

# **PodSecurityPolicy is Dead, Long Live...?**

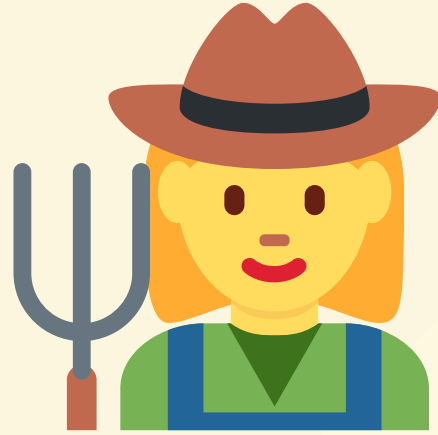
**Chris Nesbitt-Smith**

**Appvia | LearnK8s**



```
kubectl get pods
```





```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
    - name: nginx
      image: nginx:1.14.2
      ports:
        - containerPort: 80
```



# PodSecurityWhat?



kind: PodSecurityPolicy



```
apiVersion: policy/v1beta1  
kind: PodSecurityPolicy
```







# `kubectl explain PodSecurityPolicy`

**Pod Security Policies enable fine-grained authorization of pod creation and updates.**

**A Pod Security Policy is a cluster-level resource that controls security sensitive aspects of the pod specification. The PodSecurityPolicy objects define a set of conditions that a pod must run with in order to be accepted into the system, as well as defaults for the related fields.**

**<https://kubernetes.io/docs/concepts/policy/pod-security-policy/>**



# kubectl explain PodSecurityPolicy

Pod Security Policies enable **fine-grained authorization of pod creation and updates.**

A Pod Security Policy is a **cluster-level** resource that controls security sensitive aspects of the pod specification. The PodSecurityPolicy objects define a set of conditions that a pod must run with in order to be accepted into the system, as well as defaults for the related fields.



```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: example
spec:
  privileged: false
  seLinux:
    rule: RunAsAny
  supplementalGroups:
    rule: RunAsAny
  runAsUser:
    rule: RunAsAny
  fsGroup:
    rule: RunAsAny
  volumes:
    - "*"

```



```
apiVersion: v1
kind: Pod
metadata:
  name: demo
spec:
  containers:
    - name: demo
      image: alpine
      securityContext:
        privileged: true
```



**Live demo**



```
apiVersion: v1
kind: Pod
metadata:
  name: demo
spec:
  containers:
    - name: demo
      image: alpine
      volumeMounts:
        - mountPath: /storage
          name: storage
  volumes:
    - name: storage
      hostPath:
        path: /
        type: Directory
```



```
apiVersion: v1
kind: Pod
metadata:
  name: demo
spec:
  hostNetwork: true
  containers:
    - name: demo
      image: alpine
```



