Project 2: Cloud Server Hardening & Secure Access (SSH + Firewall + IAM)

Student: Vishal Sundi **Platform:** Microsoft Azure (Free Student Credits)
Environment: Ubuntu 22.04 LTS

Objective
Deploy and secure a Linux cloud server following modern cybersecurity best practices. This includes limiting SSH access, disabling root login, configuring UFW firewall, installing Fail2Ban, and enabling Auditd for system auditing.

Step 1: Deploy Ubuntu VM on Azure

Use the Azure Portal to create a new Ubuntu 22.04 LTS VM. Choose SSH Public Key authentication and allow only port 22. Set up a Resource Group, Virtual Network, and Network Security Group (NSG). Include screenshots of the Azure Portal showing VM deployment overview.

Step 2: Create a New Admin User

Create a non-root admin user and add them to the sudo group. Create a sudoers file to allow sudo without editing /etc/sudoers directly.

```
# Connect to the VM (example)
ssh azureuser@<VM_PUBLIC_IP>

# Create a new user and add to sudo group
sudo adduser adminuser
sudo usermod -aG sudo adminuser

# Create sudoers file for the admin user
echo "adminuser ALL=(ALL) NOPASSWD:ALL" | sudo tee /etc/sudoers.d/adminuser
sudo chmod 440 /etc/sudoers.d/adminuser
```

Step 3: Disable Root SSH Login

Edit the SSH daemon configuration to disable root login and password authentication for SSH to force key-based auth. Then restart the SSH service.

```
# Edit sshd_config (use nano, vim, or your editor of choice)
sudo sed -i 's/^#\?PermitRootLogin.*/PermitRootLogin no/' /etc/ssh/sshd_config
sudo sed -i 's/^#\?PasswordAuthentication.*/PasswordAuthentication no/' /etc/ssh/sshd_co

# Restart SSH service
sudo systemctl restart sshd
# Verify the SSH service status
sudo systemctl status sshd --no-pager
```

Step 4: Enable SSH Key Authentication

Generate SSH key pair locally and add the public key to the new admin user's authorized_keys to allow key-based login.

```
# Generate key pair locally (on your machine)
ssh-keygen -t ed25519 -C "your_email@example.com"

# Copy public key to server (example)
ssh-copy-id -i ~/.ssh/id_ed25519.pub adminuser@<VM_PUBLIC_IP>

# Or manually create the authorized_keys file on the server
mkdir -p /home/adminuser/.ssh
echo "<PUBLIC_KEY_CONTENT>" | sudo tee /home/adminuser/.ssh/authorized_keys
sudo chown -R adminuser:adminuser /home/adminuser/.ssh
sudo chmod 700 /home/adminuser/.ssh
sudo chmod 600 /home/adminuser/.ssh/authorized_keys
```

Step 5: Configure UFW Firewall

Install and enable UFW (Uncomplicated Firewall). Allow OpenSSH and deny or limit other incoming traffic as required.

```
# Install UFW if needed
sudo apt update
sudo apt install -y ufw

# Allow OpenSSH, enable UFW, and check status
sudo ufw allow OpenSSH
sudo ufw enable
sudo ufw status verbose
```

Step 6: Install & Configure Fail2Ban

Install Fail2Ban to protect against SSH brute-force attempts. Start the service and verify bans by checking the logs.

```
# Install Fail2Ban
sudo apt update
sudo apt install -y fail2ban

# Start and enable the service
sudo systemctl enable --now fail2ban

# Check Fail2Ban status and jail status
sudo fail2ban-client status
sudo fail2ban-client status sshd

# To simulate failed attempts, try wrong credentials or use ssh with incorrect key
```

Step 7: Enable Audit Logging (auditd)

Install auditd to capture detailed system events and administrative actions. Use ausearch to query logs.

```
# Install auditd
sudo apt update
sudo apt install -y auditd audispd-plugins

# Start and enable auditd
sudo systemctl enable --now auditd

# Search audit logs for sudo and login events
sudo ausearch -m USER_CMD,USER_LOGIN -ts recent
# Example: show events for a specific user
sudo ausearch -ua adminuser
```

Step 8 (Bonus): Configure Wazuh Log Shipping

Optionally install the wazuh-agent and connect it to your Wazuh manager for centralized log analytics.

```
# On the agent (example)
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | sudo apt-key add -
echo 'deb https://packages.wazuh.com/4.x/apt stable main' | sudo tee /etc/apt/sources.li
sudo apt update
sudo apt install -y wazuh-agent

# Configure the manager IP in /var/ossec/etc/ossec.conf and start agent
sudo systemctl enable --now wazuh-agent
```

Reflection / Learning Outcome

This project improved understanding of secure cloud deployments, Linux hardening, and intrusion prevention. Practical skills learned include managing SSH keys, configuring firewalls, using Fail2Ban for brute-force protection, and auditing with auditd.

Conclusion

By implementing these measures the server is more secure against unauthorized SSH access, brute-force attacks, and now provides audit traceability for administrative actions.