Evaluating the Effectiveness of Azure Network Security Groups in Containing Lateral Threat Movement

Filmon Mehari
Master in Cybersecurity / University West
filmon-mehari.gebrezghi@student.hv.se

***Abstract*—This study evaluates the effectiveness of Azure Network Security Groups (NSGs) in preventing lateral movement of simulated cyber threats within cloud networks. Through a controlled experimental setup in Microsoft Azure, we deploy two virtual machines in a flat network configuration to simulate worm like behavior. By applying and testing restrictive NSG rules, we measure the containment of threat propagation while maintaining operational transparency. The results demonstrate that subnet level NSG rules effectively block internal threat movement while preserving administrative access, providing empirical evidence supporting micro-segmentation implementation in cloud environments.**

I. Introduction

Lateral movement represents a critical security challenge in cloud environments, where attackers propagate internally after an initial breach. This study investigates the efficacy of Azure Network Security Groups (NSGs) as a micro-segmentation control mechanism to contain such threats. We employ a hands-on simulation approach to create and contain a simulated network worm within a controlled Azure lab environment. The primary objective is to provide practical evidence of how NSGs can implement network segmentation theory to mitigate real world threats identified in recent cloud security reports [1].

A. Research Questions

This study addresses the following research questions:

RQ1: To what extent can properly configured Azure Network Security Groups contain simulated lateral threat movement between virtual machines in a cloud subnet?

RQ2: What operational impact does such containment configuration have on normal administrative functions and connectivity?

The scope is limited to testing fundamental network segmentation controls using safe, simulated threat behaviors, providing practical evidence without ethical risks associated with real malware deployment.

II. Background and Related Work

Recent findings highlight how common lateral movement threats are in cloud environments.. The 2023 Unit 42 Cloud Threat Report indicates that 76% of organizations have overly permissive cloud security groups, facilitating internal threat propagation [2]. Research by Al-Ofeishat and Alshorman demonstrates that network segmentation is essential for containing breaches and stopping lateral movement [1]. Microsoft defines NSGs as built-in firewalls for controlling traffic between Azure resources [3]. Benzaïd et al. propose zero-trust microsegmentation frameworks for next generation networks, while Arifeen et al. demonstrate automated segmentation for IoT environments [4], [5]. This project bridges theoretical frameworks with practical implementation by testing how effectively Azure NSGs can apply micro-segmentation principles to mitigate the threats documented in contemporary research.

III. Methodology

This study employs an experimental methodology conducted entirely within Microsoft Azure. The experiment evaluates lateral movement between cloud virtual machines and the containment effectiveness of Azure NSGs.

A. Research Design Classification

This study follows an **Experimental Research Design**, characterized by intentional manipulation of an independent variable (NSG configuration) under controlled conditions to observe causal effects on dependent variables (connectivity metrics). This design enables direct measurement of security control effectiveness through controlled variable manipulation and systematic observation.

B. Experimental Design

The experimental design is summarized in Table I, which outlines the core components and variables of the study.

TABLE I
Experimental Setup and Variables

Component	Description
Cloud Platform	Microsoft Azure
Virtual Network	Single VNet with one subnet
Virtual Machines	2 × Ubuntu Server 22.04 LTS
Security Control	Azure Network Security Group (subnet-level)
Independent Variable	NSG Configuration (Default vs. Restrictive)
Dependent Variables	Ping response, Telnet connectivity, port visibility
Monitoring	Azure Network Watcher Flow Logs
Data Collection	CLI outputs, JSON flow logs, screenshots

C. Deployment Configuration

The Azure environment was deployed in Sweden Central region within resource group `Project-RG`. A virtual network `Project-VNet` (10.1.0.0/16) contained subnet `Project-Subnet` (10.1.1.0/24). Two Ubuntu Server VMs were deployed: `JumpBox` with public IP for administrative access, and `Target1` with private IP only (10.1.1.5) to simulate an internal server. Complete deployment commands and configurations are documented in the project repository [6].

