07

CNW-2511 CLOUD NETWORKING

Cloud Security Operations Report: Azure Virtual Network Deployment

Provisioning and validating a segmented Azure VNet with NSGs, Azure Firewall DNAT, a Standard load balancer, and RDP connectivity proofs

COURSE
CNW-2511 Cloud Networking

DATE
April 2025

ENVIRONMENT
Azure for Students subscription (ID 4ddeee3e-9824-4cd4-ba1e-a8304c5708e7), Central US region

PAGES (REPORT)
21

STUDENT
Cody Richard · 0005288412

ORGANIZATION
Full Sail University

TECHNICAL BRIEF · EXECUTIVE SUMMARY

Cloud Security Operations Report: Azure Virtual Network Deployment

For Lab 2.4 I provisioned and documented a segmented cloud network in Azure for Students (Central US, subscription 4ddeee3e-9824-4cd4-ba1e-a8304c5708e7) under resource group richardc-042025. The build centered on a single Virtual Network carved into an AzureFirewallSubnet and a cnw-central-compute-richardc workload subnet (10.0.1.0/24), fronted by a Standard-tier Azure Firewall (cnw-central-fw-richardc) bound to a static public IP (172.170.118.101). I attached an NSG with an RDP allow rule to the compute tier and stood up two Windows VMs — a Windows 10 Pro client (cnw-client01, 10.0.1.80) and a Windows Server 2016 Datacenter web server (cnw-websrv, 10.0.1.70) — then exposed RDP through destination-NAT rules (33890→10.0.1.70, 33891→10.0.1.80) on both the firewall and a Standard load balancer. I verified end-to-end reachability by connecting over FreeRDP to the public IP on the mapped ports and capturing ipconfig /all on each host, then tore the environment down and confirmed cleanup in the resource group view.

▸ Objectives

- Provision a segmented Azure Virtual Network with a dedicated AzureFirewallSubnet and a workload compute subnet
- Deploy and attach an Azure Firewall to a static public IP as the network's secure ingress point
- Apply a Network Security Group with a least-privilege RDP allow rule to the compute tier
- Configure destination-NAT rules that map external ports to private VM IPs for controlled RDP exposure
- Validate RDP connectivity to both the client and web-server VMs and capture their IP configuration as evidence
- Decommission all resources and verify clean teardown of the resource group

▸ Environment

```
Azure for Students subscription (ID 4ddeee3e-9824-4cd4-ba1e-a8304c5708e7), Central US region
```

```
Resource group richardc-042025 (Central US)
```

```
Azure Virtual Network with AzureFirewallSubnet and cnw-central-compute-richardc subnet on the 10.0.1.0/24 space
```

```
Azure Firewall cnw-central-fw-richardc (Standard tier, private IP 10.0.1.4) bound to public IP cnw-east-PIP-richardc / 172.170.118.101
```

```
Standard-SKU load balancer LNFI-West-LB with inbound NAT rules
```

WALKTHROUGH & EVIDENCE

For Lab 2.4 I provisioned a segmented Azure Virtual Network in the Central US region and hardened it with an Azure Firewall, a Network Security Group, and destination-NAT-based RDP exposure backed by a Standard load balancer. This walkthrough traces the build end to end: core infrastructure, subnet topology, security and firewall configuration, the two Windows VMs, RDP connectivity proofs, and verified teardown. Every figure is a captured Azure portal or FreeRDP screenshot from the live deployment under resource group richardc-042025.

SETUP

Resource Group & Public IP

Established the deployment container and the environment's static ingress address.

