apache
vm-app-windows	Windows Server 2022	App	subnet-app	No	IIS
vm-db-linux	Ubuntu 20.04	DB	subnet-db	No	Apache
vm-db-windows	Windows Server 2022	DB	subnet-db	No	IIS

Table 8 - VM Configuration Table

Common VM Settings:

- Size: Standard_B2s

- Authentication: SSH key (Linux) / Password (Windows)
- Username: azureuser

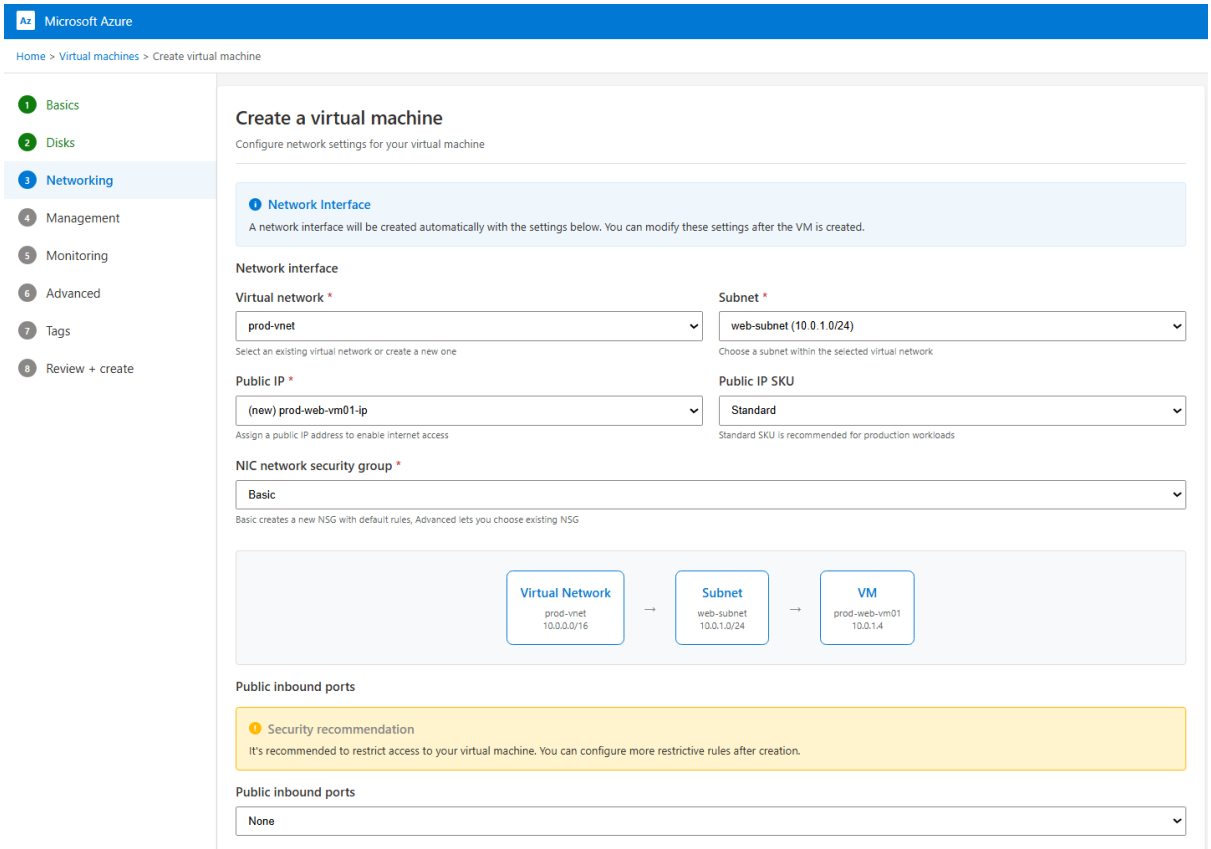

Figure 7 - VM deployment showing network configuration

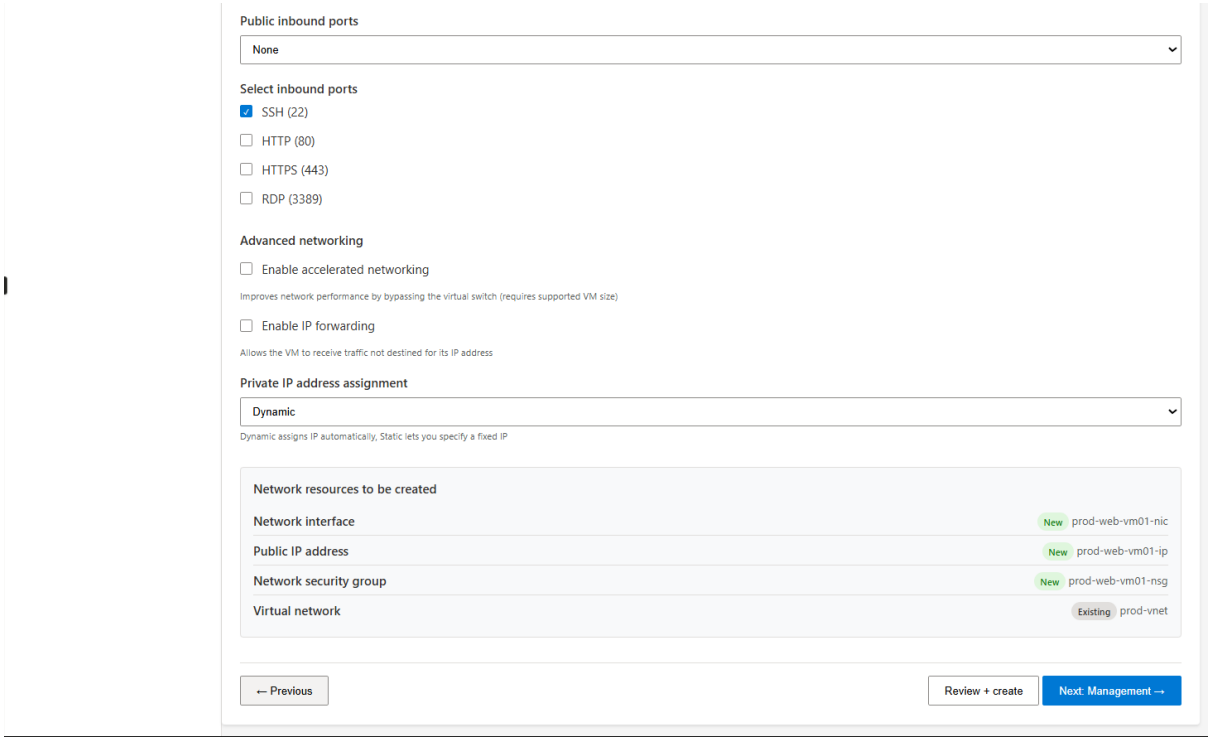

Figure 8 - VM deployment showing network configuration(continue)

Step 5: Install Web Servers

Apache on Linux VMs

Connect via SSH and execute:

```
# Update and install Apache

sudo apt update && sudo apt install apache2 -y


# Start and enable Apache

sudo systemctl start apache2

sudo systemctl enable apache2


# Create custom index page

echo "<h1>Welcome to $(hostname) - $(hostname -I)</h1>" | sudo tee
/var/www/html/index.html


# Configure firewall

sudo ufw allow 'Apache Full' && sudo ufw allow ssh && sudo ufw --force enable
```

IIS on Windows VMs

Connect via RDP and run PowerShell as Administrator:

```
# Install IIS

Install-WindowsFeature -name Web-Server -IncludeManagementTools


# Create custom index page

$content = "<h1>Welcome to $env:COMPUTERNAME - $(Get-NetIPAddress -AddressFamily
IPv4 | Where-Object {$_.IPAddress -like '10.0.*'} | Select-Object -ExpandProperty
IPAddress)</h1>"

Set-Content -Path "C:\inetpub\wwwroot\index.html" -Value $content


# Configure Windows Firewall
```

