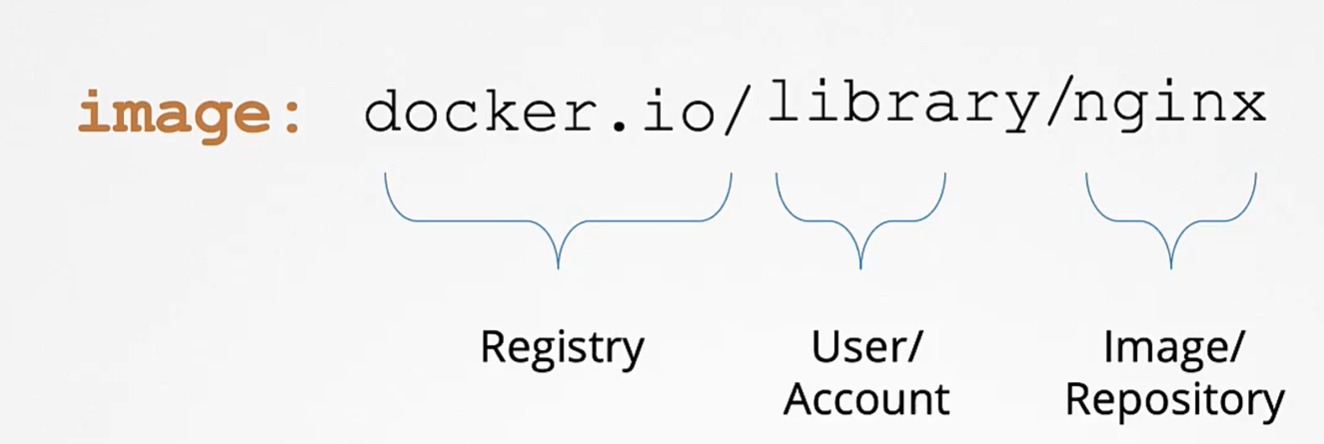
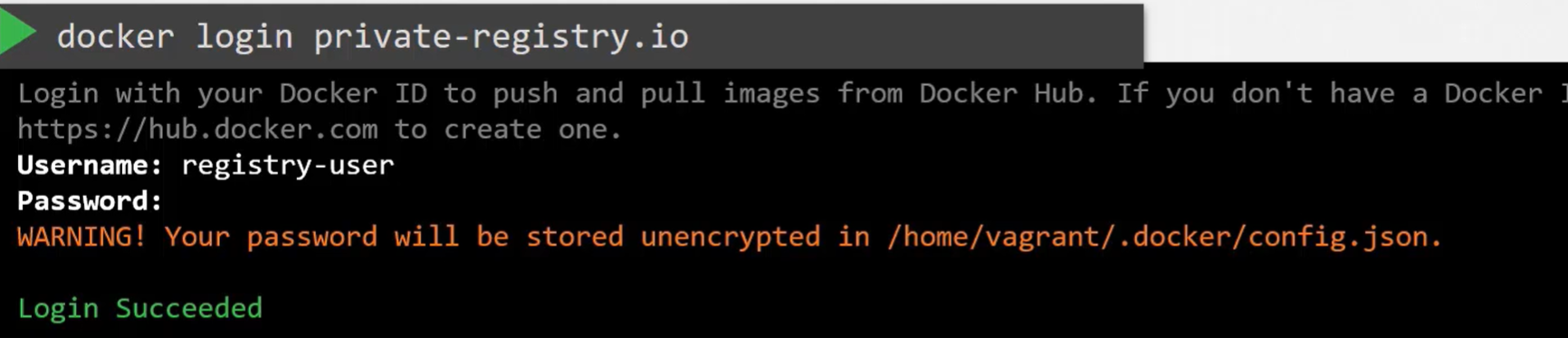
# K8S Image Security