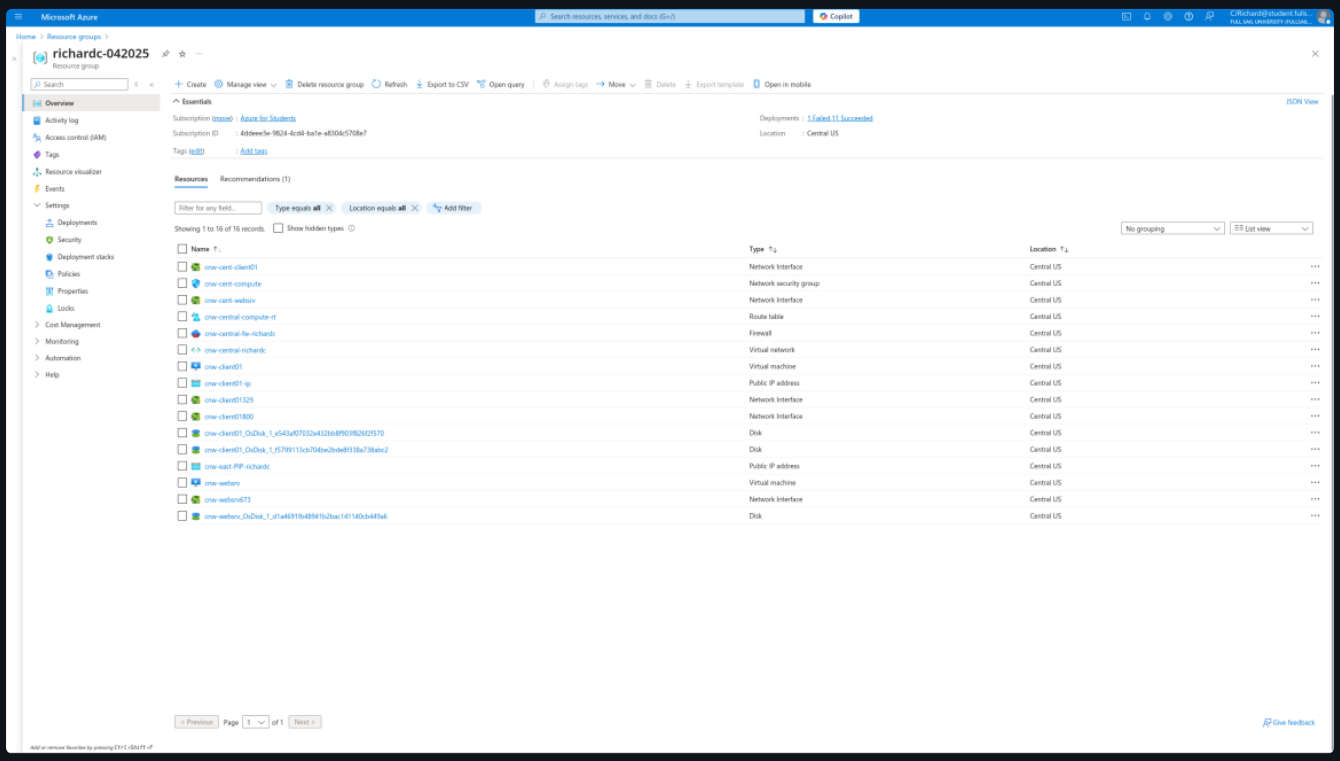

FIG 01 — Resource group richardc-042025 in Central US, the container holding every resource provisioned for this lab.

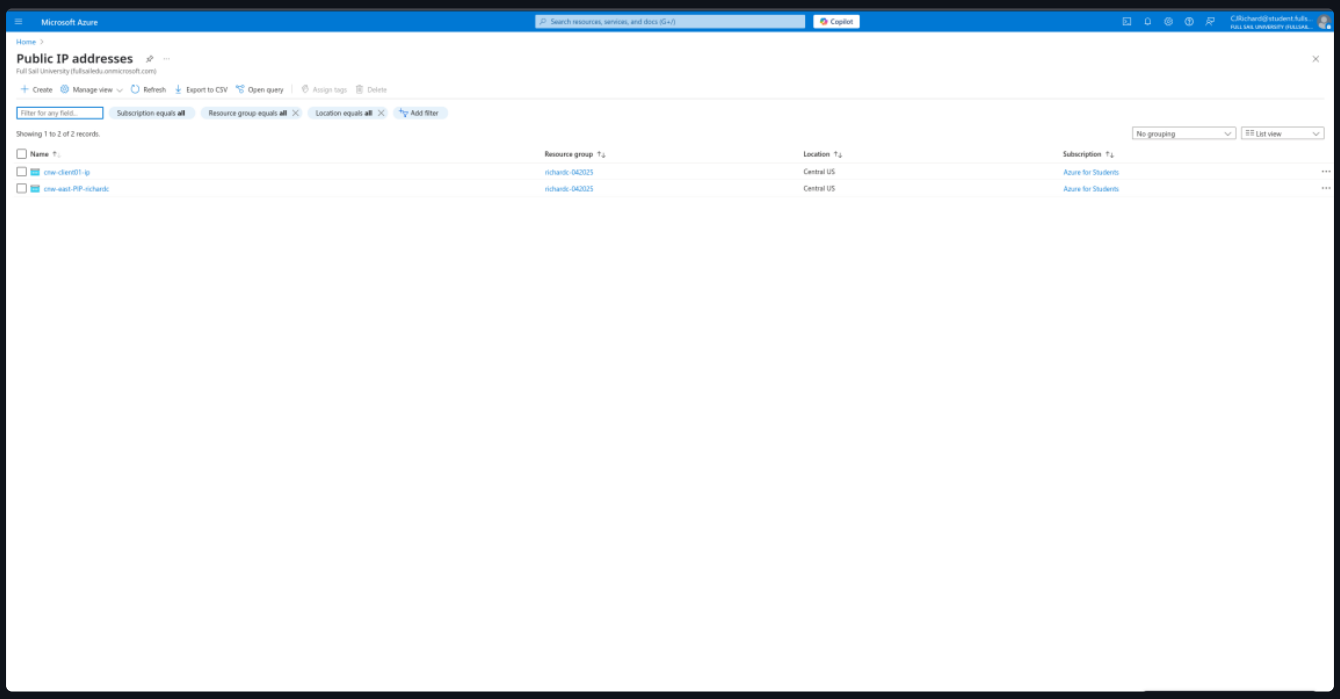

FIG 02 — Static public IP cnw-east-PIP-richardc allocated as the environment's ingress address at 172.170.118.101.

NETWORKING

Virtual Network

Provisioned the VNet that anchors the entire address space.