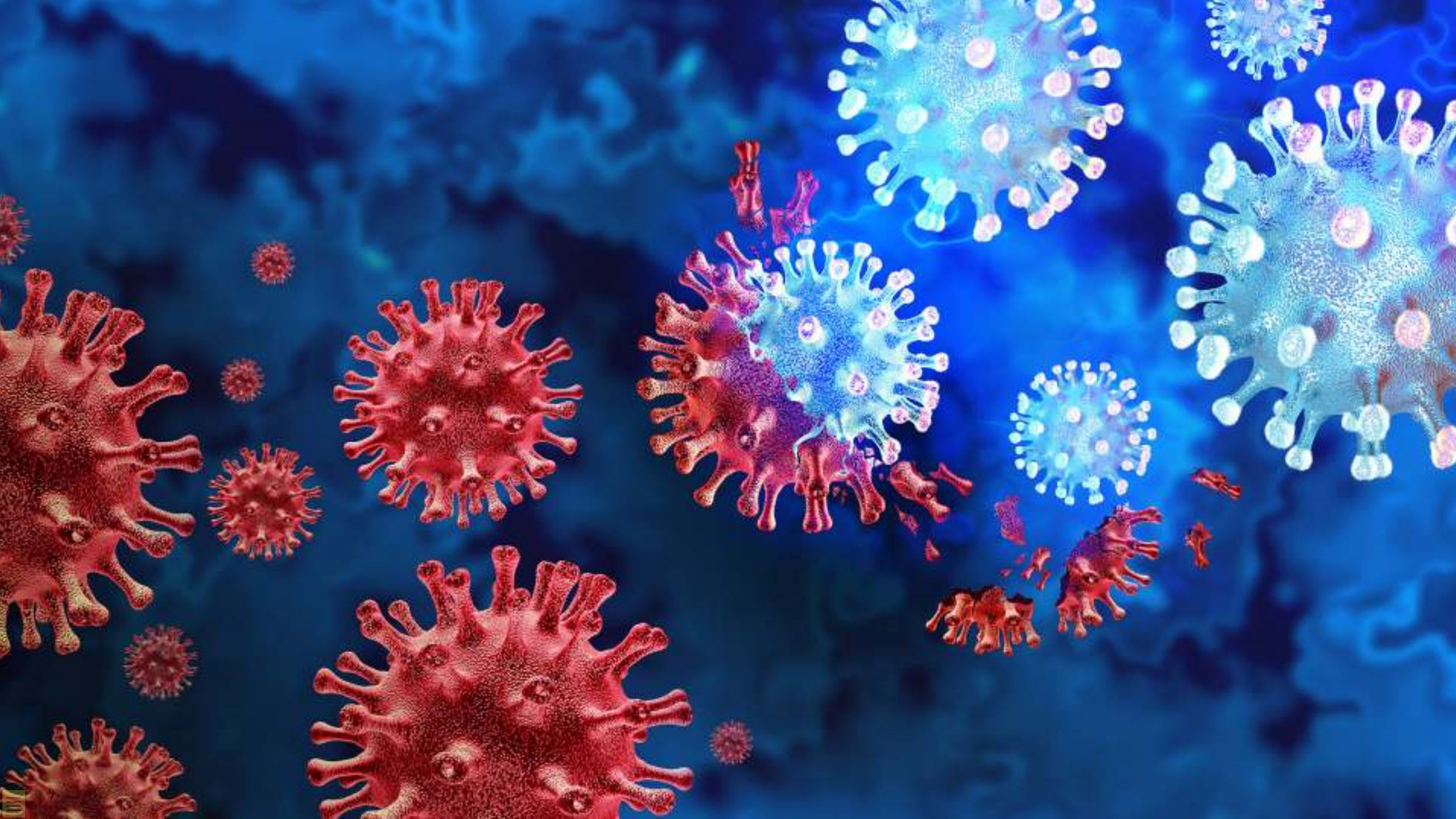


**E T C .**









**OUT OF ORDER**



**So now what?**





**Admission Control Anchore**  
**Azure Policy Istio jspolicy K-**  
**rail Kopf Kubewarden**  
**Kyverno OPA Gatekeeper**  
**Opslevel Polaris Prisma Cloud**  
**Qualys Regula Sysdig TiDB**