New-NetFirewallRule -DisplayName "Allow HTTP" -Direction Inbound -Protocol TCP -LocalPort 80 -Action Allow

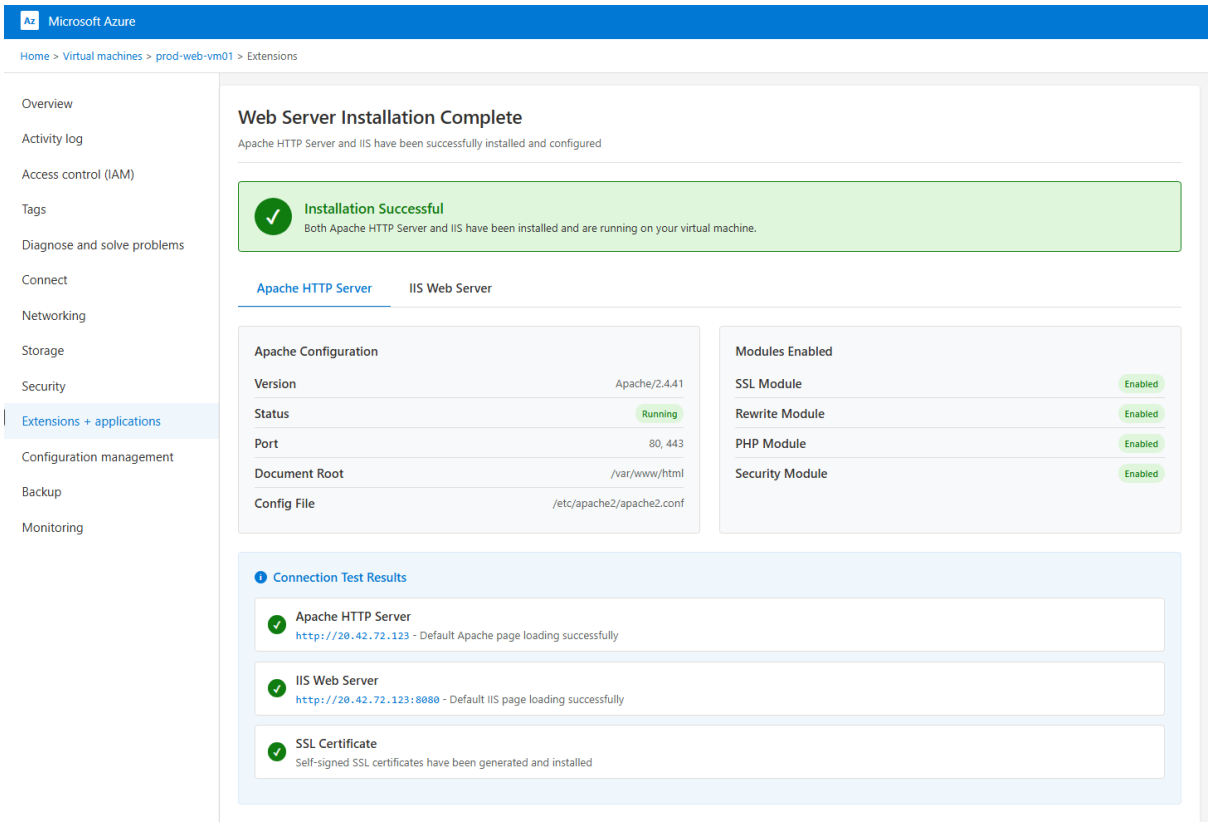

Figure 9 - Apache/IIS installation confirmation

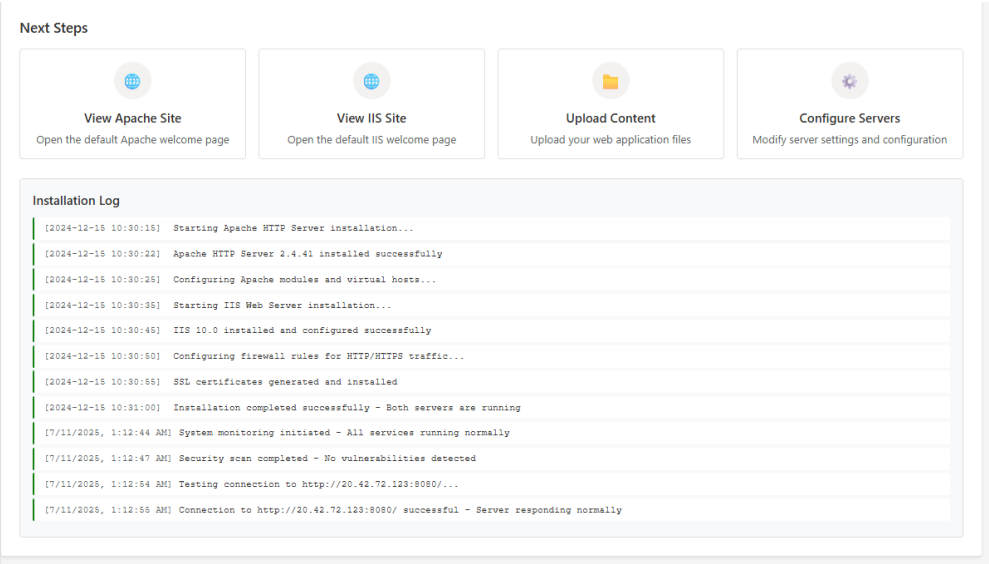

Figure 10 - Apache/IIS installation confirmation(continue)

Testing & Validation

Jump Box Access Strategy

Since App and DB tiers have no public IPs, use the **jump box method**:

1. **Direct Access**: Web tier VMs (have public IPs)
2. **Single Jump**: Web → App tier VMs
3. **Double Jump**: Web → App → DB tier VMs

Example SSH chain to access DB tier:

```
# Step 1: Connect to Web tier
ssh azureuser@<web-vm-public-ip>

# Step 2: From Web tier, connect to App tier
ssh azureuser@10.0.2.4

# Step 3: From App tier, connect to DB tier
ssh azureuser@10.0.3.4
```

Table 9 - Connectivity Validation Matrix

From Tier	To Web	To App	To DB	To Internet
Web	+	+	*	+
App	+	+	+	*
DB	*	*	+	*

In above table (+) means passed , (*) means failed

Test Commands

Internet Access Test (Web Tier Only)

```
# Should succeed from Web tier
```

```
curl -I https://www.google.com

# Should fail from App/DB tiers
curl -I https://www.google.com
```

Inter-Tier Communication Test

```
# From Web tier - test App tier access
curl http://10.0.2.4  # Should succeed
curl http://10.0.3.4  # Should fail

# From App tier - test DB tier access
curl http://10.0.3.4  # Should succeed
curl http://10.0.1.4  # Should succeed

# From DB tier - test other tiers
curl http://10.0.1.4  # Should fail
curl http://10.0.2.4  # Should fail
```

