Hochschule Darmstadt
Fachbereich Informatik
Dr. Fabio Campos

DIY Containerization

Probevorlesung

Wie in der Vorlesung besprochen, dient Isolation von Containern dazu, Anwendungen und deren Abhängigkeiten in isolierten Umgebungen auszuführen, was eine bessere Sicherheit, Portabilität und Skalierbarkeit ermöglicht. Im ersten Teil der Übung soll die minimale C-Implementierung nachvollzogen und geringfügig erweitert werden. Anschließend wird in der zweiten Aufgabe das Fehlverhalten einer Anwendung simuliert. Hierbei allokiert diese Anwendung exponentiell und unbegrenzt viele Ressourcen eines bestimmten Typs. Dieses Fehlverhalten kann in einer nicht isolierten (containerisierten) Umgebung das gesamte System blockieren. Ihre Aufgabe besteht darin, eine geeignete Anpassung der Implementierung vorzunehmen.

Unter `https://github.com/sopmacF/myCon` finden Sie die minimale C-Implementierung eines Containers.

Zur Verwendung des Containers sind folgende Schritte nötig.

- `git clone https://github.com/sopmacF/myCon.git`
- `gcc myCon.c -o myCon`
- `sudo ./myCon`

Aufgabe 1.1 (Namespaces und System Calls):

Beantworten Sie folgende Fragen zur vorliegenden C-Implementierung.

- Wie wird der Kind-Prozess in einem neuen Namespaces erzeugt?
- Wie erfolgt die Initialisierung mehrerer Namespaces?
- Erweitern sie die Implentierung um einen neuen Namespace für die Netzwerkumgebung

Aufgabe 1.2 (Ressourcenbegrenzung durch `cgroups`):

In dieser Aufgabe[1] werden cgroups zur Limitierung der Anzahl Prozesse, die im Container gestartet werden können, eingesetzt. Die aktuelle Version von MyCon verfügt über keine Begrenzung von Ressoucen, somit würde das Starten des Shellskripts zum Blockieren des Hostsystems führen.

Im zur Verfügung gestellten Dateisystem finden Sie unter `/forkBomb.sh` folgendes Shellskript:

```
1 forkBomb(){                   # Definition der Funktion
2     forkBomb | forkBomb &     # Kopie starten und Ausgabe auf weitere Kopie
3                               # umleiten und im Hintergrund ausfuehren
4 };
5 forkBomb                      # Kettenreaktion starten
```

[1] Inspiriert durch den Vortrag von Liz Rice `https://www.youtube.com/watch?v=8fi7uSYlOdc`

Das Skript[2] erzeugt exponentiell viele Prozesse, indem es rekursiv Kopien seiner selbst startet.

Hinweis: Eventuell ist die Nutzung einer virtuellen Maschine hier sinnvoll.

Als Gegenmaßnahme sollen Sie die dafür vorgesehene Funktion `void limitNumOfProcess()` um folgende Funktionalität ergänzen:

- Ermitteln Sie die ProzessID um die Limitierung mittels cgroups zuzuweisen.
- Erzeugen Sie den benötigten cgroup Ordner.
- Erzeugen Sie die notwendigen Dateien mit passendem Inhalt (s. `pids/pids.max`).
- Weisen Sie die cgroup anhand der ermittelten ProzessID zu.

Testen Sie, ob Ihre Gegenmaßnahme erfolgreich ist.

[2] s. `https://de.wikipedia.org/wiki/Forkbomb`