Q5: SecurityContext Implementation - Non-Root Deployment

Overview

This document demonstrates the implementation of SecurityContext to prevent root privileges in Kubernetes deployments.

Modified Deployment

SecurityContext Configuration

File: `manifests/aks-store-quickstart-secure.yaml`

The product-service deployment now includes:

```yaml
securityContext:
  runAsNonRoot: true
  runAsUser: 1000
  runAsGroup: 1000
  fsGroup: 1000
containers:
  - name: product-service
    securityContext:
      allowPrivilegeEscalation: false
      readOnlyRootFilesystem: false
      runAsNonRoot: true
      runAsUser: 1000
      capabilities:
        drop:
          - ALL
```

Screenshots

1. AKS Store Deployment with Security

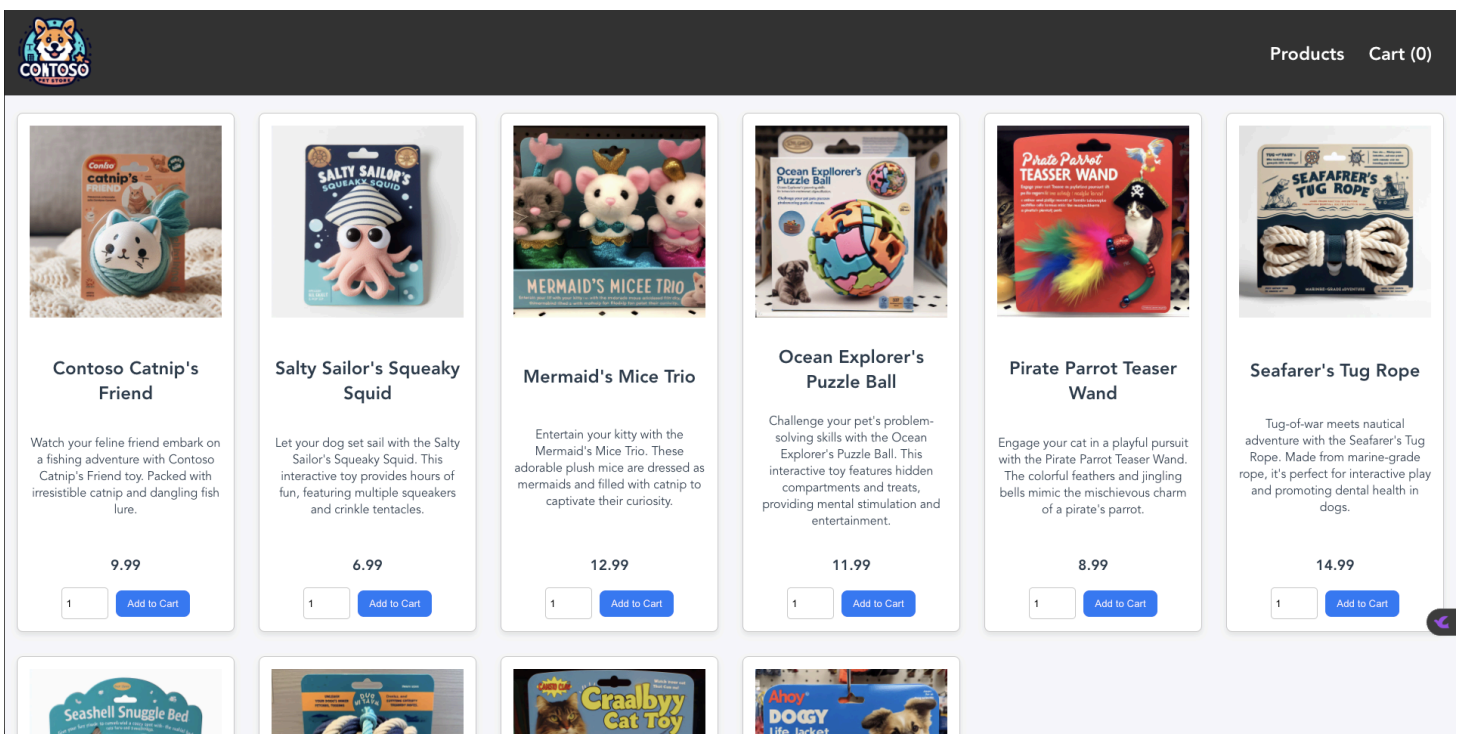
```
munachiernest-eze@Munachis-MacBook-Pro BCDV4034-FinalExam % kubectl get po
 && kubectl get services -n aks-store-demo && echo "---" && kubectl get po
 && kubectl get services -n aks-store-demo && echo "---" && kubectl get de
NAME                                READY   STATUS    RESTARTS   AGE
order-service-5c85f45984-956bv      1/1     Running   0          10m
product-service-5db5645fb7-lpgbz    1/1     Running   0          2m39s
rabbitmq-0                          1/1     Running   0          10m
store-front-6ff78d4f79-6nj4w        1/1     Running   0          2m38s
---
NAME              TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)
order-service     ClusterIP      10.0.196.199   <none>        3000/TCP
product-service   ClusterIP      10.0.111.1     <none>        3002/TCP
rabbitmq          ClusterIP      10.0.123.245   <none>        5672/TCP,156
store-front       LoadBalancer   10.0.15.153    <pending>     80:30520/TCP
---
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
order-service     1/1     1            1           10m
product-service   1/1     1            1           10m
store-front       1/1     1            1           10m
munachiernest-eze@Munachis-MacBook-Pro BCDV4034-FinalExam %

sk every time                                   Move to background
```

