

OFFIZIELLE DOCKER IMAGES

EIN ERFAHRUNGSBERICHT

Philipp Krenn

@xeraa



elastic

Infrastructure | Developer 🥑

Who uses
DOCKER?

Who uses

DOCKER IN PRODUCTION?

Who uses

STATEFUL DOCKER IMAGES?

Who uses
OUR IMAGES?

Who uses our stack with
OTHER IMAGES?

Docker: the world's most heavily funded college project

Internal quote from Slack



CONTENT

"Official"

Base Images

Release Policy

Security & Stability

Customization

Orchestration

DOCKER HUB

"OFFICIAL"

WHAT DO YOU GET?

```
docker pull elasticsearch:X
```

Same for Kibana and Logstash

IT'S COMPLICATED

Docker Inc

Deprecated

Replicated

Full Description

Supported tags and respective **Dockerfile** links

- `6.4.2` ([6/Dockerfile](#))
- `5.6.12` , `5.6` , `5` ([5/Dockerfile](#))
- `5.6.12-alpine` , `5.6-alpine` , `5-alpine` ([5/alpine/Dockerfile](#))



jethr0null commented on 5 Oct

Contributor



In the interest of providing the best and most reliable support for the official Elasticsearch Docker images, Elastic performs a rigorous set of automated and manual tests as part of the Elasticsearch release process. Beyond testing our own software, these tests have historically uncovered other items such as JVM issues and OS-dependent issues that the Elasticsearch software relied on.

In order to provide the Docker Library community with the same level of support for any Elastic images which are pulled from Docker Hub, it is necessary that these images are equivalent with those which come from the Elastic build system. This equivalence allows us to address issues in a specific version of Elasticsearch with a full understanding of exactly what exists in that version, allowing us to support and troubleshoot any issues effectively and efficiently. It also allows Elastic to respond to issues with any necessary updates rapidly, and without the risk of divergent image content.

After extensive discussions and collaboration with Docker, it has been determined that the easiest, fastest, and most stable way to accomplish this is to use the Elastic images directly rather than rely on setting up an exact replica of Elastic's build and testing systems. The code contained within this image is open (references to the code and the referenced image can be viewed within the image Dockerfile).

14 lines (8 sloc) | 616 Bytes

Raw

Blame

History



```
1  # Elasticsearch 6.4.3
2
3  # This image re-bundles the Docker image from the upstream provider, Elastic.
4  FROM docker.elastic.co/elasticsearch/elasticsearch:6.4.3@sha256:ba2d82b70025e503ded44135e2fbb79b0aefa9db17be2750d75565c28d21ea
5
6  # The upstream image was built by:
7  #   https://github.com/elastic/elasticsearch-docker/tree/6.4.3
8
9  # For a full list of supported images and tags visit https://www.docker.elastic.co
10
11 # For Elasticsearch documentation visit https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html
12
13 # See https://github.com/docker-library/official-images/pull/4916 for more details.
```

CUSTOM REGISTRY
docker.elastic.co

I'm surprised more people don't just host their own container registries since doing that is faster than every cloud offering and docker hub

<https://twitter.com/jessfraz/status/978449365261082625>

OUR MOTIVATION

Download statistics

Speed & reliability

Docker @ Elastic

At Elastic, we care about Docker. We provide Docker images for all the products in our stack, and we consider them a first-class distribution format. We also host a dedicated Docker Registry to provide the best possible experience and the most reliable service for you.

On this page, you'll find all the resources — **docker** commands, links to product release notes, documentation and source code — for installing and using the Docker images.

Watch a [presentation](#) from our developers at the Elastic{ON} conference to get details about the background behind our Docker efforts and to see a live demo for the entire stack.



Elasticsearch

6.4 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.4.3` >_ 🔗 📄

6.3 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.3.2` >_ 🔗 📄

6.2 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.2.4` >_ 🔗 📄

6.1 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.1.4` >_ 🔗 📄

6.0 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.0.1` >_ 🔗 📄

5.6 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:5.6.13` >_ 🔗 📄

5.5 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:5.5.3` >_ 🔗 📄

Docker @ Elastic

At Elastic, we care about Docker. We provide Docker images for all the products in our stack, and we consider them a first-class distribution format. We also host a dedicated Docker Registry to provide the best possible experience and the most reliable service for you.

On this page, you'll find all the resources — **docker** commands, links to product release notes, documentation and source code — for installing and using the Docker images.

Watch a [presentation](#) from our developers at the Elastic{ON} conference to get details about the background behind our Docker efforts and to see a live demo for the entire stack.