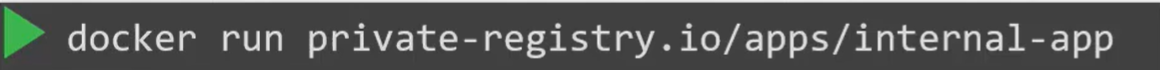


For private repository

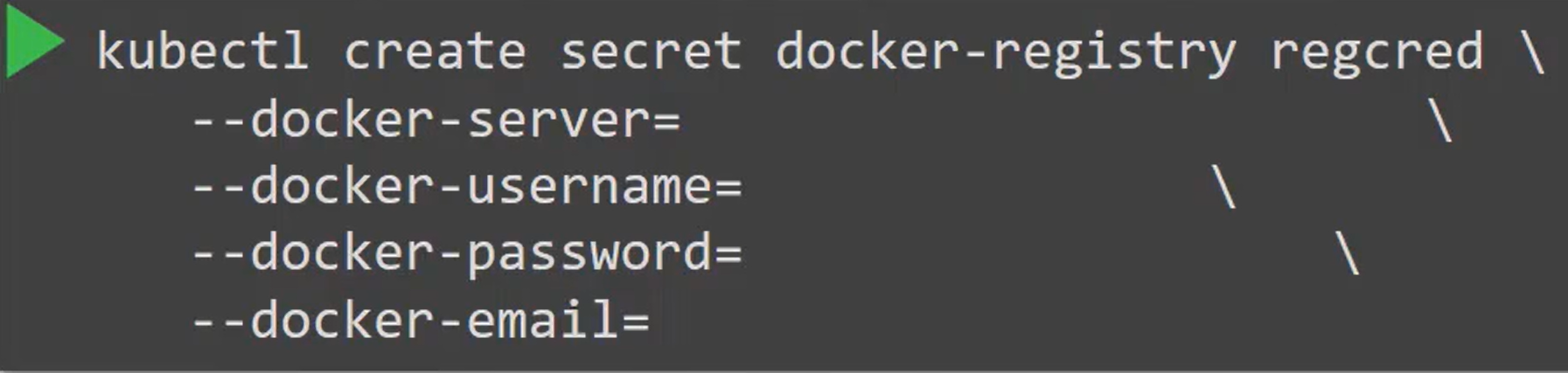
* 1. Docker Authenticate



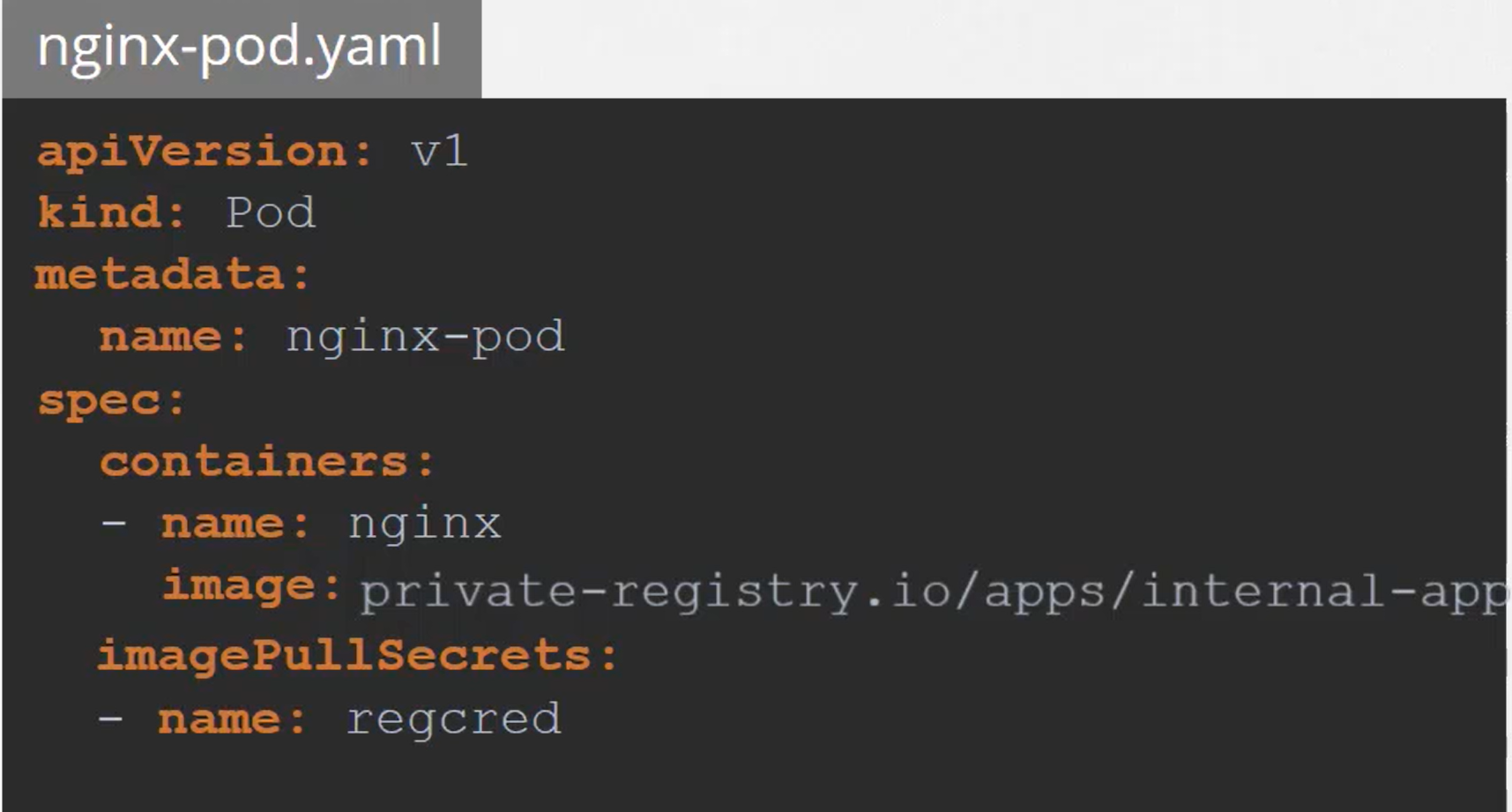
