

---

## **INTELLIGENT CLOUD INFRASTRUCTURE REVIEW WITH GENAI ASSISTANCE**

---

### **1. Introduction**

This document presents a security and optimization review of a simulated cloud infrastructure environment. The review focuses on a Linux-based virtual machine running Docker containers, along with associated storage and networking components. GenAI tools were used in an advisory role to assist in identifying potential risks and recommending improvements.

### **2. Assumed Infrastructure Architecture**

The reviewed environment consists of a Linux virtual machine with Docker installed. An Ubuntu 22.04 container is deployed to simulate an application workload. Application logs are generated inside the container and stored locally. Network access is assumed through the virtual machine's default networking configuration.

### **3. Docker Configuration Review**

Docker is installed on the Linux virtual machine and used to run an Ubuntu-based container. The container operates with default privileges and does not have explicit CPU or memory limits configured. Logs are generated within the container to simulate application and system activity.

### **4. Log Review and Observations**

The application log file contains informational messages, warning events, and error entries. These logs indicate potential issues related to resource usage and storage operations. Reviewing these logs helps identify areas where monitoring, alerting, and optimization can be improved.

### **5. Identified Risks and Misconfigurations**

Component	Observation	Risk
Docker	Container runs as root user	Privilege escalation risk
Docker	No CPU or memory limits configured	Resource exhaustion
Logging	Logs stored locally inside container	Lack of centralized monitoring
Network	No isolation or firewall rules defined	Increased attack surface

### **6. GenAI-Assisted Insights**

GenAI tools were used in a consultative and analytical capacity to assist in reviewing the infrastructure configuration and application logs. The GenAI support helped summarize common security risks, identify potential misconfigurations, and suggest industry-standard best practices. No GenAI models were deployed within the infrastructure, and no automated actions were performed based on GenAI output.

## 7. Security Hardening Recommendations

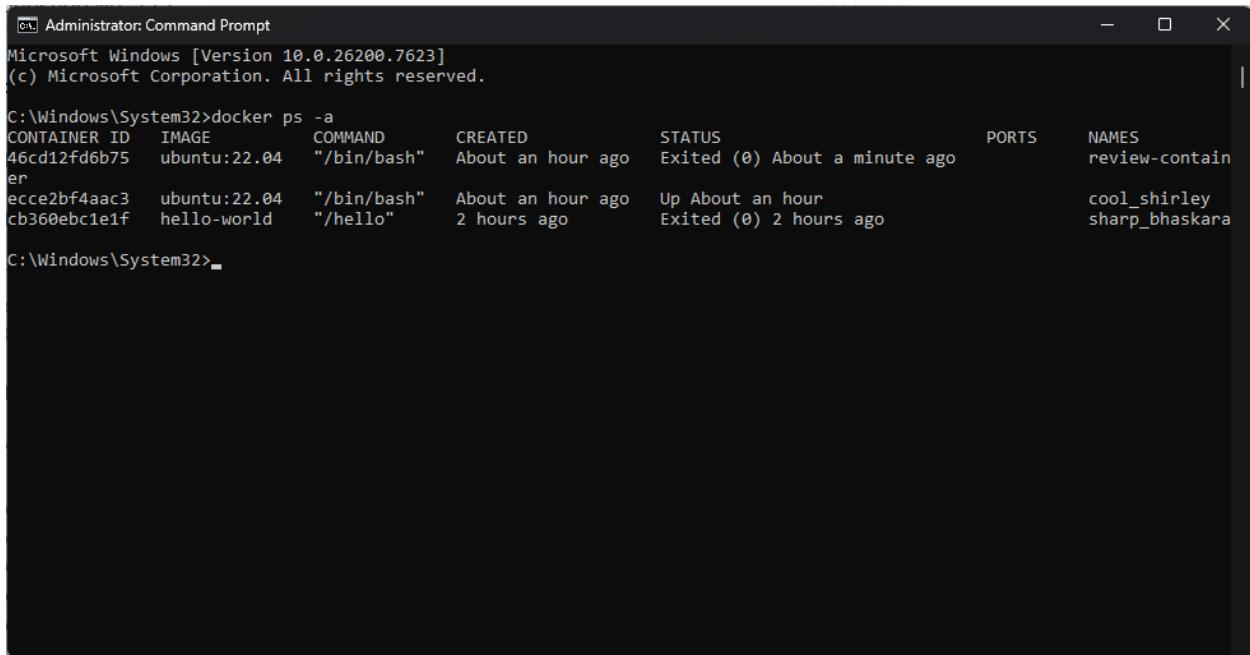
- Run Docker containers using non-root users
- Apply CPU and memory limits to containers
- Centralize log collection and monitoring
- Implement network segmentation and firewall rules
- Use minimal and secure base images for containers

## 8. Optimized Architecture Proposal

An optimized architecture would include a hardened Linux virtual machine hosting Docker with restricted container privileges. Centralized logging and monitoring services should be enabled to improve observability. Network access should be controlled using firewall rules, and storage should be secured to prevent unauthorized access.

