5A_Installation_Linux

March 27, 2018

1 Installing Curl

Open a terminal window: CTRL+ALT+T.

Type the following command and enter your password:

```
$ sudo apt install curl
```

To check, run the following command in your terminal/command line:

```
$ curl -V
```

Note: The "V" is capitalized.

2 Installing Docker

Docker provides great instructions on how to install it here.

The following directions will assume 64-bit Ubuntu 16.04 VPS, since it is the simplest way to get Docker.

For Ubuntu, you have the choice between the Community Edition (CE) or the Enterprise Edition (EE). We recommend CE, since it is ideal for developers and small teams looking to experiment with Docker.

3 Manage Docker as Non-Root User

If you don't want to use sudo when you use the docker command, create a Unix group called docker and add users to it. When the docker daemon starts, it makes the ownership of the Unix socket read/writable by the docker group.

Warning: The docker group grants privileges equivalent to the root user. For details on how this impacts security in your system, see Docker Daemon Attack Surface.

To create the docker group and add your user:

1. Create the docker group:

   $ sudo groupadd docker

2. Add your user to the docker group:

   ```
   $ sudo usermod -aG docker $USER
   ```

3. Log out and log back in, so that your group membership is re-evaluated.

4. On a desktop Linux environment such as X Windows, log out of your session completely and then log back in.

5. Verify that you can run Docker commands without sudo.

```
$ docker run hello-world
```

6. This command downloads a test image and runs it in a container. When the container runs, it prints an informational message and exits.

4 Docker Compose

To install Docker Compose, run the following commands in your terminal/command line:

```
$ sudo apt update
$ sudo apt install docker-compose
```

Check to make sure that you have Docker version 17.03.1-ce or greater, and Docker Compose version 1.9.0 or greater:

```
$ docker --version && docker-compose --version
```

5 Installing Node.js and npm

To install Node.js and npm, run the following commands in your terminal/command line:

```
$ sudo bash -c "cat >/etc/apt/sources.list.d/nodesource.list" <<EOL
deb https://deb.nodesource.com/node_6.x xenial main
deb-src https://deb.nodesource.com/node_6.x xenial main
EOL
$ curl -s https://deb.nodesource.com/gpgkey/nodesource.gpg.key | sudo apt-key add -

$ sudo apt update

$ sudo apt install nodejs

$ sudo apt install npm
```

Verify the installation, as well as the versions of both Node.js and npm, and make sure the Node.js version you are installing is greater than v6.9 (do not use v7), and the npm version is greater than 3.x:

```
$ node --version && npm --version
```

6 Installing Go Language

Visit https://golang.org/dl/ and make note of the latest stable release (v1.8 or later).

To install Go language, run the following commands in your terminal/command line:

```
$ sudo apt update

$ sudo curl -O https://storage.googleapis.com/golang/go1.9.2.linux-amd64.tar.gz
```

Note: Switch out the black portion of the URL with the correct filename.

```
$ sudo tar -xvf go1.9.2.linux-amd64.tar.gz

$ sudo mv go /usr/local

$ echo 'export PATH=$PATH:/usr/local/go/bin' >> ~/.profile

$ source ~/.profile
```

Check that the Go version is v1.8 or later:

```
$ go version
```