Installation

Für die Installation werden die Pakete von der Docker-Seite „https://download.docker.com“ für das entsprechende Betriebssystem benötigt. Im Falle eines eigenen lokalen Repository kann dann einfach installiert werden:

```
~# apt install docker-ce
```

Jetzt kann die Version von Docker angezeigt werden:

```
~:# docker --version
Docker version 29.1.2, build 890dcca
~#
```

Debmirror

Im vorliegenden Fall wurde versucht, das Docker-Repository zu spiegeln, um eine sogenannte „Air-gapped“-Umgebung zu testen, also eine Umgebung, die vom Internet „getrennt“ arbeiten kann. Dazu wurde Repository von der Seite „https://download.docker.com“ lokal mittels „`debmirror`“ gespiegelt. Leider kam es da zu dem unten angezeigten Fehler:

```
[GNUPG:] ERROR add_keyblock_resource 33554433\
[GNUPG:] NEWSIG\
[GNUPG:] ERROR keydb_search 33554445\
[GNUPG:] ERRSIG 7EA0A9C3F273FCD8 1 10 00 1764855167 9 -\
[GNUPG:] NO_PUBKEY 7EA0A9C3F273FCD8\
gpgv: unknown type of key resource 'trustedkeys.kbx'\
gpgv: Schlüsselblockhilfsmittel`/home/repo/.gnupg/trustedkeys.kbx':
Allgemeiner Fehler\
gpgv: Signatur vom Do 04 Dez 2025 14:32:47 CET\
gpgv:                mittels RSA-Schlüssel 7EA0A9C3F273FCD8\
gpgv: Signatur kann nicht geprüft werden: Kein öffentlicher Schlüssel\
.temp/.tmp/dists/trixie/Release.gpg signature does not verify.\
Failed to download some Release, Release.gpg or InRelease files!\
WARNING: releasing 1 pending lock...\
[GNUPG:] ERROR add_keyblock_resource 33554433\
[GNUPG:] NEWSIG\
[GNUPG:] ERROR keydb_search 33554445\
[GNUPG:] ERRSIG 7EA0A9C3F273FCD8 1 10 01 1764855167 9 -\
[GNUPG:] NO_PUBKEY 7EA0A9C3F273FCD8\
gpgv: unknown type of key resource 'trustedkeys.kbx'\
gpgv: Schlüsselblockhilfsmittel`/home/repo/.gnupg/trustedkeys.kbx':
Allgemeiner Fehler\
gpgv: Signatur vom Do 04 Dez 2025 14:32:47 CET\
gpgv:                mittels RSA-Schlüssel 7EA0A9C3F273FCD8\
```

```
gpgv: Signatur kann nicht geprüft werden: Kein öffentlicher Schlüssel\
.temp/.tmp/dists/trixie/InRelease signature does not verify.\
Release file does not contain Codename; using Suite (trixie).\
Errors:\
.temp/.tmp/dists/trixie/Release.gpg signature does not verify\
.temp/.tmp/dists/trixie/InRelease signature does not verify
```

Der Hintergrund ist, dass dem „`debmirror`“ die Signatur von Docker fehlt, was ihm aber beigebracht werden kann. Es muss der entsprechende Schlüssel von der Webseite heruntergeladen und importiert werden:

```
~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg && \
gpg --no-default-keyring -a --keyring /usr/share/keyrings/docker-archive-
keyring.gpg --export 7EA0A9C3F273FCD8 | \
gpg --no-default-keyring --keyring ~/.gnupg/trustedkeys.gpg --import -
```