<https://www.docker.elastic.co>



Elasticsearch 

6.4 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.4.3` >_ 🔗 📄

6.3 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.3.2` >_ 🔗 📄

6.2 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.2.4` >_ 🔗 📄

6.1 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.1.4` >_ 🔗 📄

6.0 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:6.0.1` >_ 🔗 📄

5.6 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:5.6.13` >_ 🔗 📄

5.5 > `docker pull docker.elastic.co/elasticsearch/elasticsearch:5.5.3` >_ 🔗 📄

PROBLEMS

Some broken tooling like
automated builds, Kitematic,...

China




IPv6

Repos



elastic
Elastic

<https://www.elastic.co>
Joined September 2014

	elastic/elasticsearch public	11 STARS	100K+ PULLS	DETAILS
	elastic/kibana public	9 STARS	100K+ PULLS	DETAILS
	elastic/logstash public	7 STARS	50K+ PULLS	DETAILS
	elastic/apm-server public	4 STARS	50K+ PULLS	DETAILS
	elastic/auditbeat public	2 STARS	10K+ PULLS	DETAILS
	elastic/filebeat public	8	10K+	DETAILS

Search

DashboardExploreOrganizationsCreate

xeraa

Repos

elastic

Elastic

https://www.elastic.co

Joined September 2014

elastic/elasticsearch

public

11 STARS

100K+ PULLS

DETAILS

elastic/kibana

public

9 STARS

100K+ PULLS

DETAILS

elastic/logstash

public

7 STARS

50K+ PULLS

DETAILS

elastic/apm-server

public

4 STARS

50K+ PULLS

DETAILS

elastic/auditbeat

public

2 STARS

10K+ PULLS

DETAILS

elastic/filebeat

public

8 STARS

10K+ PULLS

DETAILS

https://hub.docker.com/r/elastic/elasticsearch/

kibana/

*beat/

logstash/

apm-server/

BASE IMAGES

Elasticsearch

ALPINE

Kibana Beats Logstash

UBUNTU

Common base image in 5.4+

CENTOS 7

<https://github.com/elastic/elasticsearch-docker/blob/master/templates/Dockerfile.j2>

```
FROM centos:7 AS prep_es_files
```

```
ENV PATH /usr/share/elasticsearch/bin:$PATH
```

```
RUN curl -s https://download.java.net/java/GA/jdk11/13/GPL/openjdk-11.0.1_linux-x64_bin.tar.gz | \
    tar -C /opt -zxf -
```

```
ENV JAVA_HOME /opt/jdk-11.0.1
```

```
RUN groupadd -g 1000 elasticsearch && \
    adduser -u 1000 -g 1000 -d /usr/share/elasticsearch elasticsearch
```

Upside

SIMILAR SETUP

SHARED LAYERS

JVM LARGE ANYWAY



jarpy commented on 31 Mar • edited

Owner



Actually, my money is on the [native call to getmntent\(3\)](#) .

Oh look! What's the chances we don't have this patch in our musl libc?

<https://git.musl-libc.org/cgiit/musl/commit/src/misc/mntent.c?id=05973dc3bbc1aca9b3c8347de6879ed72147ab3b>



 jarpy added the **bug** label on 31 Mar



jarpy commented on 31 Mar

Owner



And I'm a bit late in saying this, but thank you for the excellent reproduction steps. They were extremely useful once we (kinda by accident) found the right environment to run them in.

Downside

SIZE

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	SIZE
docker.elastic.co/kibana/kibana	6.5.0	fcc1f039f61c	727MB
docker.elastic.co/elasticsearch/elasticsearch	6.5.0	ff171d17e77c	774MB
docker.elastic.co/beats/filebeat	6.5.0	aee067f4a241	299MB
docker.elastic.co/kibana/kibana	6.2.4	327c6538ba4c	933MB
docker.elastic.co/elasticsearch/elasticsearch	6.2.4	7cb69da7148d	515MB
docker.elastic.co/beats/filebeat	6.2.4	26a00abcde82	319MB
docker.elastic.co/kibana/kibana	5.6.13	59fcc69d2cc6	653MB
docker.elastic.co/elasticsearch/elasticsearch	5.6.13	21673573a265	525MB
docker.elastic.co/beats/filebeat	5.6.13	2aec30f6b3fc	284MB
docker.elastic.co/kibana/kibana	5.3.3	ffe778f7e489	679MB
docker.elastic.co/elasticsearch/elasticsearch	5.3.3	5857f98b5920	165MB
docker.elastic.co/beats/filebeat	5.3.3	c01be8a8f630	232MB

	5.3	5.6	6.2	6.5
ES	165MB	525MB	515MB	774MB
Kibana	679MB	653MB	933MB	727MB
Filebeat	232MB	284MB	319MB	299MB

Does it matter?

STATEFUL VS STATELESS

What to
INCLUDE?