## 9. Implementation Evidence (Screenshots)

Figure 1: Active Docker container running on a Linux virtual machine

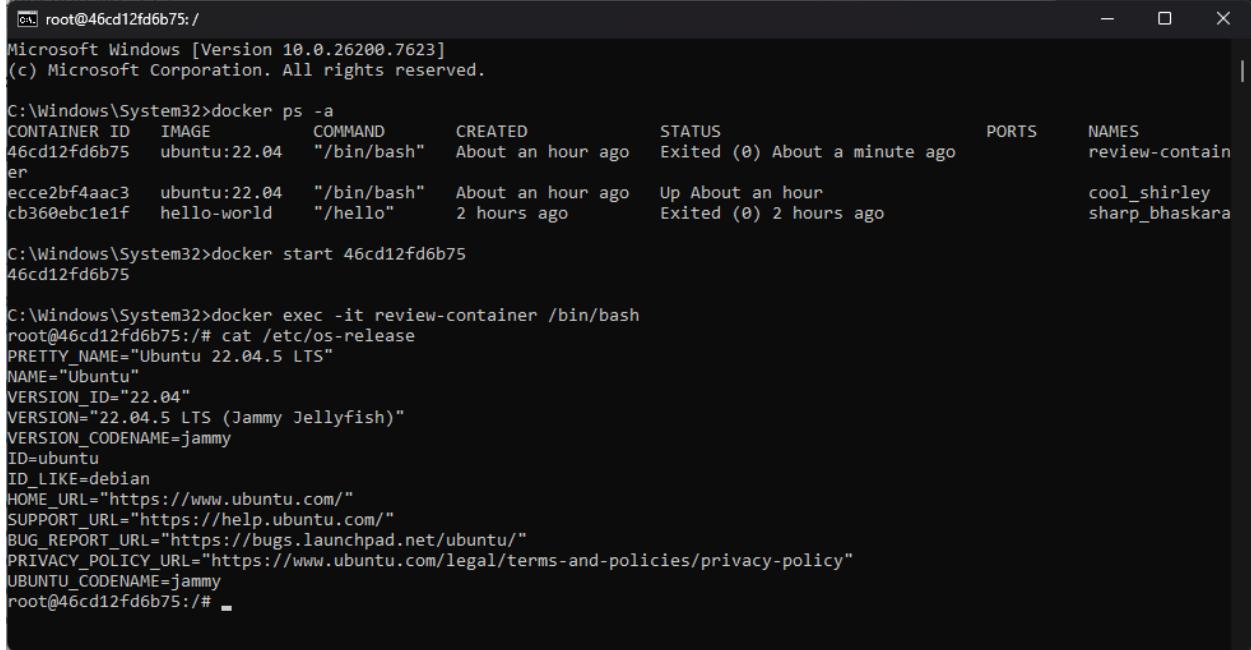


```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.26200.7623]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>docker ps -a
CONTAINER ID   IMAGE      COMMAND       CREATED        STATUS          PORTS     NAMES
46cd12fd6b75   ubuntu:22.04  "/bin/bash"   About an hour ago   Exited (0)   About a minute ago
er
ecce2bf4aac3   ubuntu:22.04  "/bin/bash"   About an hour ago   Up  About an hour
cb360ebc1e1f   hello-world  "/hello"      2 hours ago    Exited (0)   2 hours ago
                                         NAMES
review-contain
cool_shirley
sharp_bhaskara

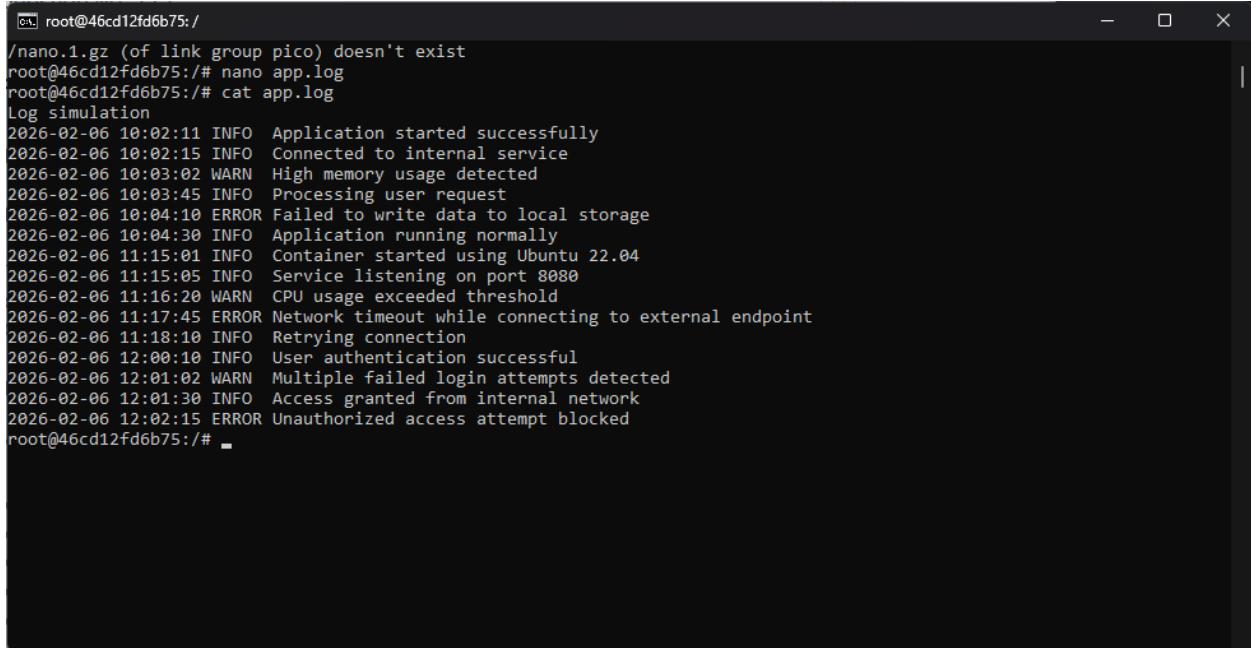
C:\Windows\System32>
```

