Có hai phương pháp để cài đặt Docker trên CentOS 7. Một phương pháp liên quan đến việc cài đặt nó trên một bản cài đặt hiện có của hệ điều hành. Cách khác liên quan đến việc quay một server bằng một công cụ có tên [Docker Machine](https://galaxyz.net/cach-cung-cap-va-quan-ly-may-chu-docker-tu-xa-bang-may-docker-tren-centos-7.1178.anews) tự động cài đặt Docker trên đó.

Trong hướng dẫn này, bạn sẽ học cách cài đặt và sử dụng nó trên bản cài đặt hiện có của CentOS 7.

Yêu cầu

* 64-bit CentOS 7 Server
* User không phải root có quyền sudo. Server CentOS 7 được cài đặt bằng [Hướng dẫn Cài đặt Ban đầu cho CentOS 7](https://galaxyz.net/thiet-lap-may-chu-ban-dau-voi-centos-7.374.anews) giải thích cách cài đặt điều này.

**Lưu ý:** Docker yêu cầu version 64-bit của CentOS 7 cũng như version kernel bằng hoặc lớn hơn 3.10. CentOS 7 Server 64-bit mặc định đáp ứng các yêu cầu này.

Tất cả các lệnh trong hướng dẫn này phải được chạy với quyền user không phải root. Nếu cần có quyền truy cập root cho lệnh, nó sẽ được đặt trước sudo . [Hướng dẫn cài đặt ban đầu cho CentOS 7](https://galaxyz.net/thiet-lap-may-chu-ban-dau-voi-centos-7.374.anews) giải thích cách thêm user và cấp cho họ quyền truy cập sudo.

Bước 1 - Cài đặt Docker

Gói cài đặt Docker có sẵn trong repository lưu trữ CentOS 7 chính thức có thể không phải là version mới nhất. Để có được version mới nhất và tốt nhất, hãy cài đặt Docker từ repository Docker chính thức. Phần này chỉ cho bạn cách thực hiện điều đó.

Nhưng trước tiên, hãy cập nhật database gói:

* sudo yum check-update

Bây giờ chạy lệnh này. Nó sẽ thêm repository Docker chính thức, download version Docker mới nhất và cài đặt nó:

* curl -fsSL https://get.docker.com/ | sh

Sau khi cài đặt xong, hãy khởi động trình Docker:

* sudo systemctl start docker

Xác minh nó đang chạy:

* sudo systemctl status docker

Đầu ra phải giống như sau , cho thấy rằng dịch vụ đang đang chạy :

Output

● docker.service - Docker Application Container Engine Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled) Active: active (running) since Sun 2016-05-01 06:53:52 CDT; 1 weeks 3 days ago Docs: https://docs.docker.com Main PID: 749 (docker)

Cuối cùng, hãy đảm bảo nó bắt đầu ở mỗi lần khởi động lại server :

* sudo systemctl enable docker

Việc cài đặt Docker giờ đây không chỉ cung cấp cho bạn dịch vụ Docker (daemon) mà còn cả tiện ích dòng lệnh docker hoặc ứng dụng client Docker. Ta sẽ khám phá cách sử dụng lệnh docker ở phần sau trong hướng dẫn này.

Bước 2 - Thực thi lệnh Docker mà không cần Sudo (Tùy chọn)

Theo mặc định, việc chạy lệnh docker yêu cầu quyền root - nghĩa là bạn phải đặt tiền tố lệnh bằng sudo . Nó cũng có thể được chạy bởi user trong group **docker** , được tạo tự động trong quá trình cài đặt Docker. Nếu bạn cố gắng chạy docker lệnh mà không đặt trước từ đó sudo hoặc không là ở group Docker, bạn sẽ nhận được một kết quả như thế này:

Output

docker: Cannot connect to the Docker daemon. Is the docker daemon running on this host?. See 'docker run --help'.