Single image for 5.x

PLATINUM TRIAL

Three flavors 6.0 to 6.2

BASIC^{*}, OSS, PLATINUM TRIAL

^{*} Default



Two flavors 6.3+

BASIC / PLATINUM TRIAL^{*}, OSS

^{*} Default



FUTURE

Multiple BASE IMAGES?

WINDOWS?!

RELEASE POLICY

No :latest



ZOMBIES

ideas that should have been killed by
evidence, but keep shambling along

6 AND 6 . 4?

WHAT'S IN A TAG?

<code>docker.elastic.co/elasticsearch/elasticsearch</code>	5.3.3	5857f98b5920	4 months ago
<code>docker.elastic.co/beats/filebeat</code>	5.3.3	c01be8a8f630	5 months ago
<code>docker.elastic.co/kibana/kibana</code>	5.3.3	ffe778f7e489	5 months ago

Currently OVERWRITE TAG

LABEL SCHEMA

```
LABEL org.label-schema.schema-version="1.0" \
  org.label-schema.vendor="Elastic" \
  org.label-schema.name="elasticsearch" \
  org.label-schema.version="{{ elastic_version }}" \
  org.label-schema.url="https://www.elastic.co/products/elasticsearch" \
  org.label-schema.vcs-url="https://github.com/elastic/elasticsearch-docker" \
{% if image_flavor == 'oss' -%}
  license="Apache-2.0"
{% else -%}
  license="Elastic License"
{% endif -%}
```

Base image & JVM direct dependencies

Future

ADD IMAGE VERSION?

SECURITY & STABILITY

Run Elasticsearch as
root

A close-up photograph of numerous cockroaches, likely German cockroaches, crawling on a light-colored, textured piece of wood. The cockroaches are dark brown with lighter, yellowish-brown stripes on their abdomens. They are scattered across the wood, some facing towards the camera and others away. The background is slightly blurred, showing more of the wood and some green foliage in the upper right corner.

COCKROACHES

claims that disappear for a while when
proved wrong, but just keep on coming
back

MODE

Production: Clusterable

Development: Local network

"Docker" mode

`discovery.type=single-node`

BOOTSTRAP CHECKS
are here to stay

6.0+ no more

DEFAULT CREDENTIALS

6.x Security non-trial requires

CERTIFICATES

The container runs Elasticsearch as
user `elasticsearch` using `uid:gid`
`1000:1000`.

[https://www.elastic.co/guide/en/elasticsearch/reference/current/
docker.html](https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html)



bluepuma77 commented on 4 Mar



Bug description

Starting elasticsearch:5.2.2 results in `Failed to created node environment`. This has been covered in [#21](#), the recommendation was to set user permissions on the host system.

```
docker run --rm -v /tmp/elastic:/usr/share/elasticsearch/data docker.elastic.co/elasticsearch:  
# breaks
```

I prefer simple fire & forget docker containers which don't need any preparation on the host system. Would it be an option to check the Elasticsearch image from Docker, that runs fine, without setting any permissions before, and replicate that behaviour in new `elasticsearch-docker`?

```
docker run --rm -v /tmp/elastic:/usr/share/elasticsearch/data elasticsearch:5.2.2  
# runs
```

```
docker run --rm -v /tmp/elastic:/usr/share/elasticsearch  
# breaks
```

I prefer simple fire & forget docker containers which don't need any permissions before, and replicate that behaviour in new e

```
docker run --rm -v /tmp/elastic:/usr/share/elasticsearch  
# runs
```



DON'T MUTATE
the bind mounted local directory

THOSE WHO DO NOT UNDERSTAND
UNIX ARE CONDEMNED TO REINVENT
IT, POORLY.

— Henry Spencer

Docker default value?

LimitNOFILE & LimitNPROC

infinity

<https://github.com/moby/moby/commit/8db61095a3d0bcb0733580734ba5d54bc27a614d> (July 2016)

TEST

```
$ docker run --rm centos:7 /bin/bash -c \  
    'ulimit -Hn && ulimit -Sn && ulimit -Hu && ulimit -Su'  
1048576  
1048576  
unlimited  
unlimited
```


Limit for virtual memory?

`vm.max_map_count`

TEST

```
$ docker run --rm centos:7 /bin/bash -c \  
  'sysctl vm.max_map_count'  
vm.max_map_count = 262144
```

Must be set on the host

Combine two new systems to get

CHAOS & DESPAIR

Docker is a
LEAKY ABSTRACTION



jarpy commented on 31 Mar

Owner



This reproduces for me on Docker for Windows version 17.03.0-ce-win1, (run from Powershell).

It does not reproduce on Docker for Linux version 17.03.0-ce.

Interesting. I think the Docker VM, as used by OSX and Windows is introducing the problem.