D. Procedure

The experimental procedure follows this sequential workflow:

TABLE II
EXPERIMENTAL PROCEDURE WORKFLOW

Step 1	Deploy VNet, subnet, and two VMs (Create flat network)
Step 2	Run baseline tests (Confirm unrestricted lateral movement)
Step 3	Enable Telnet service on Target1 (Simulate vulnerable service)
Step 4	Apply restrictive NSG rule (Implement micro-segmentation)
Step 5	Repeat connectivity tests (Evaluate containment effectiveness)
Step 6	Collect flow logs (Verify traffic blocking)
Step 7	Compare before/after results (Assess security impact)

E. Tools and Testing

Testing employed standard network diagnostics: `ping` for ICMP connectivity, `telnet` for TCP service access, `nmap` for port scanning, and Metasploit auxiliary modules for safe service enumeration. Baseline tests established unrestricted communication; post-NSG tests verified containment.

F. Simulation Implementation

To ensure consistent and reproducible testing sequences, a custom automation script was developed. The script coordinated: (1) host discovery within the subnet, (2) service probing using safe scanning techniques, (3) controlled propagation simulation attempts, and (4) comprehensive logging of all activities. This automation maintained identical testing conditions across baseline and containment phases, supporting experimental reliability.

G. Reliability and Repeatability

The experimental setup ensures reproducibility through: (1) Azure Resource Manager templates for consistent deployment, (2) documented command sequences in the GitHub repository [6], (3) identical VM configurations (size: Standard_B1s, OS: Ubuntu 22.04), and (4) timestamped flow logs for verification. The procedure can be replicated by deploying the provided templates and executing the documented test sequence.

H. Hypotheses

Based on the experimental design, we formulate two hypotheses:

H1 (Effective Containment): A restrictive Azure NSG configuration will block all simulated lateral movement attempts (ping, Telnet) between virtual machines in the same subnet.

H2 (Operational Transparency): Proper NSG configuration will maintain administrative access (SSH) while blocking malicious traffic, ensuring security does not disrupt normal operations.

IV. RESULTS

The experimental results provide clear, quantitative evidence of Azure NSG effectiveness in containing lateral movement while maintaining operational functionality.

A. Baseline Connectivity (Pre-NSG)

Before applying NSG rules, baseline tests confirmed unrestricted lateral movement:

- **Ping (ICMP):** 100% success rate (4/4 packets received, 0% loss) with average latency of 0.8ms between VMs
- **Telnet (Port 23):** Successful TCP handshake and service banner retrieval within 2ms
- **Service Scanning:** Nmap identified 3 open ports (22/SSH, 23/Telnet, 80/HTTP) on Target1

B. Post-NSG Containment Results

After applying a restrictive NSG rule (`Deny` all internal traffic), all lateral movement was blocked:

TABLE III
CONNECTIVITY TEST RESULTS COMPARISON

Test Type	**Before NSG**	**After NSG**	**Containment**
Ping (ICMP Echo)	100% success	100% blocked	100%
Telnet (TCP/23)	Connected	Timeout	100%
Port Scanning	3 ports visible	0 ports detected	100%
Service Enumeration	Services identified	No detection	100%

C. Flow Log Evidence

Azure Network Watcher flow logs (`PT1H.json`) provided forensic verification: 42 internal connection attempts logged as ”Deny” while 22 SSH connections logged as ”Allow”, confirming selective traffic blocking.

TABLE IV
NETWORK SECURITY GROUP FLOW LOG VALIDATION

Observation	**Flow Log Evidence**
Ping (ICMP) traffic	Denied NSG flow logged
Telnet (TCP/23) traffic	Denied NSG flow logged
Port scanning traffic	Denied NSG flow logged
SSH administrative traffic	Allowed NSG flow logged

D. Operational Transparency Verification

While lateral movement was blocked, administrative functions remained accessible: SSH success rate maintained at 100% (22/22 attempts) and Azure Portal management was unaffected by NSG rules.

V. DISCUSSION

The experimental findings strongly validate both research hypotheses while offering practical insights for cloud security implementation.