Nếu bạn muốn tránh nhập sudo khi nào bạn chạy lệnh docker , hãy thêm tên user của bạn vào group docker:

* sudo usermod -aG docker $(whoami)

Bạn cần đăng xuất khỏi Server và đăng nhập lại với quyền là user cũ để thực hiện thay đổi này.

Nếu bạn cần thêm user vào group docker mà bạn chưa đăng nhập, hãy khai báo tên user đó một cách rõ ràng bằng cách sử dụng:

* sudo usermod -aG docker username

Phần còn lại của bài viết này giả định bạn đang chạy lệnh docker với quyền là user trong group user docker. Nếu bạn chọn không, vui lòng thêm các lệnh với sudo .

Bước 3 - Sử dụng lệnh Docker

Với Docker được cài đặt và hoạt động, bây giờ là lúc để làm quen với tiện ích dòng lệnh. Sử dụng docker bao gồm việc chuyển cho nó một chuỗi các tùy chọn và lệnh con theo sau là các đối số. Cú pháp có dạng sau:

* docker [option] [command] [arguments]

Để xem tất cả các lệnh con có sẵn, hãy nhập:

* docker

Kể từ Docker 1.11.1, danh sách đầy đủ các lệnh con có sẵn bao gồm:

Output

attach Attach to a running container build Build an image from a Dockerfile commit Create a new image from a container's changes cp Copy files/folders between a container and the local filesystem create Create a new container diff Inspect changes on a container's filesystem events Get real time events from the server exec Run a command in a running container export Export a container's filesystem as a tar archive history Show the history of an image images List images import Import the contents from a tarball to create a filesystem image info Display system-wide information inspect Return low-level information on a container or image kill Kill a running container load Load an image from a tar archive or STDIN login Log in to a Docker registry logout Log out from a Docker registry logs Fetch the logs of a container network Manage Docker networks pause Pause all processes within a container port List port mappings or a specific mapping for the CONTAINER ps List containers pull Pull an image or a repository from a registry push Push an image or a repository to a registry rename Rename a container restart Restart a container rm Remove one or more containers rmi Remove one or more images run Run a command in a new container save Save one or more images to a tar archive search Search the Docker Hub for images start Start one or more stopped containers stats Display a live stream of container(s) resource usage statistics stop Stop a running container tag Tag an image into a repository top Display the running processes of a container unpause Unpause all processes within a container update Update configuration of one or more containers version Show the Docker version information volume Manage Docker volumes wait Block until a container stops, then print its exit code

Để xem các lựa chọn có sẵn cho một lệnh cụ thể, hãy nhập:

* docker docker-subcommand --help

Để xem thông tin toàn hệ thống, hãy sử dụng:

* docker info

Bước 4 - Làm việc với Docker Images

Vùng chứa Docker được chạy từ Docker image . Theo mặc định, nó lấy những hình ảnh này từ Docker Hub, một nơi đăng ký Docker được quản lý bởi Docker, công ty đứng sau dự án Docker. Bất kỳ ai cũng có thể xây dựng và lưu trữ Docker image của họ trên Docker Hub, vì vậy hầu hết các ứng dụng và bản phân phối Linux bạn cần chạy containers Docker có hình ảnh được lưu trữ trên Docker Hub.

Để kiểm tra xem bạn có thể truy cập và download hình ảnh từ Docker Hub hay không, hãy nhập:

* docker run hello-world

Đầu ra, phải bao gồm những điều sau, phải cho biết Docker đang hoạt động chính xác:

Output

Hello from Docker. This message shows that your installation appears to be working correctly. ...