CUSTOMIZATION

FEATURE REQUEST

```
if [ -f /custom/user_init.sh ]  
then  
  . /custom/user_init.sh  
fi
```

Or customization through environment variables

No

RUNTIME MUTATION

DOCKERFILE

```
ARG ELASTIC_VERSION
FROM docker.elastic.co/elasticsearch/elasticsearch:${ELASTIC_VERSION}

RUN bin/elasticsearch-plugin install analysis-phonetic --batch

ARG ELASTIC_VERSION
RUN bin/elasticsearch-plugin install \
    https://github.com/spinscale/elasticsearch-ingest-langdetect/releases/download/
    ${ELASTIC_VERSION}.1/ingest-langdetect-${ELASTIC_VERSION}.1.zip --batch
```

GENERATE KEYSTORE

```
$ docker run -p 9200:9200 -p 9300:9300 -e "discovery.type=single-node" \
-v /Users/philipp/Desktop/demo/config:/usr/share/elasticsearch/config/ \
-it docker.elastic.co/elasticsearch/elasticsearch:6.4.3 /bin/bash
[root@1006ed50b646 elasticsearch]# ./bin/elasticsearch-keystore create
Created elasticsearch keystore in /usr/share/elasticsearch/config
[root@1006ed50b646 elasticsearch]# ./bin/elasticsearch-keystore add test
Enter value for test:
[root@1006ed50b646 elasticsearch]# exit
exit
$ cat config/elasticsearch.keystore
??lelasticsearch.keystore?@g?o!?$?K?Lf?w?VAEŠHm?[?a6?B??? y?,!B}??H?æ?AU=?C?:?o?
?W?08?}U?;p?т???cQ????7?JY?      2A?:???ZUY??2V?9?9??(??0?q\
```

MOUNT KEYSTORE (DOCKER COMPOSE)

elasticsearch:

image: `docker.elastic.co/elasticsearch/elasticsearch:${ELASTIC_VERSION}`

secrets:

- source:** `elasticsearch.keystore`

- target:** `/usr/share/elasticsearch/config/elasticsearch.keystore`

ORCHESTRATION

Who uses KUBERNETES?

Who uses
SWARM?

Who uses
MESOS?

Who uses
NOMAD?

NO ORCHESTRATION
yet

Kubernetes 1.8 allows dots in env vars

<https://github.com/kubernetes/kubernetes/issues/2707>



Pires

@pires_oss

Follow



today, i decided to end my commitment to maintaining github.com/pires/kubernetes it's been 3 years since i last used Elasticsearch and other projects need all the attention i can give them. it was a great run, thank you ALL.



pires/kubernetes-elasticsearch-cluster

Elasticsearch cluster on top of Kubernetes made easy. -
pires/kubernetes-elasticsearch-cluster

github.com

11:10 AM - 7 Nov 2018

You know, for Kubernetes Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor Apache-2.0

Branch: master New pull request Create new file Upload files Find file Clone or download

Crazybus first commit		Latest commit 0f99a55 2 days ago	
elasticsearch	first commit	2 days ago	
helpers	first commit	2 days ago	
kibana	first commit	2 days ago	
.gitignore	first commit	2 days ago	
LICENSE	first commit	2 days ago	
README.md	first commit	2 days ago	
requirements.txt	first commit	2 days ago	

README.md

Elastic Stack Kubernetes Helm Charts

Kubernetes has made huge improvements in the ability to run stateful workloads including databases and message queues, but I still prefer not to run them on Kubernetes.

<https://twitter.com/kelseyhightower/status/963413508300812295>



elastic cloud
enterprise

CONCLUSION

"DOCKER IS DISRUPTING THE
INDUSTRY"

"CAN I RUN ELASTICSEARCH ON
DOCKER?"

"SHOULD I RUN ELASTICSEARCH ON
DOCKER?"

Even when stateful services do the right things managing state is still hard. Mixing stateful and stateless applications on the same cluster elevates the complexity of the entire cluster. Cluster security and upgrades become much harder.

<https://twitter.com/kelseyhightower/status/963417215608369153>

QUICK QUIZ

What do you get?

```
docker pull elastic/elasticsearch
```

```
$ docker pull elastic/elasticsearch
```

```
Using default tag: latest
```

```
Error response from daemon: manifest  
for elastic/elasticsearch:latest not found
```

```
docker pull logstash:alpine
```

`docker pull metricbeat:6.4.3`


```
$ docker pull metricbeat:6.4.3
```

```
Error response from daemon: pull access denied for metricbeat,  
repository does not exist or may require 'docker login'
```

```
docker pull elastic/metricbeat:6.4.3
```

```
docker pull docker.elastic.co/apm/apm-server:6.4.3
```

"AGGREGIERTE LOGGING-PATTERNS"

Tomorrow 15:45 @ Gustav Mahler II

QUESTIONS & DISCUSSION

Philipp Krenn

@xeraa