* 1. Use private repository



1. K8S POD
   1. Create Secret for the repository



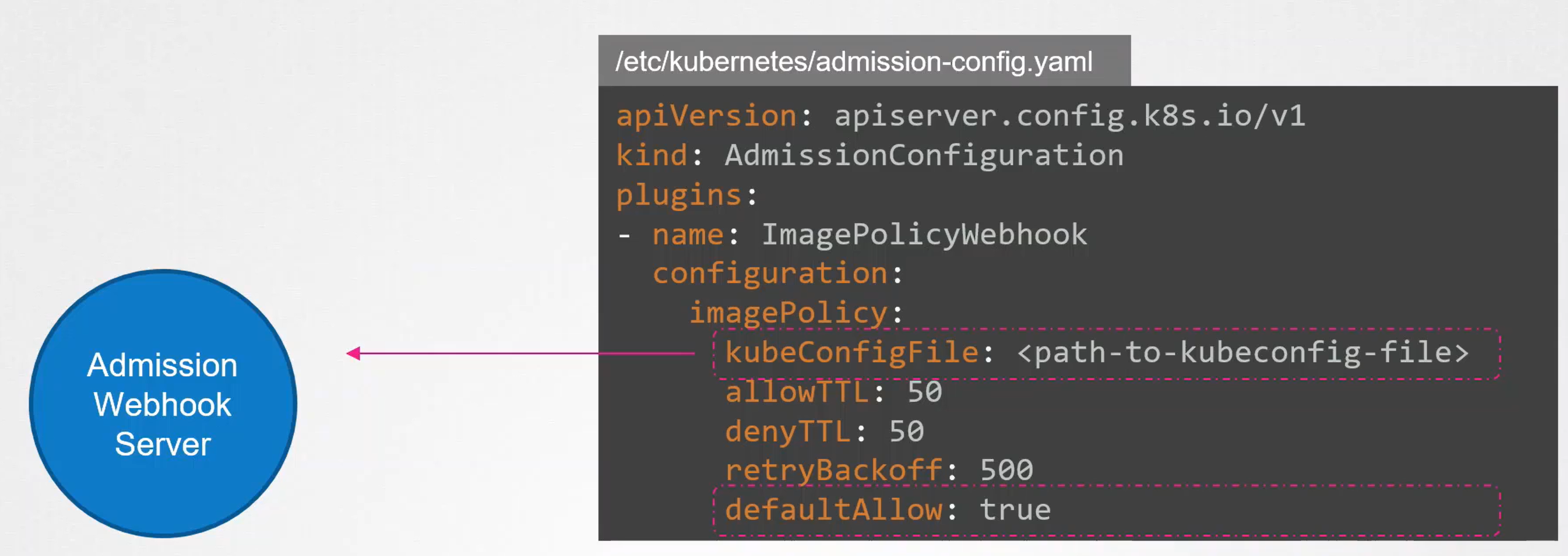
* 1. Use the secret in PODs