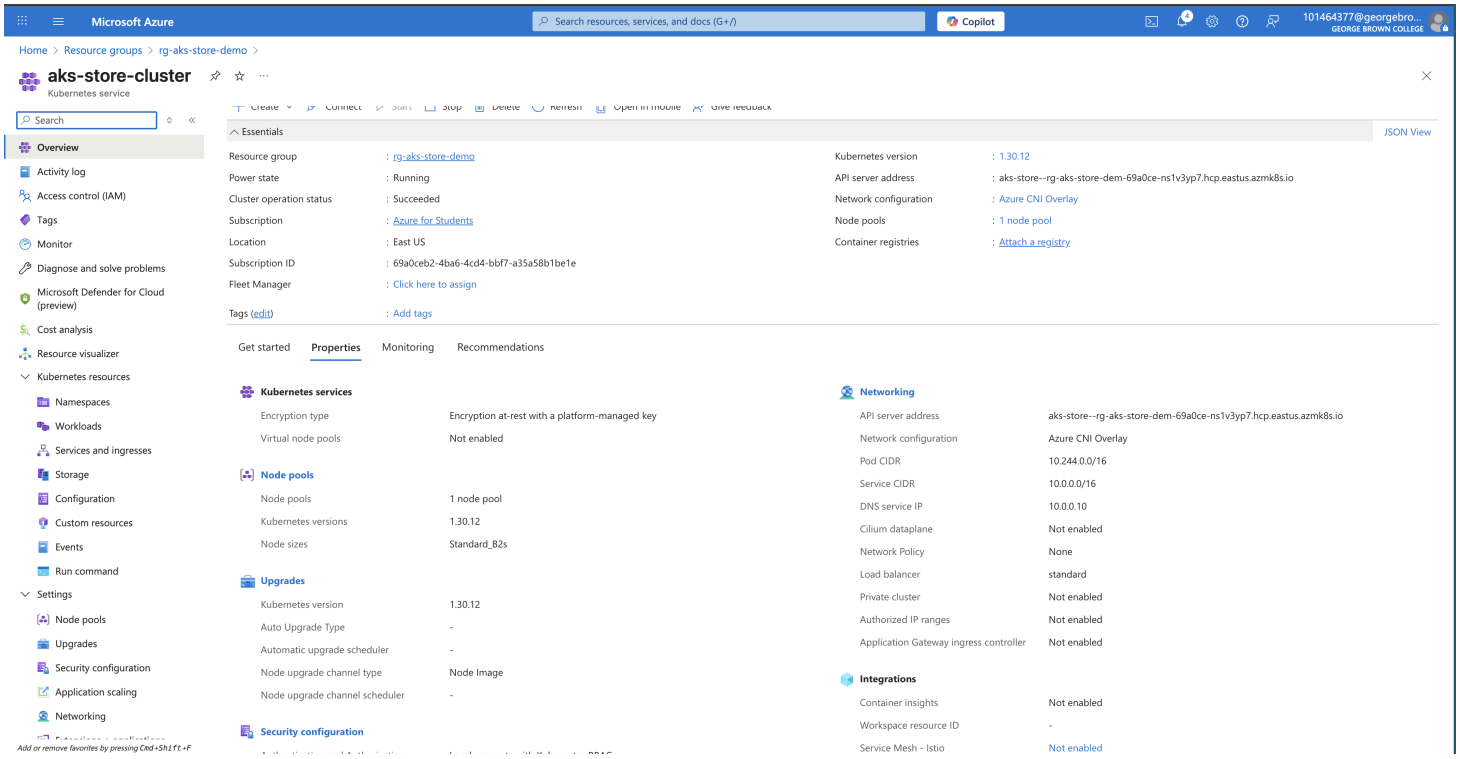
2. SecurityContext Verification

```
munachiernest-eze@Munachis-MacBook-Pro BCDV4034-FinalExam % kubectl get pod product-service-5db5645fb7-lpgbz -n aks-store-demo -o yaml | grep -A 20 securi
tyContext     kubectl get pod product-service-5db5645fb7-lpgbz -n aks-store-demo -o yaml | grep -A 20 securityContext
    securityContext:
      allowPrivilegeEscalation: false
      capabilities:
        drop:
        - ALL
      readOnlyRootFilesystem: false
      runAsNonRoot: true
      runAsUser: 1000
    terminationMessagePath: /dev/termination-log
    terminationMessagePolicy: File
    volumeMounts:
    - mountPath: /var/run/secrets/kubernetes.io/serviceaccount
      name: kube-api-access-zvdzp
      readOnly: true
  dnsPolicy: ClusterFirst
  enableServiceLinks: true
  nodeName: aks-default-39692054-vmss000002
  nodeSelector:
    kubernetes.io/os: linux
  preemptionPolicy: PreemptLowerPriority
  priority: 0
--
  securityContext:
    fsGroup: 1000
    runAsGroup: 1000
    runAsNonRoot: true
    runAsUser: 1000
  serviceAccount: default
  serviceAccountName: default
  terminationGracePeriodSeconds: 30
  tolerations:
  - effect: NoExecute
    key: node.kubernetes.io/not-ready
    operator: Exists
    tolerationSeconds: 300
  - effect: NoExecute
    key: node.kubernetes.io/unreachable
    operator: Exists
    tolerationSeconds: 300
  - effect: NoSchedule
    key: node.kubernetes.io/memory-pressure
    operator: Exists
  volumes:
```