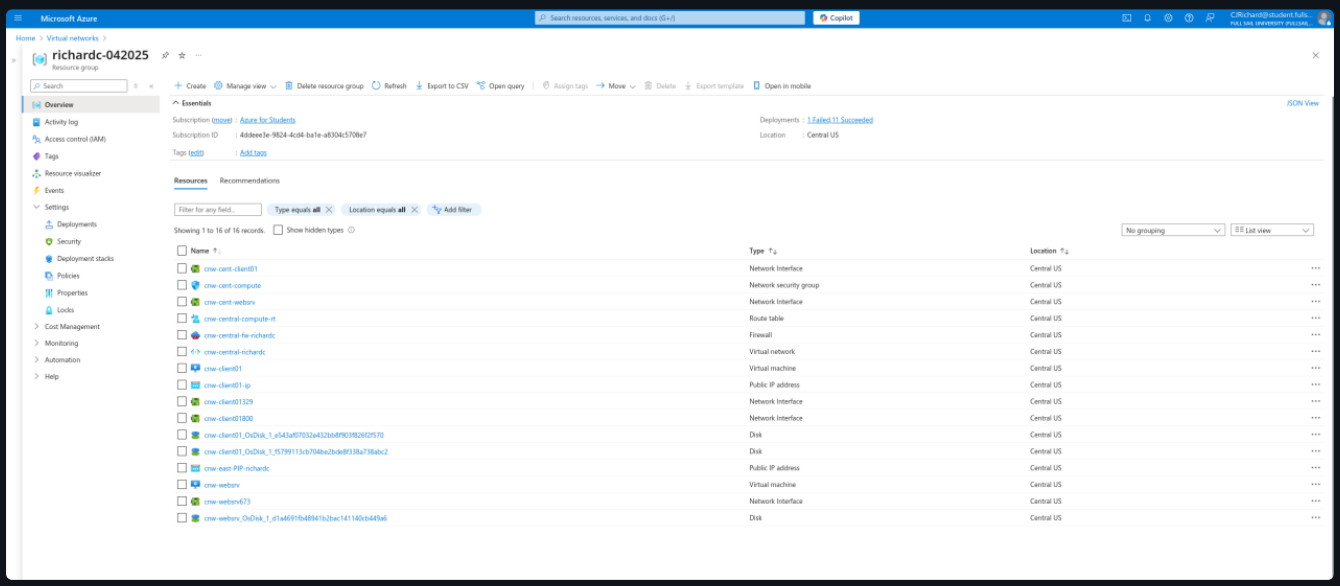

FIG 03 — Virtual Network overview for the deployment, defining the address space carved up across the firewall and compute tiers.

NETWORKING

Subnet Segmentation

Split the VNet into a dedicated firewall subnet and an isolated workload tier.

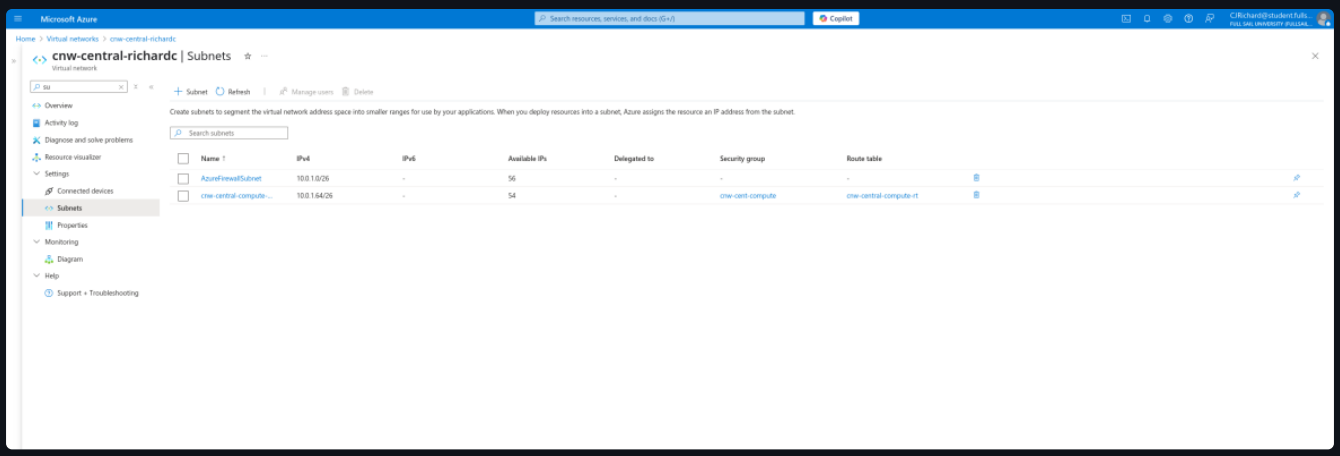

FIG 04 — Subnets view showing the dedicated AzureFirewallSubnet alongside the cnw-central-compute-richardc workload subnet on 10.0.1.0/24.