### Whitelist allowed registries

“Image Policy Webhook” in admission controller

1. Admission Configure File

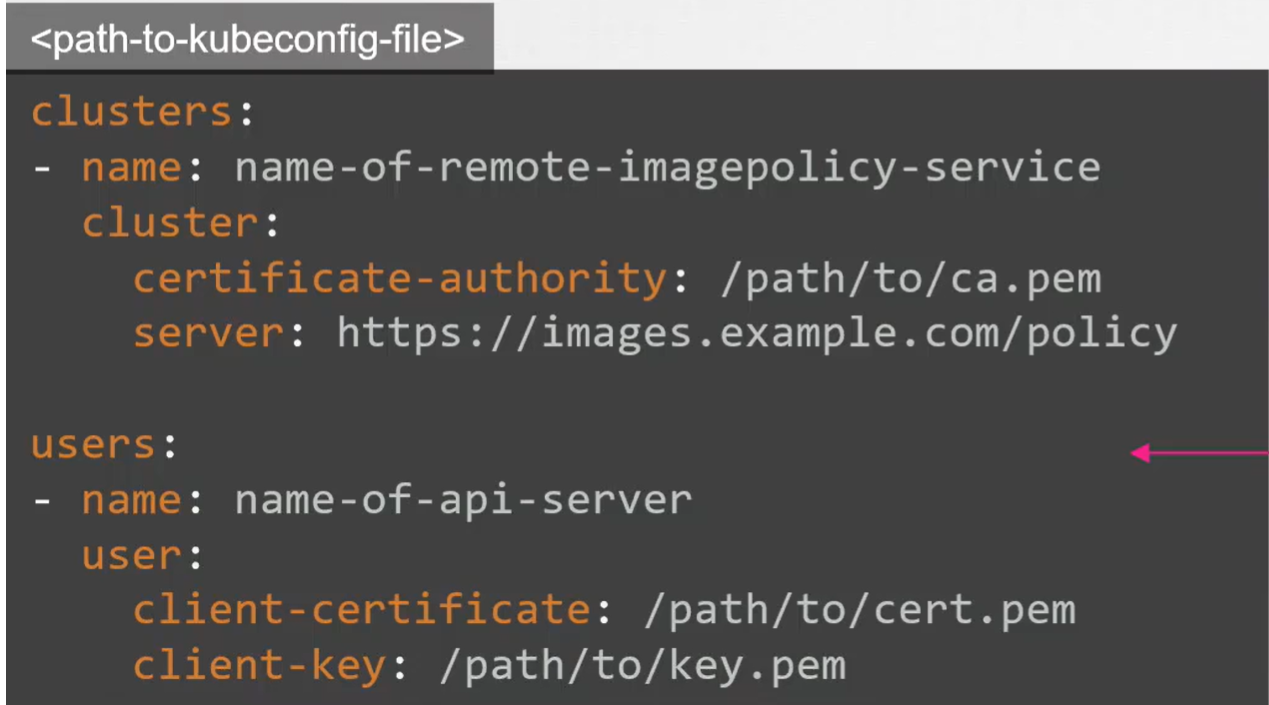


DefaultAllow property:

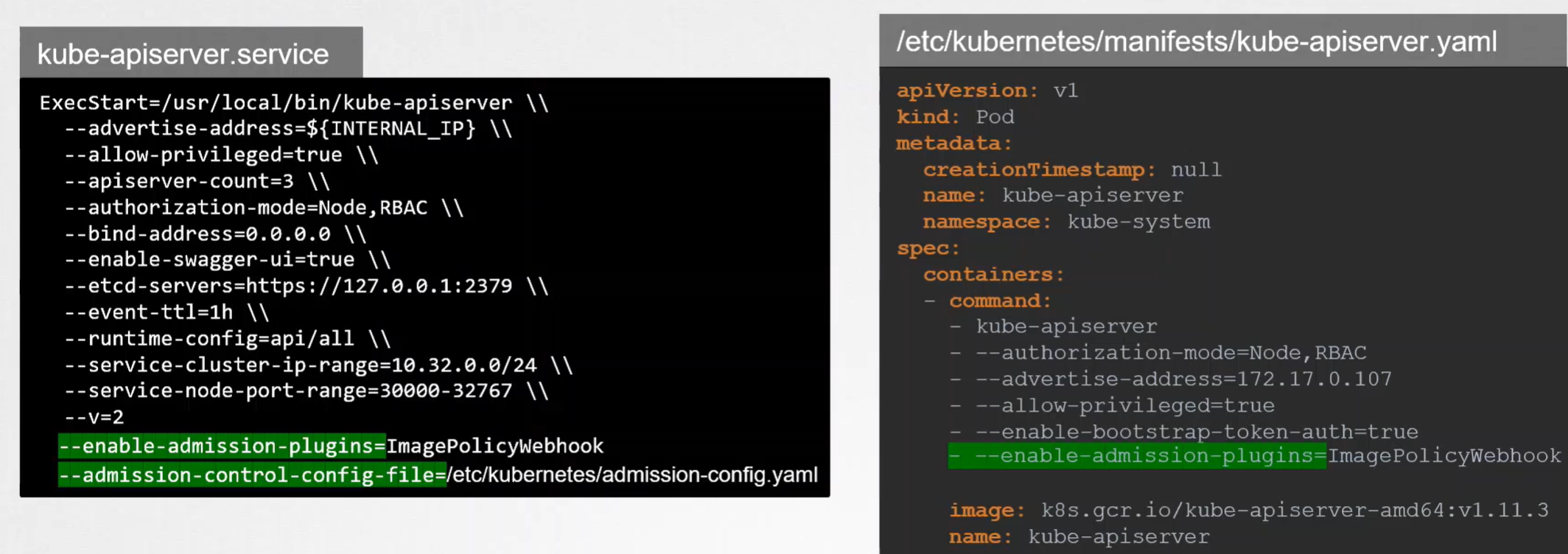
true: pod will be allowed to create even if webhook server doesn’t existed or it is not contactable

false: all requests will be rejected unless explicitly allowed by webhook server.

KubeconfigFile



1. Enable Admission Controller



Reference:

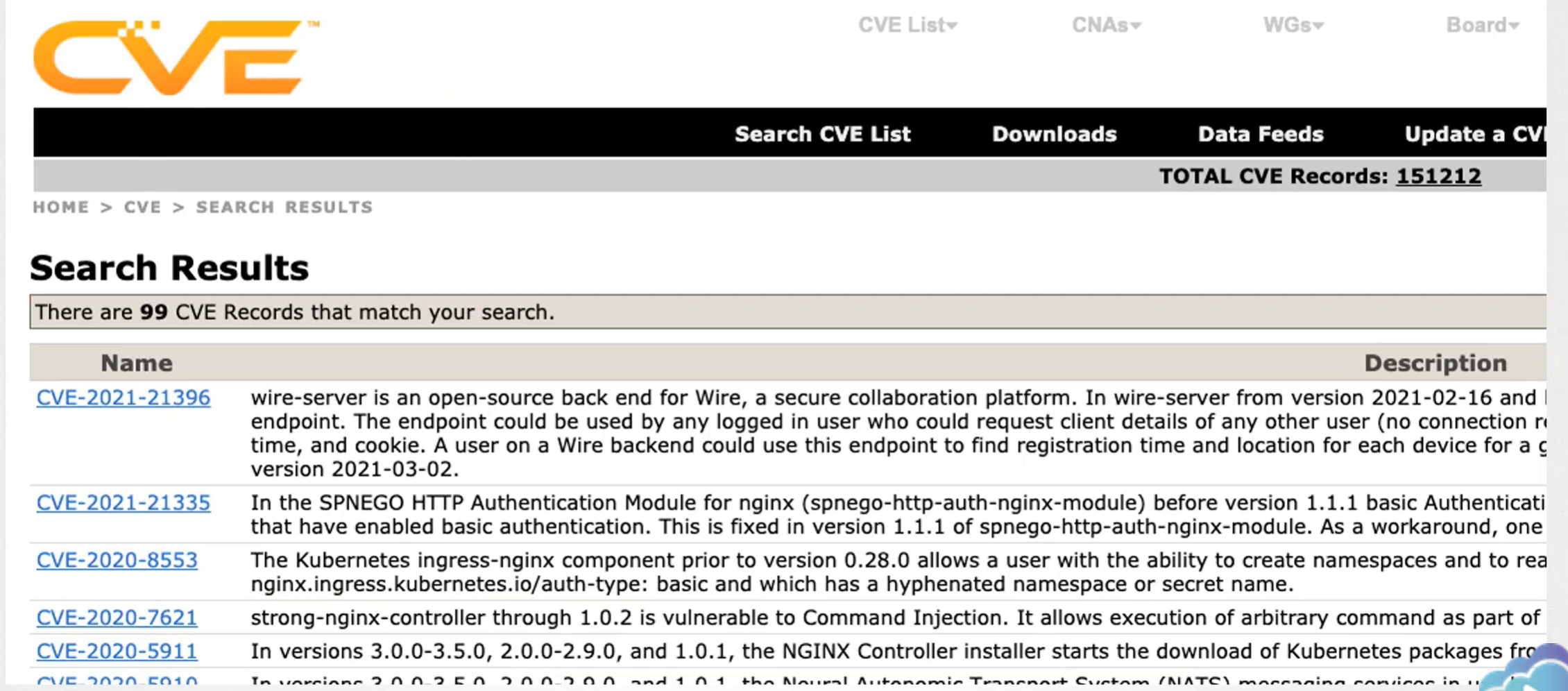
<https://github.com/kainlite/kube-image-bouncer>