Web Server Access Test

```
# Test each VM's web server
curl http://<web-vm-public-ip>     # Direct access
curl http://10.0.2.4               # Via jump box
curl http://10.0.3.4               # Via double jump
```

```
Azure Connectivity Tests - PowerShell

PS C:\Users\admin> az login
Opening browser for authentication...
✓ Successfully logged in to Azure
[2025-07-11 14:23:15]

═══════════════════════════════════════════════════════════════════════

PS C:\Users\admin> az network vnet list --resource-group "prod-rg" --output table
Name          ResourceGroup  Location      AddressSpace
prod-vnet     prod-rg        eastus        10.0.0.0/16
staging-vnet  prod-rg        westus2       10.1.0.0/16

PS C:\Users\admin> Test-NetConnection -ComputerName "prod-vm.eastus.cloudapp.azure.com" -Port 443
Testing connection to prod-vm.eastus.cloudapp.azure.com:443...

ComputerName : prod-vm.eastus.cloudapp.azure.com
RemoteAddress : 20.85.123.45
RemotePort : 443
InterfaceAlias : Ethernet
SourceAddress : 192.168.1.100
TcpTestSucceeded : True
[2025-07-11 14:24:32]

PS C:\Users\admin> Test-NetConnection -ComputerName "staging-vm.westus2.cloudapp.azure.com" -Port 22
Testing connection to staging-vm.westus2.cloudapp.azure.com:22...

ComputerName : staging-vm.westus2.cloudapp.azure.com
RemoteAddress : 40.112.67.89
RemotePort : 22
InterfaceAlias : Ethernet
SourceAddress : 192.168.1.100
TcpTestSucceeded : False
WARNING: TCP connect to 40.112.67.89:22 failed
[2025-07-11 14:25:01]

═══════════════════════════════════════════════════════════════════════
```