TOPOLOGY

Network Topology

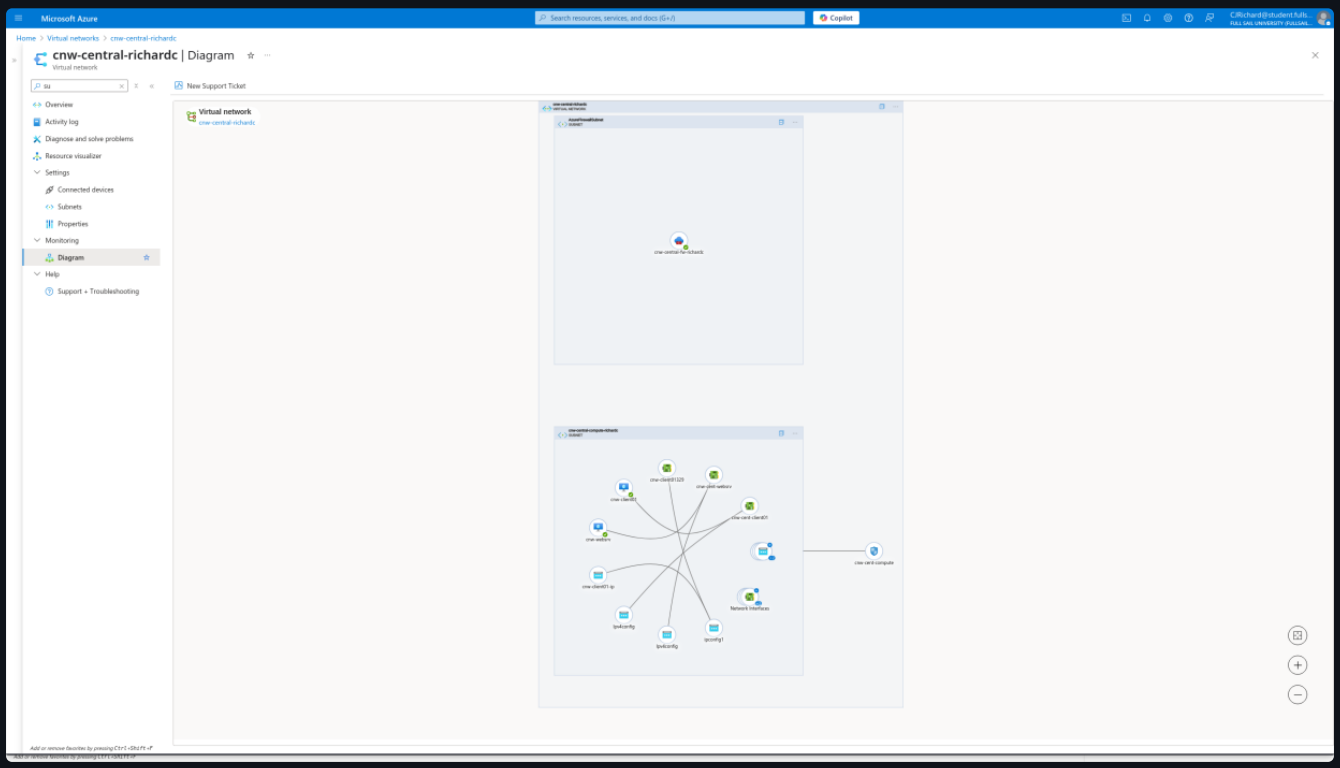

FIG 05 — Azure diagram view of the deployed topology, mapping the firewall, subnets, NICs, and VMs into a single network picture.

INTERFACES

Network Interfaces

Confirmed both workload VMs are wired into the compute subnet.

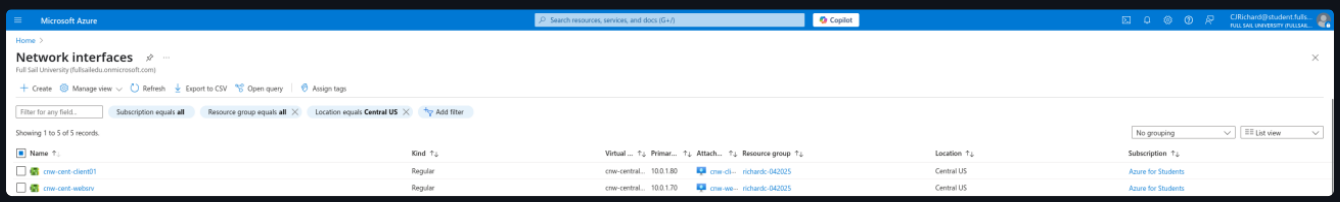

FIG 06 — Two NICs attached to the compute tier, one bound to cnw-client01 and one to cnw-websrv.

SECURITY

Network Security Group

Scoped inbound access on the compute tier to RDP only.