### Scan images for known vulnerabilities

Keep in mind: “The lesser the number of packages there are, the more reduced the attack surface is” So, it is better to choose an image that is striped down of all unnecessary packages.

CVE: Common vulnerabilities and Exposures

Report bugs



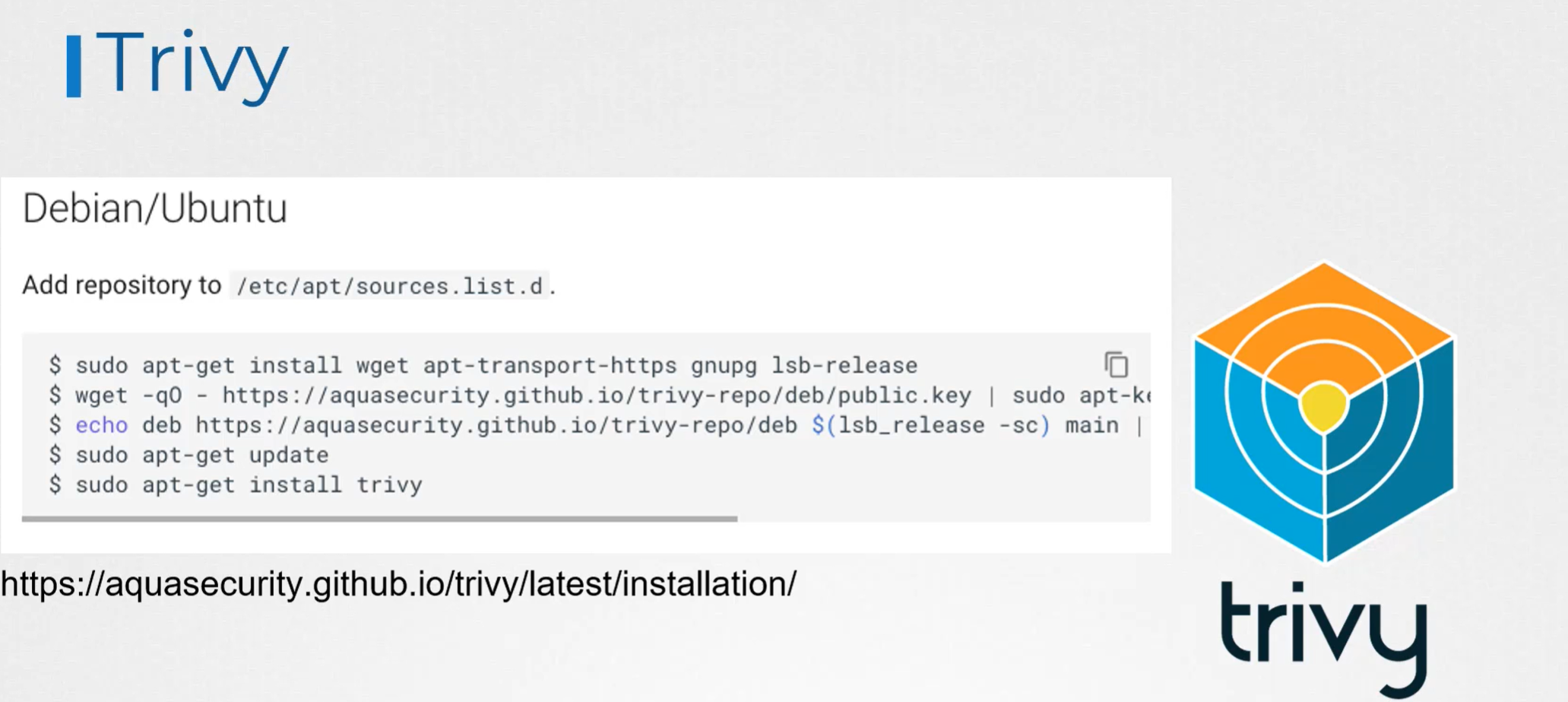
The bug with higher score, the more dangerous the it is

Cve Scanners:

Can help us know how vulnerable the packages on our system are and how vulnerable the containers are.