Bạn có thể tìm kiếm hình ảnh có sẵn trên Docker Hub bằng cách sử dụng lệnh docker với lệnh con search . Ví dụ: để tìm kiếm hình ảnh CentOS, hãy nhập:

* docker search centos

Tập lệnh sẽ thu thập dữ liệu Docker Hub và trả về danh sách tất cả các hình ảnh có tên trùng với chuỗi tìm kiếm. Trong trường hợp này, kết quả sẽ tương tự như sau:

Output

NAME DESCRIPTION STARS OFFICIAL AUTOMATED centos The official build of CentOS. 2224 [OK] jdeathe/centos-ssh CentOS-6 6.7 x86\_64 / CentOS-7 7.2.1511 x8... 22 [OK] jdeathe/centos-ssh-apache-php CentOS-6 6.7 x86\_64 / Apache / PHP / PHP M... 17 [OK] million12/centos-supervisor Base CentOS-7 with supervisord launcher, h... 11 [OK] nimmis/java-centos This is docker images of CentOS 7 with dif... 10 [OK] torusware/speedus-centos Always updated official CentOS docker imag... 8 [OK] nickistre/centos-lamp LAMP on centos setup 3 [OK] ...

Trong cột **CHÍNH THỨC** , **OK** cho biết hình ảnh được xây dựng và hỗ trợ bởi tổ chức đứng sau dự án. Khi bạn đã xác định được hình ảnh mà bạn muốn sử dụng, bạn có thể tải nó xuống máy tính của bạn bằng cách sử dụng lệnh con pull , như sau:

* docker pull centos

Sau khi một hình ảnh đã được download , bạn có thể chạy một containers bằng cách sử dụng hình ảnh đã download với lệnh con run . Nếu một hình ảnh chưa được download khi docker được thực thi với lệnh con run , thì ứng dụng client Docker trước tiên sẽ download hình ảnh, sau đó chạy một containers bằng cách sử dụng nó:

* docker run centos

Để xem các hình ảnh đã được download máy tính của bạn, hãy nhập:

* docker images

Đầu ra sẽ giống như sau:

[secondary\_lable Output] REPOSITORY TAG IMAGE ID CREATED SIZE centos latest 778a53015523 5 weeks ago 196.7 MB hello-world latest 94df4f0ce8a4 2 weeks ago 967 B

Như bạn sẽ thấy ở phần sau của hướng dẫn này, hình ảnh mà bạn sử dụng để chạy containers có thể được sửa đổi và sử dụng để tạo hình ảnh mới, sau đó có thể được tải lên (được *đẩy* là thuật ngữ kỹ thuật) lên Docker Hub hoặc các tổ chức đăng ký Docker khác.

Bước 5 - Chạy Docker Container

Vùng chứa hello-world bạn đã chạy ở bước trước là một ví dụ về containers chạy và thoát, sau khi phát ra thông báo thử nghiệm. Tuy nhiên, các containers có thể hữu ích hơn thế nhiều và chúng có thể tương tác. Xét cho cùng, chúng tương tự như máy ảo, chỉ thân thiện hơn với tài nguyên.

Ví dụ: hãy chạy một containers bằng hình ảnh mới nhất của CentOS. Sự kết hợp của các lựa chọn **-i** và **-t** cho phép bạn truy cập shell tương tác vào containers :

* docker run -it centos

Dấu nhắc lệnh của bạn sẽ thay đổi để cho biết là bạn hiện đang làm việc bên trong containers và sẽ có dạng sau:

Output

[root@59839a1b7de2 /]#

**Quan trọng:** Lưu ý id containers trong dấu nhắc lệnh. Trong ví dụ trên, nó là 59839a1b7de2 .

Đến đây bạn có thể chạy bất kỳ lệnh nào bên trong containers . Ví dụ, hãy cài đặt server MariaDB trong containers đang chạy. Không cần đặt trước bất kỳ lệnh nào với sudo , vì bạn đang hoạt động bên trong containers với các quyền root:

* yum install mariadb-server

Bước 6 - Commit các thay đổi trong containers thành Docker image

Khi bạn khởi động Docker image , bạn có thể tạo, sửa đổi và xóa các file giống như bạn có thể làm với máy ảo. Những thay đổi bạn thực hiện sẽ chỉ áp dụng cho containers đó. Bạn có thể bắt đầu và dừng nó, nhưng một khi bạn xóa nó bằng lệnh docker rm , các thay đổi sẽ bị mất.