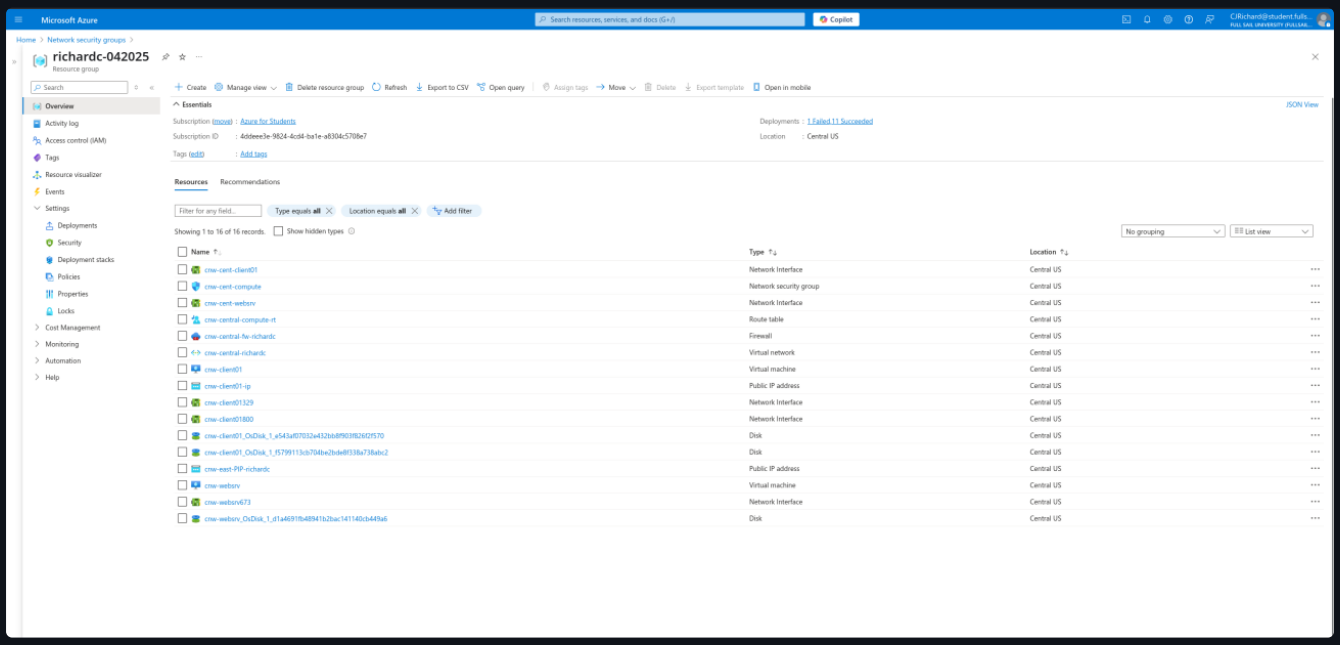

FIG 07 — NSG cnw-east-compute carrying the least-privilege RDP allow rule that fronts the compute subnet.

FIREWALL

Azure Firewall

Deployed the Standard-tier firewall as the network's secure ingress point.

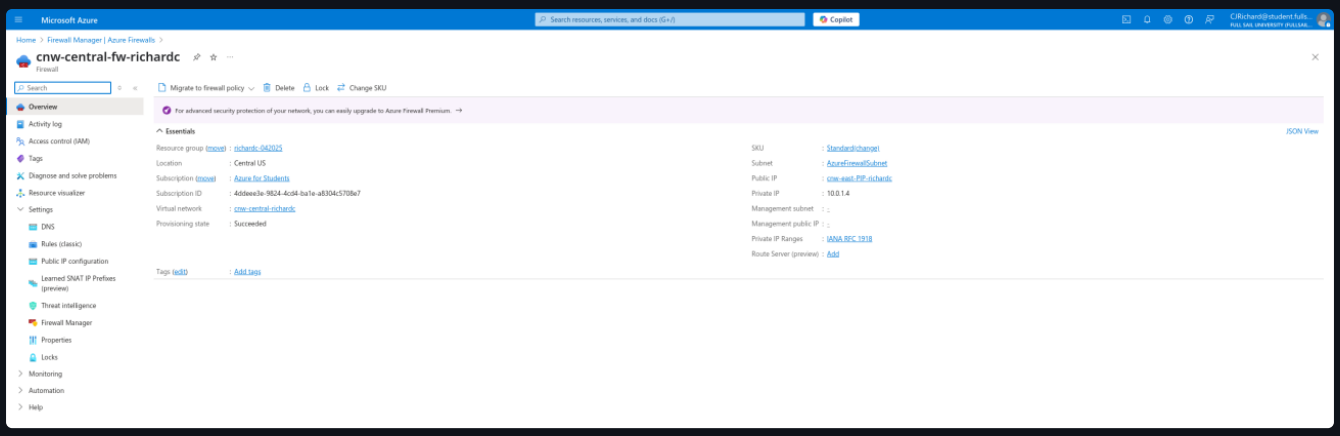

FIG 08 — Azure Firewall cnw-central-fw-richardc (Standard tier, private IP 10.0.1.4) bound to the public IP 172.170.118.101.

FIREWALL

Firewall DNAT Rules

Mapped external ports to private VM IPs for controlled RDP exposure.

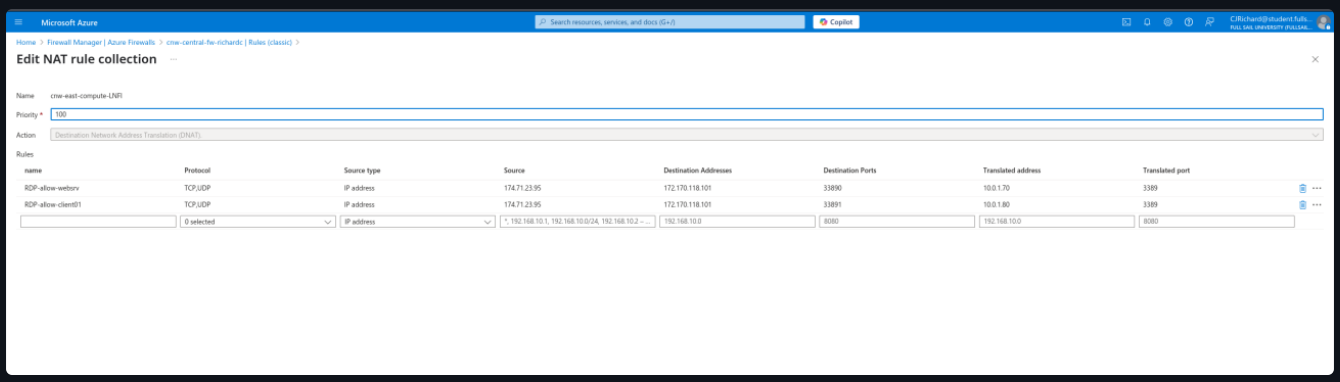

FIG 09 — Firewall destination-NAT rules translating port 33890 to 10.0.1.70 (web server) and 33891 to 10.0.1.80 (client).

