# How-To: Update Anchore Feeds in an Air-Gapped-Env.

And use anchor-cli to verify feeds

# 

Validating the anchor feeds and the vulnerability database in an air-gapped environment



Created by: Danial Taracks - 19 February 2021

[**How-To Update Anchore Feeds in an Air-Gapped-Env.**](#_gjdgxs) **1**

[Updating Vulnerability Feed in Airgapped Environments](#_iyv6z44ykadv) 3

[To automatically update the vulnerability database, you can:](#_7p1jkeq3qcgc) 3

[Schedule a chron job to run the script on a chosen schedule (e.g. every day):](#_d2f2rp3mhgzg) 4

[Validate the Anchore Vulnerability Feeds](#_3u2nwyv3q0b4) 5

[As root from a system that has kubectl admin access:](#_pvmjxshgw8kp) 5

[# kubectl get po -n sysdig | grep anchore](#_chrvo6m50cnf) 5

[# kubectl exec -it sysdigcloud-anchore-core-654fdd5b45-jl5sd -n sysdig -- sh](#_vgnackpcwimf) 5

[sh-4.4$ env | grep PASSWORD](#_noogd4crlc6c) 5

[sh-4.4$ anchore-cli --u admin --p N0OD83BTjD2zwZ8W system status](#_5wid18acrj8h) 5

[sh-4.4$ anchore-cli --u admin --p N0OD83BTjD2zwZ8W system feeds list](#_92k6tyk06nfv) 5

## Updating Vulnerability Feed in Airgapped Environments

NOTE: Sysdig Secure customers who install in an airgapped environment, do not have internet access to the continuous checks of vulnerability databases that are used in image scanning.

As of installer version 3.2.0-9, airgapped environments can also receive periodic vulnerability database updates.

When you install with the "airgapped\_" parameters enabled (see [Full Airgap Install](https://docs.sysdig.com/en/installer--kubernetes---openshift-.html#UUID-acab6479-7c5a-03cd-fdf2-d675cc10f72d_UUID-8748c672-f9e1-38c8-a3da-b169b39d7882) instructions), the installer will automatically push the latest vulnerability database to your environment. Follow the steps below to reinstall/refresh the vuln db, or use the script and cron job to schedule automated updates (daily, weekly, etc.).

**NOTE**: If you installed the Sysdig Platform without the "airgapped\_" parameter, you can add the following to your values.yaml and the run "installer deploy" to add the 2 additional pods “sysdigcloud-feeds-db” and “feeds-api”.

sysdig:

secure:

scanning:

feedsEnabled: true

### To automatically update the vulnerability database, you can:

1. Download the image file [quay.io/sysdig/vuln-feed-database:latest](https://quay.io/sysdig/vuln-feed-database:latest) from the Sysdig registry to the jump box server and save it locally.
2. Move the file from the jump box server to the airgapped environment (if needed)
3. Load the image file and push it to the airgapped image registry.
4. Restart the pod sysdigcloud-feeds-db
5. Restart the pod feeds-api

The following script (feeds\_database\_update.sh) performs the five steps:

#!/bin/bash

QUAY\_USERNAME="<change\_me>"

QUAY\_PASSWORD="<change\_me>"

# Download image

docker login quay.io/sysdig -u ${QUAY\_USERNAME} -p ${QUAY\_PASSWORD}

docker image pull quay.io/sysdig/vuln-feed-database:latest

# Save image

docker image save quay.io/sysdig/vuln-feed-database:latest -o vuln-feed-database.tar

# Optionally move image

mv vuln-feed-database.tar /var/shared-folder

# Load image remotely

ssh -t user@airgapped-host "docker image load -i /var/shared-folder/vuln-feed-database.tar"

# Push image remotely

ssh -t user@airgapped-host "docker tag vuln-feed-database:latest airgapped-registry/vuln-feed-database:latest"

ssh -t user@airgapped-host "docker image push airgapped-registry/vuln-feed-database:latest"

# Restart database pod

ssh -t user@airgapped-host "kubectl -n sysdigcloud scale deploy sysdigcloud-feeds-db --replicas=0"

ssh -t user@airgapped-host "kubectl -n sysdigcloud scale deploy sysdigcloud-feeds-db --replicas=1"

# Restart feeds-api pod

ssh -t user@airgapped-host "kubectl -n sysdigcloud scale deploy sysdigcloud-feeds-api --replicas=0"

ssh -t user@airgapped-host "kubectl -n sysdigcloud scale deploy sysdigcloud-feeds-api --replicas=1"

### Schedule a chron job to run the script on a chosen schedule (e.g. every day):

0 8 \* \* \* feeds-database-update.sh >/dev/null 2>&1

### 

### 

## Validate the Anchore Vulnerability Feeds

### As root from a system that has kubectl admin access:

#### # kubectl get po -n sysdig | grep anchore

**sysdigcloud-anchore-core-654fdd5b45-jl5sd** 1/1 Running 0 42h

sysdigcloud-anchore-worker-69f86f988c-mqrx8 1/1 Running 0 42h

#### # kubectl exec -it sysdigcloud-anchore-core-654fdd5b45-jl5sd -n sysdig -- sh

##### **sh-4.4$** env | grep PASSWORD

ANCHORE\_IO\_PASSWORD=

ANCHORE\_AUTH\_ENABLE\_HASHED\_PASSWORDS=false

**ANCHORE\_ADMIN\_PASSWORD=N0OD83BTjD2zwZ8W**

ANCHORE\_DB\_PASSWORD=Tu8EsonRDDLnmQm2

##### **sh-4.4$** anchore-cli --u admin --p N0OD83BTjD2zwZ8W system status

Service catalog (sysdigcloud-anchore-core-654fdd5b45-jl5sd, http://sysdigcloud-anchore-core:8082): up

Service policy\_engine (sysdigcloud-anchore-core-654fdd5b45-jl5sd, http://sysdigcloud-anchore-core:8087): up

Service analyzer (sysdigcloud-anchore-worker-69f86f988c-mqrx8, http://172.17.0.16:8084): up

Service simplequeue (sysdigcloud-anchore-core-654fdd5b45-jl5sd, http://sysdigcloud-anchore-core:8083): up

Service apiext (sysdigcloud-anchore-core-654fdd5b45-jl5sd, http://sysdigcloud-anchore-core:8228): up

Engine DB Version: 0.0.13

Engine Code Version: 0.8.1

##### **sh-4.4$** anchore-cli --u admin --p **N0OD83BTjD2zwZ8W** system feeds list

Feed Group **LastSync** RecordCount

nvdv2 nvdv2:cves 2021-02-18T11:52:49.895235 156367

vulndb vulndb:vulnerabilities pending 238000

vulnerabilities alpine:3.10 2021-02-18T12:53:03.796648 2120

vulnerabilities alpine:3.11 2021-02-18T12:53:03.365761 2318

...

vulnerabilities ubuntu:20.10 2021-02-18T12:53:06.629998 5762

sh-4.4$ exit