Phần này hướng dẫn bạn cách lưu trạng thái của containers dưới dạng Docker image mới.

Sau khi cài đặt server MariaDB bên trong containers CentOS, bây giờ bạn có một containers đang chạy hình ảnh, nhưng containers khác với hình ảnh bạn đã sử dụng để tạo nó.

Để lưu trạng thái của containers dưới dạng hình ảnh mới, trước tiên hãy thoát khỏi nó:

* exit

Sau đó, commit các thay đổi đối với version Docker image mới bằng lệnh sau. Lựa chọn **-m** dành cho thông báo commit giúp bạn và những người khác biết bạn đã thực hiện những thay đổi nào, trong khi **-a** được sử dụng để chỉ định tác giả. ID containers là ID bạn đã lưu ý trước đó trong hướng dẫn khi bạn bắt đầu phiên docker tương tác. Trừ khi bạn đã tạo các repository bổ sung trên Docker Hub, repository thường là tên user Docker Hub của bạn:

* docker commit -m "What did you do to the image" -a "Author Name" container-id repository/new\_image\_name

Ví dụ:

* docker commit -m "added mariadb-server" -a "Sunday Ogwu-Chinuwa" 59839a1b7de2 finid/centos-mariadb

**Lưu ý:** Khi bạn *commit*một hình ảnh, hình ảnh mới sẽ được lưu local , tức là trên máy tính của bạn. Phần sau của hướng dẫn này, bạn sẽ học cách đẩy một hình ảnh vào register Docker như Docker Hub để nó có thể được bạn và những người khác đánh giá và sử dụng.

Sau khi thao tác đó hoàn tất, liệt kê các Docker image hiện trên máy tính của bạn sẽ hiển thị hình ảnh mới, cũng như hình ảnh cũ mà nó được kế thừa từ :

* docker images

Đầu ra phải thuộc loại này:

Output

REPOSITORY TAG IMAGE ID CREATED SIZE finid/centos-mariadb latest 23390430ec73 6 seconds ago 424.6 MB centos latest 778a53015523 5 weeks ago 196.7 MB hello-world latest 94df4f0ce8a4 2 weeks ago 967 B

Trong ví dụ trên, **centos-mariadb** là hình ảnh mới, được lấy từ hình ảnh CentOS hiện có từ Docker Hub. Sự khác biệt về kích thước phản ánh những thay đổi đã được thực hiện. Và trong ví dụ này, thay đổi là server MariaDB đã được cài đặt. Vì vậy, lần tới khi bạn cần chạy containers bằng CentOS với server MariaDB được cài đặt sẵn, bạn chỉ có thể sử dụng hình ảnh mới. Hình ảnh cũng có thể được tạo từ thứ được gọi là Dockerfile. Nhưng đó là một quá trình rất liên quan nằm ngoài phạm vi của bài viết này. Ta sẽ khám phá điều đó trong một bài viết trong tương lai.

Bước 7 - Liệt kê các containers Docker

Sau khi sử dụng Docker một thời gian, bạn sẽ có nhiều containers hoạt động (đang chạy) và không hoạt động trên máy tính của bạn . Để xem những cái đang hoạt động, hãy sử dụng:

* docker ps

Bạn sẽ thấy kết quả tương tự như sau:

Output

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES f7c79cc556dd centos "/bin/bash" 3 hours ago Up 3 hours silly\_spence

Để xem tất cả các containers - đang hoạt động và không hoạt động, hãy chuyển nó vào lựa chọn -a :

* docker ps -a

Để xem containers mới nhất bạn đã tạo, hãy chuyển nó qua lựa chọn -l :

* docker ps -l

Dừng một containers đang chạy hoặc đang hoạt động đơn giản bằng lệnh :

* docker stop container-id

container-id có thể được tìm thấy trong kết quả từ lệnh docker ps .