COMPUTE

Client VM

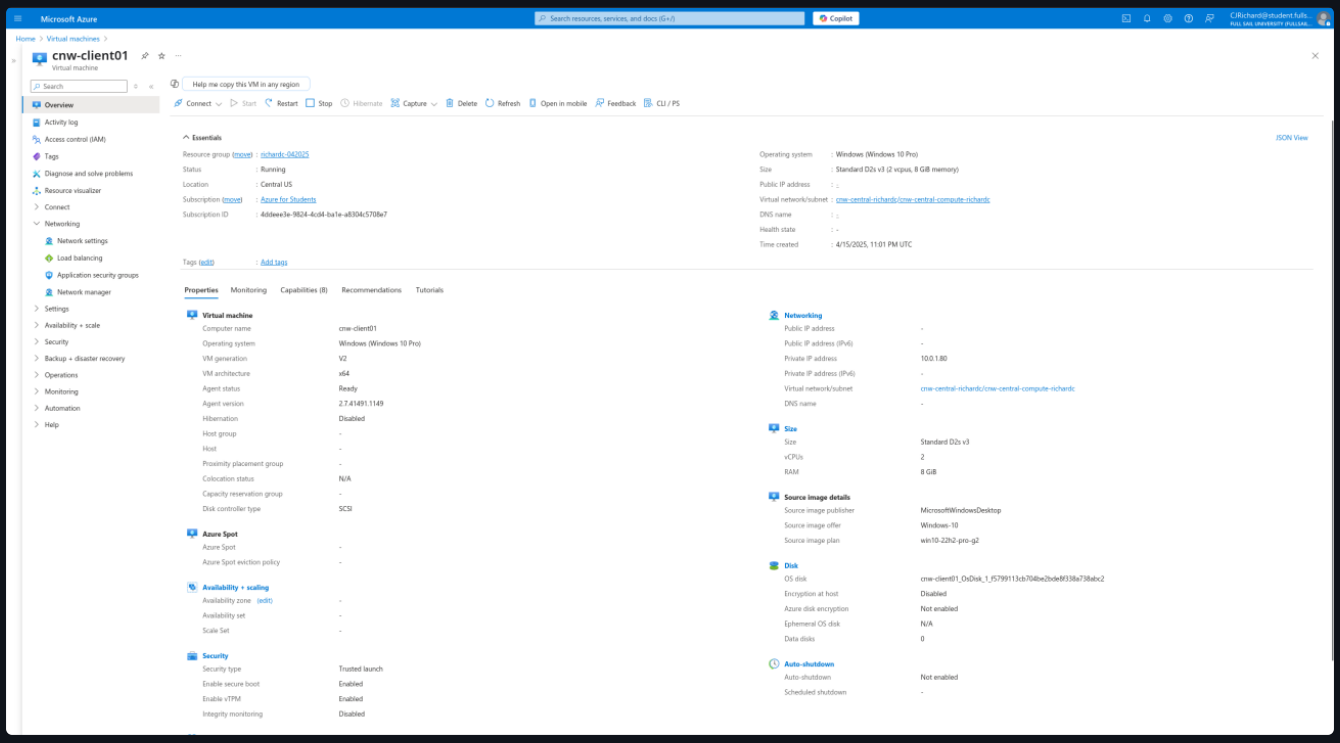

FIG 10 — Client VM cnw-client01 running Windows 10 Pro on private IP 10.0.1.80.

