List of threats used in the First experiment (Kubernetes Scenario)

Each threat has a unique ID alongside a description, the corresponding STRIDE threat (i.e., Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege), and assumptions used to validate the existence of the threat. The last field, whether the threat is valid or not, was not shown to the participants.

Unique Qualtrics ID	Description	STRIDE threat	Assumption	Affected Components	Real
LEAKED-PRIVILEGE-REMOTE	An attacker uses a leaked cluster configuration file to interact with the cluster, potentially jeopardizing running workloads and creating malicious pods.	EoP	The attacker finds the leaked information and can interact with the cluster.	Not present in the DFD.	Yes
SPOOFING-AUTH-WORKLOAD	Spoofing a cluster adminby stealing the authentication credentialsvia a social engineering attack.	Spoofing	The attacker carries out a successful spoofing attack with valid credentials.	-Cluster configuration -Running workloads (pods, container, etc.) - Access control (of users and services) In general, he/she can take over all clusters.	Yes
DOS-WORKERNODE	An attacker with shell access to only one pod which has no resource limits applied can crash the worker node on which that pod is running, causing a DoS.	DoS	The attacker got a remote shell on a pod.	Compromised pod, worker node, and all the other pods running on the same node.	Yes
ELEVATION-PRIVILEGE-MALICIOUS-IMG	An attacker is able to upload images to a container registry from which a K8s cluster retrieves pods images, he or she can potentially execute a malicious pod inside the cluster and get a shell from the malicious container.	EoP	The attacker has permission to upload or modify images in the K8s registry.	-Images Registry - Running pod	Yes
EXPLOIT-PRIVILEGED-CONTAINER	Privileged containers run as root on the host, thus an attacker compromisingone of such containers, gets automatically root access on the host.	EoP	1. The attacker gets access to a privileged container running in the cluster 2. The attacker escapes the container.	Pod, worker node	Yes

PORT-JAMMING-NETWORK-POLICIES	If an attacker deploys a malicious pod into a namespace of the cluster with network policies in place, the attacker can use such a pod to send bogus network packagesto pods in different namespaces to jam the exposed ports, causing adisruption. bogus network packagesto pods in different namespaces to jam the exposed ports, causing adisruption.	DoS	The network policies are implemented correctly to segment the namespaces.	Pods in different namespaces	No
LEAKED-SECRET-DOCKERFILE	If a developer embeds a secret in a Dockerfile A and then builds an image (image A), an attacker with access to image A can reverse the image's layers to not only observethe embedded secret in Dockerfile A but also from other images that were built from different Dockerfiles.	Information Disclosure	The attacker has access to the image.	Cluster secrets/credentials	No
CHAIN-ATTACK-MALICIOUS-INPUTS	If an attacker compromises a running container, he/she can only continue exploitingthe container by runningthe software specified inDockerfile (by the developers) with malicious inputs.	EoP	The attacker got access to a running container.	Running pod/container	No
UNAUTH-CONFIG-TAMPERING	An unauthenticated and non-privileged attacker can still upload custom pod configurations intothe cluster.	EoP	The attacker can reach the cluster API server (e.g., through the internet)	New pods	No
SPOOFING-LAYER-3	If an attacker compromises a pod in a K8s cluster using a Layer3 network plugin, he/she can steal other pods' identities and laterally move within the cluster using the network bridge.	Spoofing	The attacker exploits one container and the CNI works at layer 3.	Lateral movement between pods	No