Figure 2: Ubuntu Linux environment running inside the Docker container



```
root@46cd12fd6b75:/  
Microsoft Windows [Version 10.0.26200.7623]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Windows\System32>docker ps -a  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
46cd12fd6b75 ubuntu:22.04 "/bin/bash" About an hour ago Exited (0) About a minute ago  
er  
ecce2bf4aac3 ubuntu:22.04 "/bin/bash" About an hour ago Up About an hour  
cb360ebc1e1f hello-world "/hello" 2 hours ago Exited (0) 2 hours ago  
  
C:\Windows\System32>docker start 46cd12fd6b75  
46cd12fd6b75  
  
C:\Windows\System32>docker exec -it review-container /bin/bash  
root@46cd12fd6b75:/# cat /etc/os-release  
PRETTY_NAME="Ubuntu 22.04.5 LTS"  
NAME="Ubuntu"  
VERSION_ID="22.04"  
VERSION="22.04.5 LTS (Jammy Jellyfish)"  
VERSION_CODENAME=jammy  
ID=ubuntu  
ID_LIKE=debian  
HOME_URL="https://www.ubuntu.com/"  
SUPPORT_URL="https://help.ubuntu.com/"  
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"  
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"  
UBUNTU_CODENAME=jammy  
root@46cd12fd6b75:/#
```

Figure 3: Sample application log reviewed inside the container



```
root@46cd12fd6b75:/  
/nano.1.gz (of link group pico) doesn't exist  
root@46cd12fd6b75:/# nano app.log  
root@46cd12fd6b75:/# cat app.log  
Log simulation  
2026-02-06 10:02:11 INFO Application started successfully  
2026-02-06 10:02:15 INFO Connected to internal service  
2026-02-06 10:03:02 WARN High memory usage detected  
2026-02-06 10:03:45 INFO Processing user request  
2026-02-06 10:04:10 ERROR Failed to write data to local storage  
2026-02-06 10:04:30 INFO Application running normally  
2026-02-06 11:15:01 INFO Container started using Ubuntu 22.04  
2026-02-06 11:15:05 INFO Service listening on port 8080  
2026-02-06 11:16:20 WARN CPU usage exceeded threshold  
2026-02-06 11:17:45 ERROR Network timeout while connecting to external endpoint  
2026-02-06 11:18:10 INFO Retrying connection  
2026-02-06 12:00:10 INFO User authentication successful  
2026-02-06 12:01:02 WARN Multiple failed login attempts detected  
2026-02-06 12:01:30 INFO Access granted from internal network  
2026-02-06 12:02:15 ERROR Unauthorized access attempt blocked  
root@46cd12fd6b75:/#
```

## **10. Conclusion**

This review demonstrates how Docker-based workloads running on virtual machines can be analyzed for security and optimization concerns. The use of GenAI as an advisory tool enabled efficient identification of risks and improvement recommendations, contributing to an improved cloud security posture.