COMPUTE

Web Server VM

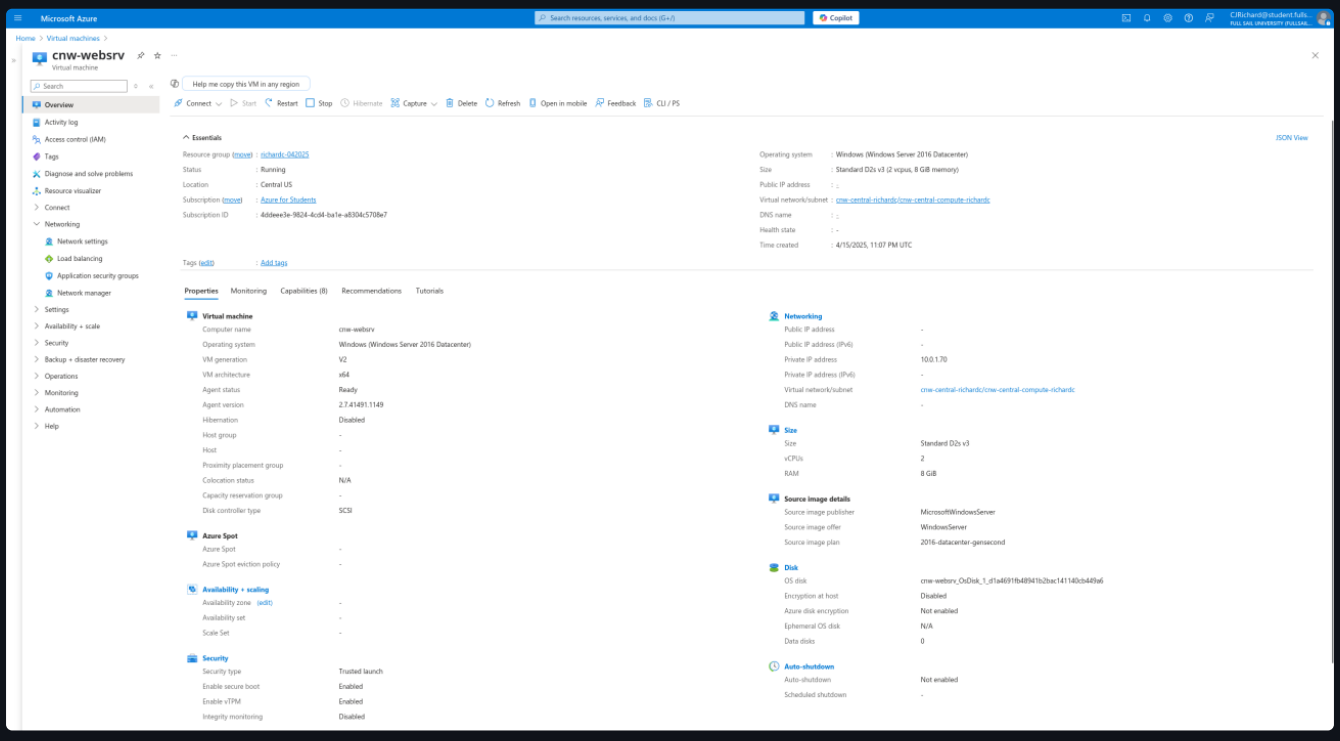

FIG 11 — Web-server VM cnw-websrv running Windows Server 2016 Datacenter on private IP 10.0.1.70.

VALIDATION

RDP Proof — Client

Confirmed reachability to the client over FreeRDP and captured its IP configuration.

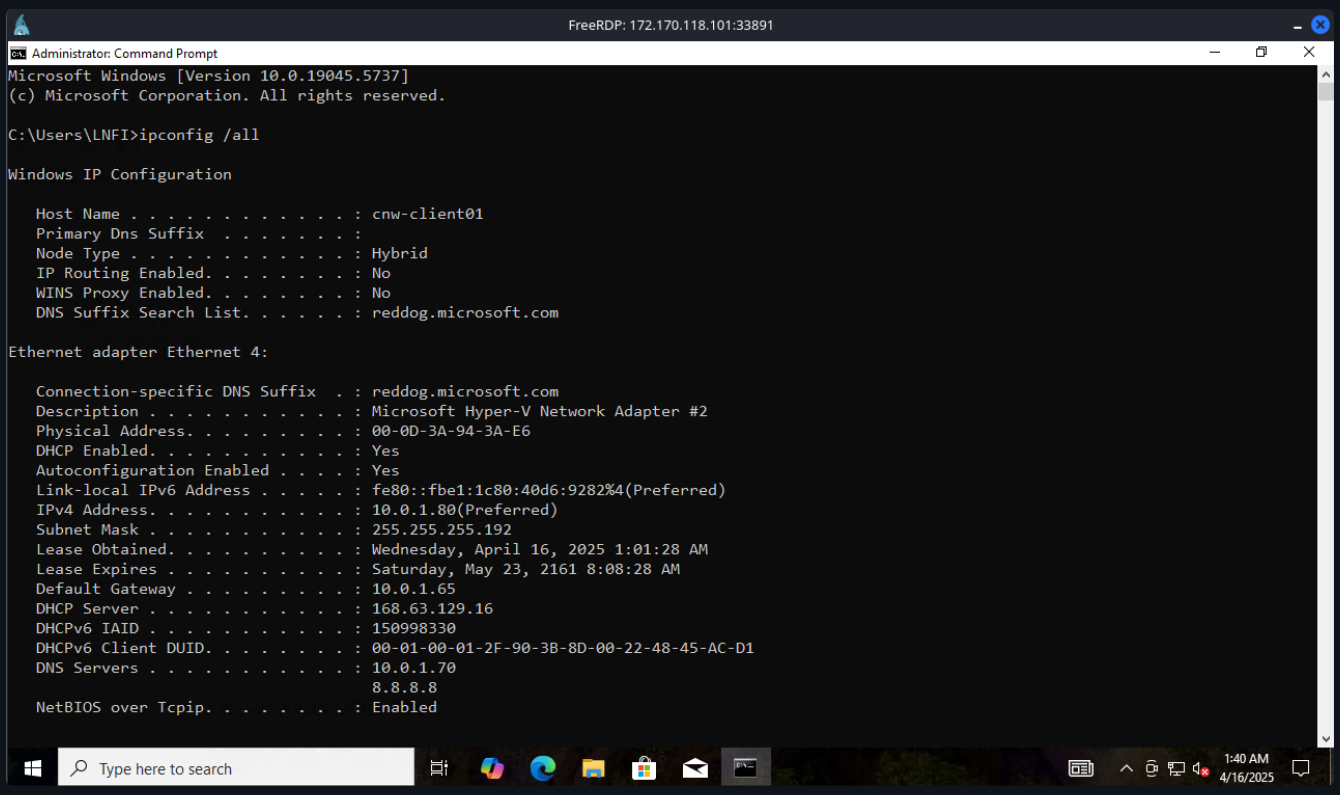

FIG 12 — ipconfig /all on cnw-client01 after RDP over NAT, confirming IPv4 10.0.1.80, gateway 10.0.1.65, and DNS 10.0.1.70 / 8.8.8.8.

VALIDATION

RDP Proof — Web Server

Confirmed reachability to the web server over the mapped NAT port.