Admission Control Anchore

Azure Policy Istio jspolicy K-

rail Kopf **Kubewarden**

**Kyverno OPA Gatekeeper**

Opslevel Polaris Prisma Cloud

Qualys Regula Sysdig TiDB



**Wait, what about  
Pod Security Standards  
&  
Pod Security Admission?**



# Privileged



# Baseline



# Restricted





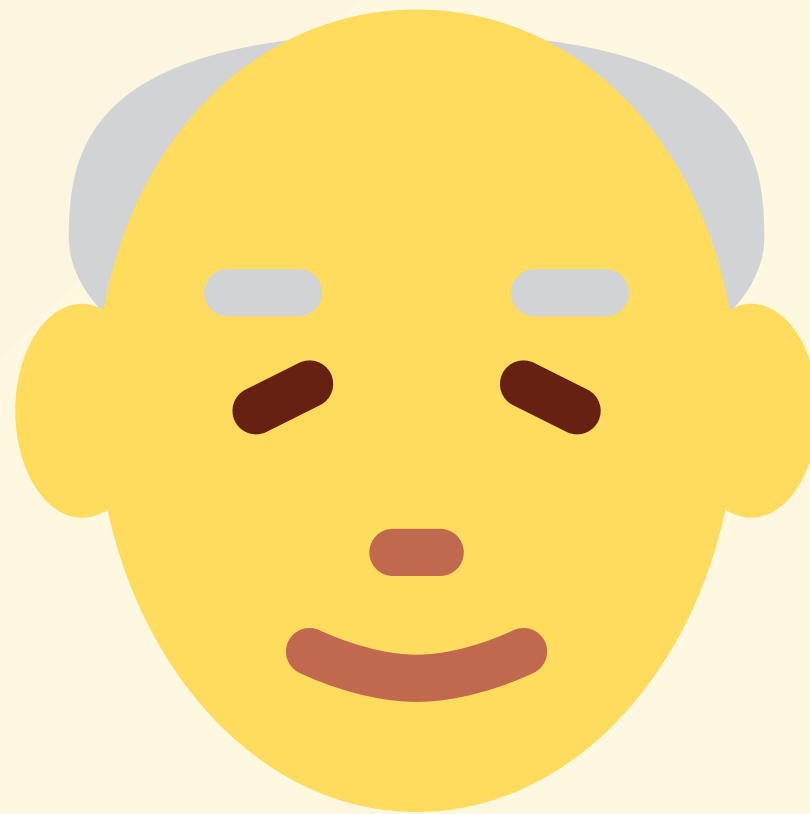




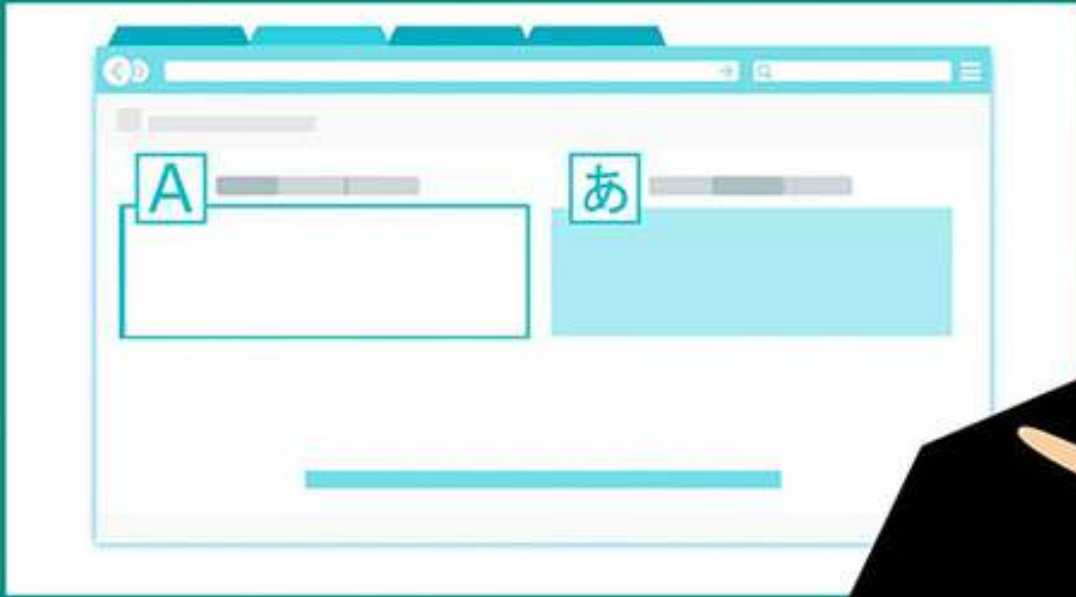












vigl:privileged cns\$ c

# PodSecurityPolicy Migrator

UPLOAD

Select an example

GATEKEEPER

KUBEWARDEN

KYVERNO

```
1 apiVersion: policy/v1beta1
2 kind: PodSecurityPolicy
3 metadata:
4   name: policy
5 spec:
6
7   runAsUser:
8     rule: 'RunAsAny'
9   selinux:
10    rule: 'RunAsAny'
11   fsGroup:
12    rule: 'RunAsAny'
13   supplementalGroups:
14    rule: 'RunAsAny'
15   volumes:
16    - '*'
```

```
1
```

