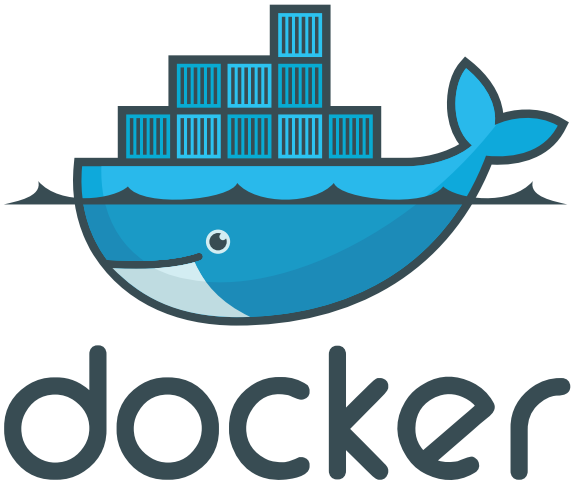

MSE | MASTER OF SCIENCE IN ENGINEERING

Hes·so
Haute Ecole Spécialisée de Suisse occidentale
Fachhochschule Westschweiz
University of Applied Sciences and Arts Western Switzerland

Projet de semestre Docker and embedded systems

Auteur :
Gary Marigliano

Encadrant :
Jean-Roland Schüler

Contact :
gary.marigliano@master.hes-so.ch

Mandant :
Haute École d'ingénierie et d'architecture de Fribourg

Version 0.0.1
2 mai 2016

Historique

Version	Date	Auteur(s)	Modifications
0.0.1	02.05.16	Gary Marigliano	Création du document

Table des matières

1. Introduction

1.1 Contexte

Ce document est le rapport de fin de projet de semestre Docker and embedded systems. Un des buts de ce projet est de cross compiler Docker à partir de ses sources pour produire un binaire exécutable sur un Odroid XU3 (ARMv7). De plus, une partie concernant la sécurité de Docker est également traitée.

Lien : `https://github.com/krypty/docker_and_embedded_systems`

Il est important de noter que la vitesse de développement de Docker est assez hallucinante. En effet, sur Github (`https://github.com/docker/docker`) les commits se succèdent à vitesse grand V. Entre chaque version de Docker qui sortent environ tous les mois, il est courant d'avoir plus de 3000 commits qui ont été *pushés*. Tout ceci pour dire qu'à la lecture de ce document, il est quasiment sûr que certaines pistes explorées soient définitivement obsolètes ou au contraire deviennent la voie à suivre du à une mise à jour quelconque.

1.2 Objectifs

De manière plus précise, ce projet vise à maitriser les parties suivantes :

1. Construction d'un système Linux capable de faire tourner Docker et son *daemon* en utilisant Buildroot. Pour générer le dit système, on dispose d'un *repository* Gitlab hébergé à la Haute École d'ingénierie et d'architecture de Fribourg
2. Cross compilation de Docker et de son *daemon*, capable de faire tourner des containers
3. Comprendre, analyser et évaluer l'aspect sécurité de Docker dans le cadre d'une utilisation avec une carte embarquée

Les deux premiers points ont été traités dans un précédent rapport appelé "État de l'art à la mi-projet de semestre Docker and embedded systems - Ou comment ne pas cross compiler Docker sur ARM".

Ce document se concentre, dès lors, sur le dernier point ainsi que sur le déroulement global du projet.

2. Présentation de Docker

Remarque : Si le lecteur a déjà lu le rapport "État de l'art à la mi-projet de semestre Docker and embedded systems - Ou comment ne pas cross compiler Docker sur ARM", il ne lui est pas nécessaire de relire ce chapitre sachant qu'il s'agit du même contenu.

TODO : inclure un .tex séparé commun aux deux rapports

3. Objectif 3 - TODO

3.1 Situation actuelle

TODO : dire qu'on part d'une distribution Archlinux ARM et qu'on utilise un Odroid XU3 car pas réussi à cross compiler Docker. Tout comme cela avait été fait pour le travail de Bachelor précédent.

3.2 Structure de la suite du document

Pour ce projet, il a été décidé d'étudier la question de la sécurité avec Docker avec une approche en couches. A peu à la manière du modèle OSI [1] en réseau, chaque couche représente un ensemble de fonctionnalités qui, dans le cas de ce projet, doit faire l'objet d'une évaluation de la sécurité.

L'étude de la sécurité de Docker a donc été séparée avec les couches arbitraires suivantes :

- Compilation et installation de Docker : en particulier les options de compilation
- Configuration du système d'exploitation hôte : configuration du kernel, configuration des options de lancement de Docker, etc.
- Création et utilisation des images Docker : Bonnes pratiques et contraintes liées au monde de l'embarqué
- Utilisation des containers

Remarque : Chacune de ces couches fait l'objet d'un chapitre dans ce rapport excepté le premier point : Compilation et installation de Docker. En effet, celui-ci n'est pas traité car, comme annoncé précédemment, la cross compilation de Docker sur un système ARM n'a pas aboutie. L'effort est alors concentré sur les autres points.

1. Modèle OSI : `https://fr.wikipedia.org/wiki/Mod%C3%A8le_OSI`

4. Configuration du système d'exploitation hôte

TODO TODO TODO TODO TODO TODO TODO TODO TODO Dans ce chapitre, on présente diverses bonnes pratiques et configurations dans le but de sécuriser Docker et/ou le système l'hébergeant.

Parmi ces techniques, on peut citer :

— TODO
— TODO

4.1 Passage en revue du benchmark de sécurité : CIS Docker 1.11.0 Benchmark

TODO décrire ce que c'est ce bench, énumerer les points testés et en explorer un certain nombre

4.1.1 Ne pas utiliser AUFS

TODO

4.1.2 User namespace

TODO

4.1.3 Interdiction de la communication réseau entre containers

TODO

4.2 Séparation des données Docker dans une partition chiffrée

TODO

4.3 TODO

TODO

5. Création et utilisation des images Docker

TODO

6. Utilisation des containers

TODO

Bibliographie

[1] Center for Internet Security. CIS Docker 1.11.0 Benchmark, 2016. `https://benchmarks.cisecurity.org/tools2/docker/cis_docker_1.11.0_benchmark_v1.0.0.pdf`.

[2] Adrian Mouat. *Using Docker*. O'Reilly Media, 2005.

Appendices

A. TODO

TODO