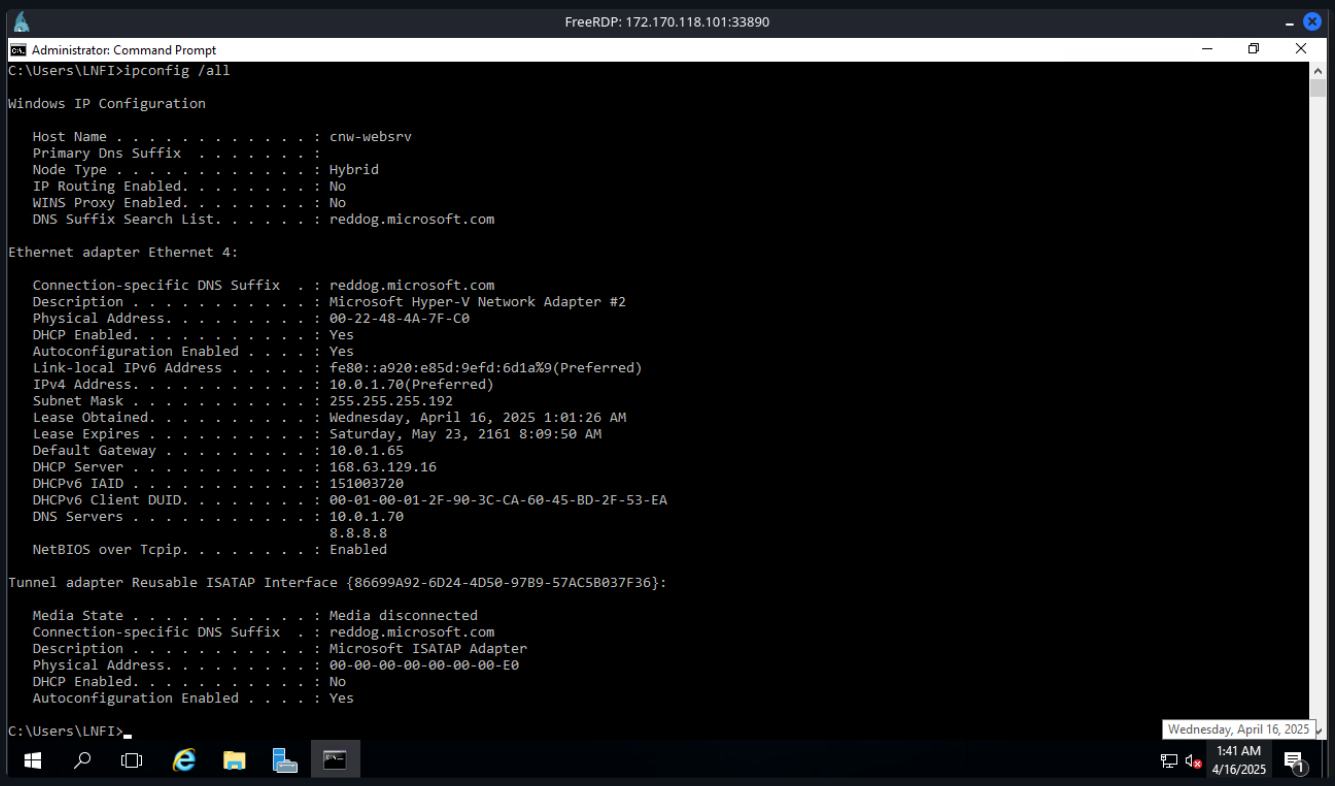

FIG 13 — ipconfig /all on cnw-websrv after RDP over NAT, verifying the web-server host's IPv4 lease on 10.0.1.70 and end-to-end connectivity.

TEARDOWN

Resource Cleanup Verification

Decommissioned the environment and confirmed a clean resource group.

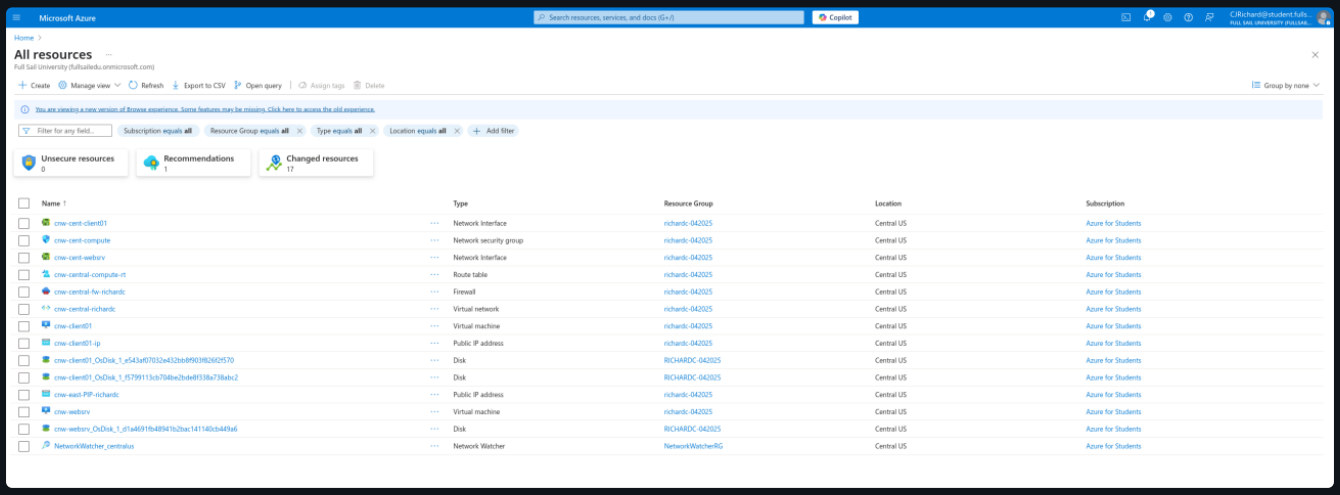

FIG 14 — All-resources view of richardc-042025 after teardown, confirming the deployment was fully decommissioned.