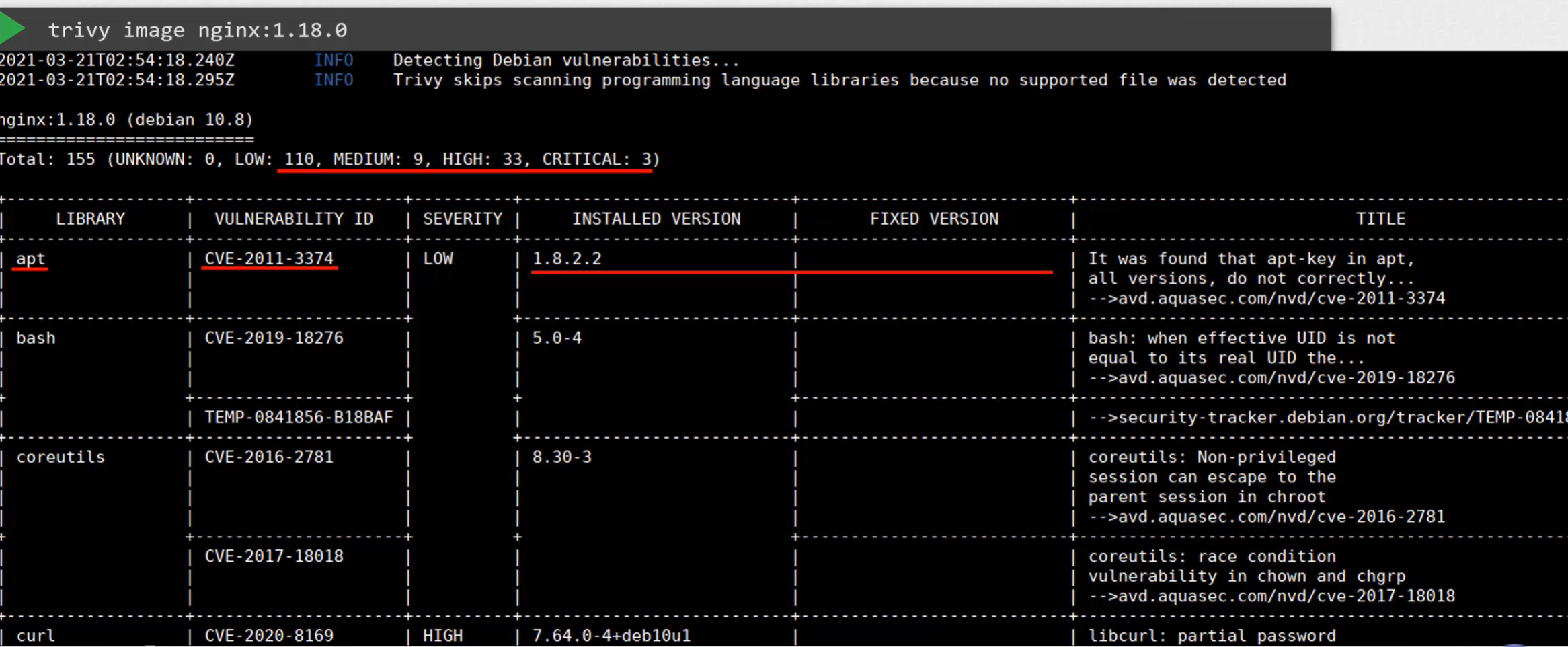
Tools:

Trivy



https://aquasecurity.github.io/trivy/v0.18.3/installation/

command





--ignore-unfixed: show the vulnerabilities we can immediately fix by upgrading vulnerable software packages.

In K8S, Admission Controller can be used to initiate scan of images every time before the pods gets deployed. But it would slow down the deploy process.

So, integrate scanning into your CI/CD pipeline. In this way, every time code is pushed and new image of application is built a scan of the image is automatically performed and vulnerabilities are reported at the source.