A. Hypothesis Validation

H1 (Effective Containment) is quantitatively confirmed: The 100% containment rate across all test vectors demonstrates that Azure NSGs can completely isolate threats within subnets when properly configured, validating micro-segmentation theory [1].

H2 (Operational Transparency) is operationally verified: Maintaining 100% SSH accessibility while blocking malicious traffic demonstrates that security and functionality are not mutually exclusive in cloud environments [3].

B. Scope and Claims

This study demonstrates a specific security control: Azure NSGs effectively block lateral movement at the subnet level. It does not claim to address all attack vectors (e.g., application layer attacks, insider threats, or advanced persistent threats). The focused scope allows clear validation of micro-segmentation's fundamental value in cloud network design.

C. Methodological Reflections

As noted in project plan feedback, our simulation represents common post exploitation techniques rather than true worm propagation. This methodological choice provides actionable insights for real-world defense while maintaining ethical research boundaries.

D. Validity Considerations

Internal Validity: Controlled variables (VM size, OS, network) were held constant to isolate NSG effects. **External Validity:** Results generalize to flat Azure networks, though organizational variations may affect outcomes. **Construct Validity:** Tests validly represent reconnaissance/lateral movement techniques.

E. Practical Implications

- **Immediate Implementation:** Organizations can achieve immediate risk reduction through NSG configuration
- **Cost Effective:** NSGs provide enterprise segmentation at no additional cost
- **Compliance Alignment:** Supports frameworks requiring network segmentation

F. Limitations and Future Work

Limitations include the two VM scope and simulated threat behavior. Future research could explore multi-subnet architectures and advanced threat simulations.

Ethical Considerations

All testing was conducted within a private Azure subscription under academic license. No real malware was deployed; simulated attacks used only safe scanning tools. Network traffic was confined to the isolated virtual network, complying with Azure acceptable use policies.

VI. Conclusion

This study demonstrates that properly configured Azure Network Security Groups can effectively contain simulated lateral threat movement between virtual machines in cloud subnets. The experimental results show 100% blockage of internal threat propagation while maintaining administrative access, validating both the security efficacy and operational practicality of NSG based micro-segmentation. These findings provide empirical evidence supporting the implementation of network segmentation controls in cloud environments, addressing the widespread problem of overly permissive security groups identified in contemporary threat reports. The experimental approach validates the core claim while maintaining academic integrity: subnet level segmentation works for its intended purpose. This focused validation provides stronger evidence than broader, less testable claims about cloud security overall.

References

[1] H. A. Al-Ofeishat and R. Alshorman, "Build a secure network using segmentation and micro-segmentation techniques," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 1, pp. 1048–1057, 2024. [Online]. Available: https://ijece.iaescore.com/index.php/IJECE/article/view/54671

[2] Palo Alto Networks Unit 42, "Unit 42 cloud threat report, volume 7," October 2023, accessed: 2025-11-17. [Online]. Available: https://www.paloaltonetworks.com/unit42/cloud-threat-report-vol-7

[3] Microsoft, *What is a Network Security Group?*, Microsoft Azure Documentation, August 2023, accessed: 2025-11-17. [Online]. Available: https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview

[4] C. Benzaïd, N. Guerd, N. El Houda Rehouma, K. Zeraoulia, and T. Taleb, "A multi-layered zero trust microsegmentation solution for cloud-native 5g & beyond networks," in *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, 2025, pp. 1–7. [Online]. Available: https://doi.org/10.1109/WCNC61545.2025.10978671

[5] M. Arifeen, A. Petrovski, and S. Petrovski, "Automated microsegmentation for lateral movement prevention in industrial internet of things (iiot)," in *2021 14th International Conference on Security of Information and Networks (SIN)*, vol. 1, 2021, pp. 1–6. [Online]. Available: https://doi.org/10.1109/SIN54109.2021.9699232

[6] group07, "Network-worm-containment-with-azure-nsgs: Supporting material for academic project," https://github.com/FilmonMeharii/network-worm-containment-with-azure-nsgs, January 2026, gitHub Repository. Accessed: 2026-01-15.