Figure 11 - Failed test

```
PS C:\Users\admin> az network nic show-effective-route-table --resource-group "prod-rg" --name "prod-vm-nic"
Getting effective routes for network interface...

{
  "value": [
    {
      "name": "default-route",
      "source": "Default",
      "state": "Active",
      "addressPrefix": ["0.0.0.0/0"],
      "nextHopType": "Internet"
    }
  ]
}

✓ Route table retrieved successfully
[2025-07-11 14:25:45]

PS C:\Users\admin> az network nsg rule list --resource-group "prod-rg" --nsg-name "prod-nsg" --output table
Name          Priority      Direction     Access        Protocol      Port
AllowHTTPS    1000          Inbound       Allow         TCP           443
DenySSH       1001          Inbound       Deny          TCP           22

PS C:\Users\admin> ping azure.microsoft.com
Pinging azure.microsoft.com [20.70.246.20] with 32 bytes of data:
Reply from 20.70.246.20: bytes=32 time=23ms TTL=56
Reply from 20.70.246.20: bytes=32 time=21ms TTL=56
Reply from 20.70.246.20: bytes=32 time=22ms TTL=56
Reply from 20.70.246.20: bytes=32 time=24ms TTL=56

Ping statistics for 20.70.246.20:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 21ms, Maximum = 24ms, Average = 22ms
[2025-07-11 14:26:15]

PS C:\Users\admin> az network watcher test-connectivity --source-resource "prod-vm" --dest-address "10.1.0.4" --dest-port 3389
Running Azure Network Watcher connectivity test...

ConnectionStatus: Failed
AvgLatencyInMs: 0
MaxLatencyInMs: 0
MinLatencyInMs: 0
ProbesSent: 3
ProbesFailed: 3

Issue: NetworkSecurityRule
Details: Traffic blocked by Network Security Group rule 'DenyRDP'
[2025-07-11 14:27:03]
```

Figure 12 - Successful Test

```
PS C:\Users\admin> az network watcher test-connectivity --source-resource "prod-vm" --dest-address "10.1.0.4" --dest-port 3389
Running Azure Network Watcher connectivity test...

ConnectionStatus: Failed
AvgLatencyInMs: 0
MaxLatencyInMs: 0
MinLatencyInMs: 0
ProbesSent: 3
ProbesFailed: 3

Issue: NetworkSecurityRule
Details: Traffic blocked by Network Security Group rule 'DenyRDP'
[2025-07-11 14:27:03]

═══════════════════════════════════════════════════════════════════════════

PS C:\Users\admin> echo "Connectivity Test Summary:"
Connectivity Test Summary:

✓ HTTPS to prod-vm (443) - SUCCESS
✗ SSH to staging-vm (22) - FAILED (NSG Rule)
✓ Internet connectivity - SUCCESS
✗ RDP to internal VM (3389) - FAILED (NSG Rule)

⚠ Review Network Security Group rules for failed connections
[2025-07-11 14:27:30]

PS C:\Users\admin> _
```

Figure 13 - Failed Test

Challenges & Solutions

Common Issues

Issue	Cause	Solution
Cannot access VMs	NSG rules too restrictive	Verify SSH/RDP rules allow access from correct sources
Internet access blocked	Default deny rules	Ensure outbound internet rules have correct priority
Jump box access fails	Missing inter-tier communication rules	Verify App tier allows inbound from Web tier
Web servers not accessible	Firewall misconfiguration	Check both Azure NSG and OS-level firewall rules