↑ PodSecurity Policy goes here ↑

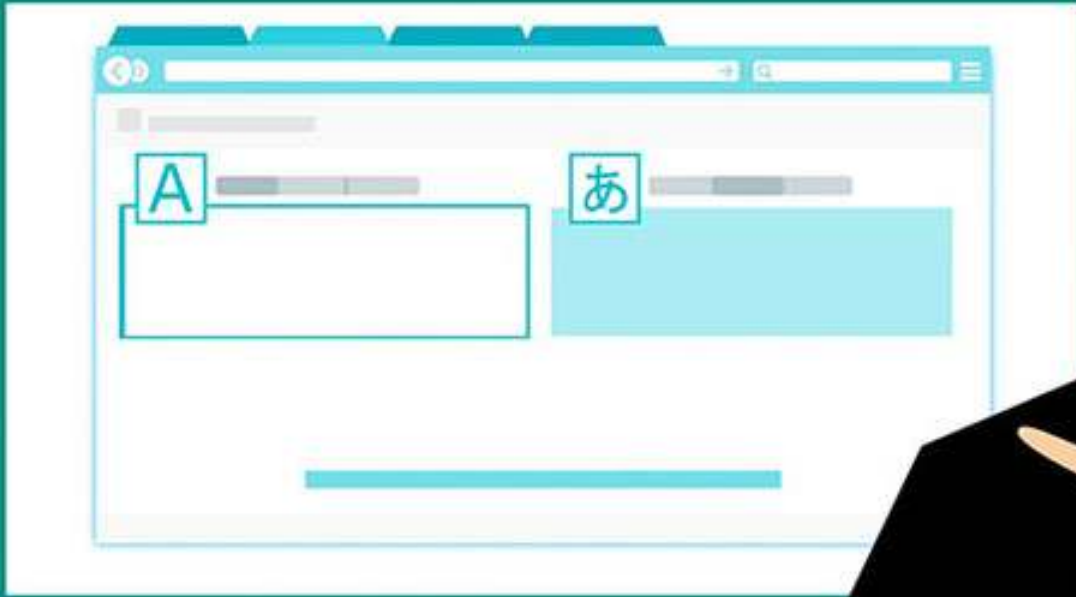
↑ Alternative Policy comes out here ↑



REPORT A BUG







Live demo



# PodSecurityPolicy

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: example
spec:
  privileged: false
  seLinux:
    rule: RunAsAny
  supplementalGroups:
    rule: RunAsAny
  runAsUser:
    rule: RunAsAny
  fsGroup:
    rule: RunAsAny
  volumes:
    - "*"

```



# Kyverno

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
  name: example
spec:
  rules:
    - validate:
      pattern:
        spec:
          =(initContainers)":
            - =(securityContext)":
              =(privileged)": false
          =(ephemeralContainers)":
            - =(securityContext)":
              =(privileged)": false
        containers:
          - =(securityContext)":
            =(privileged)": false
      message: Rejected by psp-privileged-0 rule
  match:
    resources:
      kinds:
        - Pod
  name: psp-privileged-0
```

# Kubewarden

```
apiVersion: policies.kubewarden.io/v1alpha2
kind: ClusterAdmissionPolicy
metadata:
  name: example
spec:
  module: registry://ghcr.io/kubewarden/policies/pod-privileged:v0.1.9
  rules:
    - apiGroups:
      - ""
      apiVersions:
      - v1
      resources:
      - pods
      operations:
      - CREATE
      - UPDATE
  mutating: false
  settings: null
```



# OPA Gatekeeper

```
apiVersion: constraints.gatekeeper.sh/v1beta1
kind: K8sPSPPrivilegedContainer
metadata:
  name: example
spec:
  match:
    kinds:
      - apiGroups:
          - ""
        kinds:
          - Pod
  parameters: null
```





**But, should you migrate from  
PodSecurityPolicy?**







PWNKIT



CVE-2021-4034

**Sorry**

**(not sorry)**



**EXTRA! EXTRA!**

**READ ALL ABOUT IT!**

**GOOD  
NEWS!!**





**SPOILER**  
**ALERT!**

**AppArmor Continuous Integration  
Cultural Change eBPF GitOps Keep it  
Stupid Simple Kernel Level Protection  
Policy as code seccomp Secure By  
Design Security Profiles Operator  
SELinux Shared Responsibility Model  
Shift Left Testing Version Controlled  
Policy Zero trust**



🙏 Thanks 🙏

- [cns.me](https://cns.me)
- [github.com/chrisns](https://github.com/chrisns)
- [github.com/appvia](https://github.com/appvia)
- [appvia.io/blog](https://appvia.io/blog)

**Chris Nesbitt-Smith**