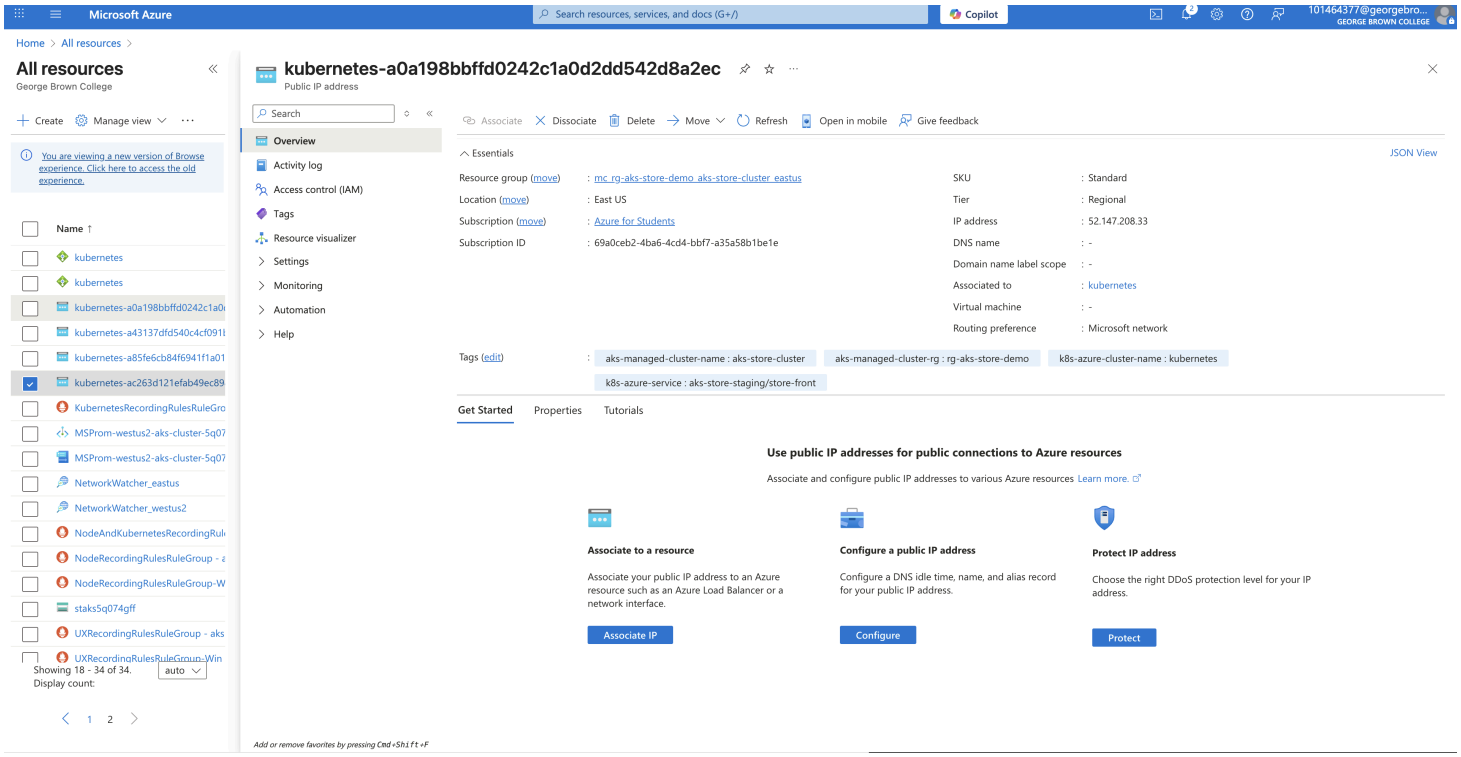
3. Application Running with Non-Root User



4. Cluster with Deployments



5. Deployment Status



Security Features Implemented

- **runAsNonRoot**: Prevents containers from running as root

- **runAsUser**: Runs container as user ID 1000
- **allowPrivilegeEscalation**: Prevents privilege escalation
- **capabilities.drop**: Removes all Linux capabilities
- **fsGroup**: Sets file system group ownership

Files Submitted

- `manifests/aks-store-quickstart-secure.yaml` - Deployment with SecurityContext
- `deploy-aks-store-secure.sh` - Deployment script