Table 10 - Common Issues

Troubleshooting Commands

Linux:

Check service status

sudo systemctl status apache2

Test network connectivity

telnet <ip> <port>

Check firewall

sudo ufw status

Windows:

Check IIS status

Get-Service W3SVC

Test connectivity

Test-NetConnection -ComputerName <ip> -Port <port>

Check firewall

Get-NetFirewallRule | Where-Object {$_.DisplayName -like "*HTTP*"}

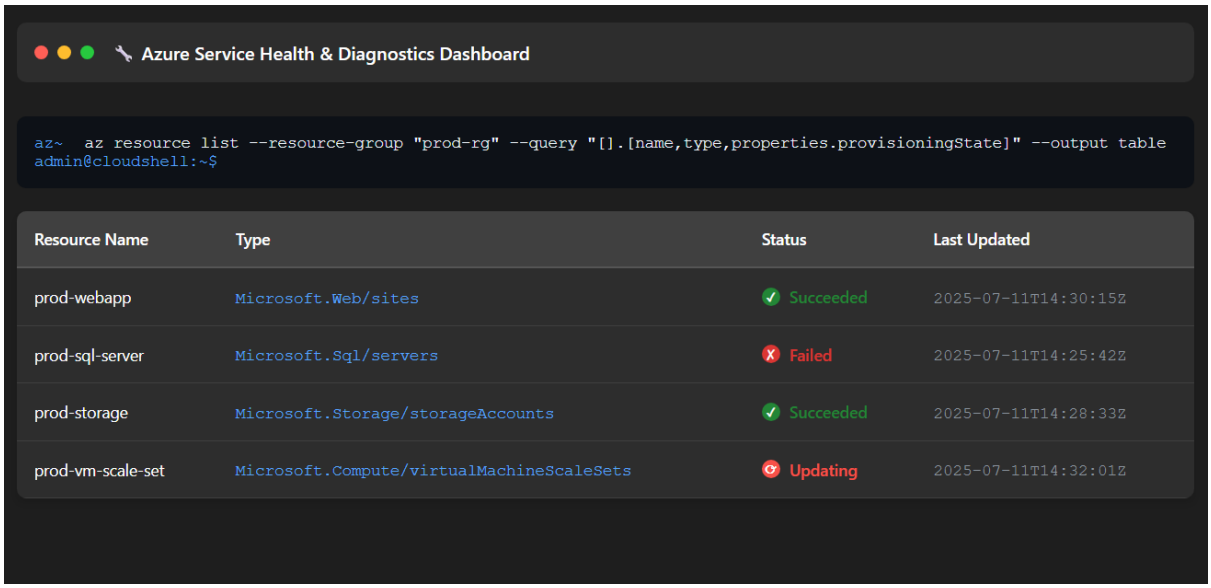

Figure 14 - Troubleshooting output showing service status

```
az-admin@cloudshell:~$ az monitor metrics list --resource "/subscriptions/xxx/resourceGroups/prod-rg/providers/Microsoft.Web/sites/prod-webapp" --metric "Http5xx,ResponseTime" --interval PT1M

🔍 Application Insights - prod-webapp

HTTP 5xx Errors (Last 5 min):                    23 errors
Average Response Time:                           2.8 seconds
Requests/min:                                    1,247
CPU Usage:                                       87%
Memory Usage:                                    94%

⚠ PERFORMANCE ALERT
High memory usage detected. Consider scaling up the App Service plan.
```

Figure 15 - monitor metrics list

```
az-admin@cloudshell:~$ az sql db show-connection-strings --server "prod-sql-server" --name "prod-database" --client ado.net

🔍 SQL Database Health Check

ERROR: Server 'prod-sql-server.database.windows.net' is not responding
Connection timeout occurred after 30 seconds

🔄 Running diagnostic tests...

Server Status:                                   OFFLINE
Last Successful Connection:                      2025-07-11T13:45:22Z
Database State:                                  UNKNOWN
Failover Status:                                 IN PROGRESS

🚨 CRITICAL ALERT
SQL Server failover in progress. Expected restoration time: 5-10 minutes.
Failover Progress: 65% complete
```