Harbor als Proxy

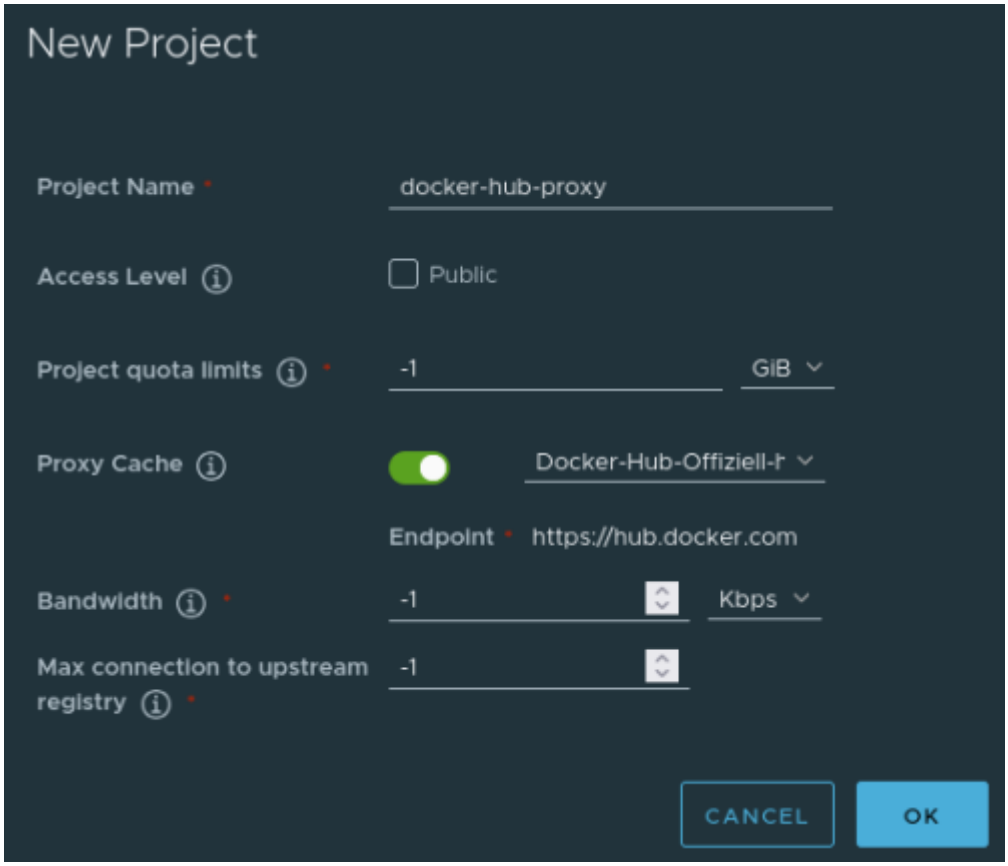
Sollte ein Harbor als Registrierung vorhanden sein (sehr hilfreich in Umgebungen ohne direkten Internetzugang), gibt es die Möglichkeit, Seiten von denen Container heruntergeladen werden (zum Beispiel Docker Hub), im Harbor zwischenzuspeichern. Damit fungiert der Harbor als Proxy. Am Beispiel von Docker Hub soll das demonstriert werden. Die Informationen wurden teilweise von der Webseite „blogs.vmware.com“ entnommen.

Voraussetzung

Es wird vorausgesetzt, dass im Harbor bereits ein Registrierung zum Docker Hub besteht, wie es unter Punkt 6 hier beschrieben steht.

Proxy-Projekt

Nach der Einrichtung der Registrierung im Harbor muss unter „`Projects`“ ein neues Projekt erstellt werden:



Wichtig ist hier die Aktivierung von „`Proxy Cache`“.

Nginx von Docker Hub

Jetzt kann Nginx von Docker Hub gezogen werden, über den Harbor als Proxy.

Direkter Aufruf ohne Harbor:

```
~# docker pull nginx:latest
```

Aufruf über den Harbor:

```
~# docker
pull
slxharbor.fritz.box /
docker-hub -
proxy /
nginx
:
latest
 <font inherit/monospace;;inherit;;#000000background-color:#ffffff;>Error
response from daemon: failed to resolve reference
"slxharbor.fritz.box/docker-hub-proxy/nginx:latest": failed to do request:
```

```
Head "https://slxharbor.fr</font>    \
itz.box/v2/docker-hub-proxy/nginx/manifests/latest": tls: failed to verify
certificate: x509: certificate signed by unknown authority \
~#
```

Zertifikatsproblem

Dem Harbor-Zertifikat wird nicht vertraut, weil es nicht bekannt ist. Das liegt daran, dass es ein selbst erstelltes Zertfikat ist. Damit Docker dem Zertifikat vertraut, muss es in das Verzeichnis „/usr/local/share/ca-certificates/“ kopiert und dem Zertifikatsspeicher hinzugefügt werden.