Bước 8 - Đẩy Docker image vào Kho lưu trữ Docker

Bước hợp lý tiếp theo sau khi tạo một hình ảnh mới từ một hình ảnh hiện có là chia sẻ nó với một vài người bạn được chọn của bạn, cả thế giới trên Docker Hub hoặc register Docker khác mà bạn có quyền truy cập. Để đẩy một hình ảnh lên Docker Hub hoặc tổ chức đăng ký Docker nào khác, bạn phải có account ở đó.

Phần này hướng dẫn bạn cách đẩy một Docker image vào Docker Hub.

Để tạo account trên Docker Hub, hãy đăng ký tại [Docker Hub](https://hub.docker.com/) . Sau đó, để đẩy hình ảnh của bạn, trước tiên hãy đăng nhập vào Docker Hub. Bạn sẽ được yêu cầu xác thực:

* docker login -u docker-registry-username

Nếu bạn chỉ định password chính xác, xác thực sẽ thành công. Sau đó, bạn có thể đẩy hình ảnh của riêng mình bằng cách sử dụng:

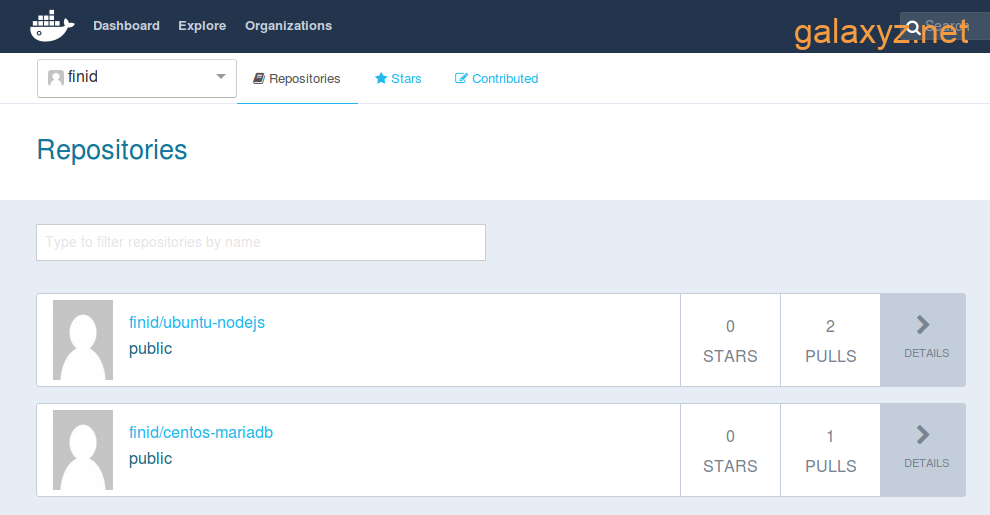
* docker push docker-registry-username/docker-image-name

Sẽ mất một lúc để hoàn thành và khi hoàn thành, kết quả sẽ có dạng như sau:

Output

The push refers to a repository [docker.io/finid/centos-mariadb] 670194edfaf5: Pushed 5f70bf18a086: Mounted from library/centos 6a6c96337be1: Mounted from library/centos ...

Sau khi đẩy hình ảnh vào register , hình ảnh đó sẽ được liệt kê trên trang tổng quan của account của bạn, giống như hiển thị trong hình ảnh bên dưới.



Nếu một việc push dẫn đến lỗi thuộc loại này, thì có thể bạn đã không đăng nhập:

Output

The push refers to a repository [docker.io/finid/centos-mariadb] e3fbbfb44187: Preparing 5f70bf18a086: Preparing a3b5c80a4eba: Preparing 7f18b442972b: Preparing 3ce512daaf78: Preparing 7aae4540b42d: Waiting unauthorized: authentication required

Đăng nhập, sau đó lặp lại việc push .