Figure 16 - SQL Database Health Check

```
az-admin@cloudshell:~$ az vm list --resource-group "prod-rg" --show-details --query "[].{Name:name,PowerState:powerState,Size:hardwareProfile.vmSize}" --output table
```

🖥️ **Virtual Machine Status**

VM Name	Power State	Size	Health
prod-web-vm-01	VM running	Standard_D2s_v3	✓ Healthy
prod-web-vm-02	VM running	Standard_D2s_v3	⚠ High CPU
prod-db-vm	VM stopped	Standard_D4s_v3	✗ Stopped

Figure 17 - Virtual Machine Status

```
az-admin@cloudshell:~$ az monitor activity-log list --resource-group "prod-rg" --start-time 2025-07-11T13:00:00Z --query "[?level=='Error']" --output table
```

📋 **Recent Error Events**

2025-07-11T13:47:33Z
ERROR: SQL Server connection failed
Resource: prod-sql-server
Operation: Database.Connect

2025-07-11T14:15:18Z
ERROR: VM prod-db-vm shutdown unexpectedly
Resource: prod-db-vm
Operation: VM.PowerOff

2025-07-11T14:22:45Z
WARNING: Memory threshold exceeded
Resource: prod-webapp
Operation: AppService.MemoryAlert

Figure 18 - Recent Error Events

```
az-admin@cloudshell:~$ az network watcher show-topology --resource-group "prod-rg" --target-resource-group "prod-rg"
```

🌐 **Network Topology Health**

Load Balancer Status:	✓ Healthy
Backend Pool Health:	⚠ 1 of 2 endpoints healthy
Network Security Groups:	✓ All rules active
VNet Connectivity:	✓ All subnets reachable

Figure 19 - Network Topology Health

```
az-admin@cloudshell:~$ az monitor diagnostic-settings list --resource "/subscriptions/xxx/resourceGroups/prod-rg" --query "[].{Name:name,Status:properties.logs[0].enabled}"

📊 Diagnostic Summary
```

Service	Status	Issues	Action Required
Web App	⚠ Degraded	High memory usage	Scale up plan
SQL Database	✗ Critical	Server offline	Monitor failover
Storage Account	✓ Healthy	None	None
Virtual Machines	⚠ Partial	1 VM stopped	Restart DB VM

Figure 20 - Diagnostic Summary

```
az-admin@cloudshell:~$ echo "=== TROUBLESHOOTING COMPLETE ==="

🔍 TROUBLESHOOTING SUMMARY

🚨 CRITICAL ISSUES FOUND:

  • SQL Server is offline - Failover in progress (65% complete)
  • Database VM unexpectedly stopped

⚠ WARNING ISSUES:

  • Web app memory usage at 94%
  • Load balancer backend pool partially healthy

✓ SERVICES OPERATING NORMALLY:

  • Storage account fully operational
  • Network connectivity stable
  • Security groups functioning correctly
[Diagnostic completed at 2025-07-11T14:35:42Z]

az-admin@cloudshell:~$ _
```

Figure 21 - TROUBLESHOOTING SUMMARY

Conclusion

Achievements

- Implemented 3-tier network architecture with proper segmentation
- Configured granular NSG rules enforcing security boundaries
- Deployed 6 VMs across multiple tiers with mixed OS platforms
- Successfully configured Apache and IIS web servers
- Validated network isolation and jump box access methodology

Key Learnings

- NSG rule priorities are critical for proper traffic control
- Jump box methodology is essential for managing isolated infrastructure
- Subnet-level NSG association provides better security than VM-level

- Comprehensive testing validates security implementation effectiveness
- Proper documentation ensures reproducible deployments

This implementation demonstrates enterprise-grade network security practices while maintaining operational accessibility through controlled jump box access patterns.