Schritt 1: Kopieren der CA:

```
~# cp -v ca.crt /usr/local/share/ca-certificates/
'ca.crt' -> '/usr/local/share/ca-certificates/ca.crt'
~#
```

Schritt 2: Kopieren des Server-Zertifikates:

```
~# cp -v slxharbor.fritz.box.crt /usr/local/share/ca-certificates/
'slxharbor.fritz.box.crt' -> '/usr/local/share/ca-
certificates/slxharbor.fritz.box.crt'
~#
```

Schritt 3: Hinzufügen des Zertifikates zum Zertifikatsspeicher:

```
~# update-ca-certificates --fresh
Clearing symlinks in /etc/ssl/certs...
done.
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt, it does not contain exactly
one certificate or CRL
rehash: warning: skipping duplicate certificate in slxharbor.fritz.box.pem
151 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
~#
```

Schritt 4: Docker-Dienst neu starten:

```
~# systemctl restart docker
~#
```

Erneuter Pull-Versuch:

```
~# docker pull slxharbor.fritz.box/docker-hub-proxy/nginx:latest
Error response from daemon: unknown: failed to resolve reference
"slxharbor.fritz.box/docker-hub-proxy/nginx:latest": unexpected status from
HEAD request to
https://slxharbor.fritz.box/v2/docker-hub-proxy/nginx/manifests/latest: 401
```

```
Unauthorized
~#
```

Authorisierung

Damit Daten vom Harbor geholt werden können, muss eine (einmalige) Authorisierung erfolgen:

```
~# docker login slxharbor.fritz.box --username sborne\
Password:

WARNING! Your credentials are stored unencrypted in
'/root/.docker/config.json'.\
Configure a credential helper to remove this warning. See\
https://docs.docker.com/go/credential-store/

Login Succeeded

~#
```

Nächster Pull-Versuch:

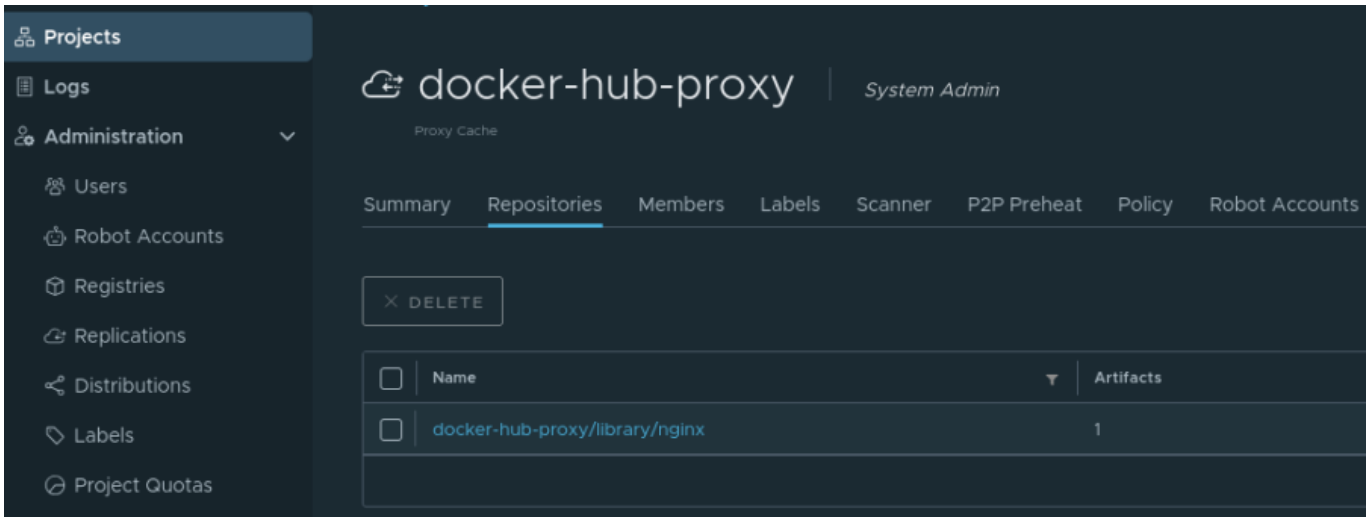
```
~# docker pull slxharbor.fritz.box/docker-hub-proxy/nginx:latest\
latest: Pulling from docker-hub-proxy/nginx\
5b5fa0b64d74: Pull complete \
1733a4cd5954: Pull complete \
5b219a92f92a: Pull complete \
ee3a09d2248a: Pull complete \
7382b41547b8: Pull complete \
9ee60c6c0558: Pull complete \
114e699da838: Pull complete \
adeb5aba46ee: Download complete \
11488ed04caf: Download complete \
Digest:
sha256:fb01117203ff38c2f9af91db1a7409459182a37c87cced5cb442d1d8fcc66d19\
Status: Downloaded newer image for slxharbor.fritz.box/docker-hub-
proxy/nginx:latest\
slxharbor.fritz.box/docker-hub-proxy/nginx:latest

~#
```

Geschafft!

Kontrolle im Harbor

Das gezogene Nginx ist jetzt auch im Harbor zu sehen:



.Ende des Dokuments