JUST DO IT
HÖHERE ABTEILUNG FÜR
INFORMATIONSTECHNOLOGIE

tgm
Die Schule der Technik

Laborprotokoll

Verteilte Systeme

Systemtechnik Labor
5YHITM 2016/17, Gruppe A

Maximilian Seidl

Version 0.1
Begonnen am 05.04.2017
Beendet am 07.04.2017

Note:
Betreuer: Th.Micheler

Inhaltsverzeichnis

1 Installation und Implementierung

Ori is a distributed file system built for offline operation and empowers the user with control over synchronization operations and conflict resolution. We provide history through light weight snapshots and allow users to verify the history has not been tampered with. Through the use of replication instances can be resilient and recover damaged data from other nodes.

Installieren Sie Ori und testen Sie die oben beschriebenen Eckpunkte dieses verteilten Dateisystems (DFS). Verwenden Sie dabei auf jeden Fall alle Funktionalitäten der API von Ori um die Einsatzmöglichkeiten auszuschöpfen. Halten Sie sich dabei zuallererst an die Beispiele aus dem Paper im Kapitel 2 [2]. Zeigen Sie mögliche Einsatzgebiete für Backups und Roadwarriors (z.B. Laptopbenutzer möchte Daten mit zwei oder mehreren Servern synchronisieren). Führen Sie auch die mitgelieferten Tests aus und kontrollieren Sie deren Ausgaben (Hilfestellung durch Wiki [3]).

1.1 Gegenüberstellung

Wo gibt es Überschneidungen zu anderen Implementierungen von DFS? Verwenden Sie dabei HDFS [4], GlusterFS [5] oder ein DFS Ihrer Wahl als Gegenspieler zu Ori. Um aussagekräftige Vergleiche anstellen zu können, wäre es von Vorteil die anderen Systeme ebenfalls - zumindest oberflächlich - zu testen. Dokumentieren Sie die Einsatzgebiete der DFS.

1.2 Info

Gruppengröße: 2 Mitglieder Gesamtpunkte: 16

- Installation und Testdurchlauf von OriFS: 3 Punkte
- Installation und Testdurchlauf eines anderen DFS: 3 Punkte
- Einsatz/Dokumentation der Ori API (Waehlen Sie 6 Funktionen aus folgender Liste: replicate, snapshot, checkout, graft, filelog, list, log, merge, newfs, pull, remote, removefs, show, status, tip, varlink): 4 Punkte
- Gegenüberstellung der beiden DFS (Allgemein, Staerken, Schwaechen): 4 Punkte
- Einsatzgebiete von DFS : 2 Punkte

1.3 Quellen

- Ori File System, Stanford Website, online: http://ori.scs.stanford.edu/, visited: 2016-04-01
- Ori File System, Bitbucket Wiki, online: https://bitbucket.org/orifs/ori/wiki/Home, visited: 2016-04-01
- Ali José Mashtizadeh, Andrea Bittau, Yifeng Frang Huang, David Mazières. Replication, History, and Grafting in the Ori File System. In Proceedings of the 24th Symposium on Operating Systems Principles, November 2013. Paper.

- Apache Hadoop FileSystem, http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/HdfsUserGuide.html, visited: 2016-04-01
- GlusterFS, http://gluster.readthedocs.org/en/latest/, visited: 2016-04-01

2 Ergebnisse

Aufgrund der Aufgabenstellung wurden folgende Funktionalitäten bei GlusterFS und bei OriFS konfiguriert und getestet.

- replicate – Erstellung einer replizierten (verteilten) Speicherschnittstelle
- snapshot – Anlegen einer Sicherung
- newfs – Aufsetzen/Konfiguriation eines Dateisystems
- pull – Aktualisieren der lokalen Speicherständen mit Master-Entität
- checkout – Zurücksetzen auf früheren Zustand
- log – Auslesen von Informationen über bisherigen Zustände des Dateisystems

3 GlusterFS

3.1 Installation

Um die Installation von GlusterFS durchzuführen müssen zwei virtuelle Maschinen aufgesetzt werden. Es gibt zwei Möglichkeiten dies zu tun, entweder über das Aufsetzen von zwei Systemen unter VirtualBox, oder die schnellere Variante unter Vagrant.

```
1   vagrant init ubuntu/trusty64; vagrant up --provider virtualbox
```

Listing 1: Vagrant Setup

Damit die Maschinen Ihren Speicher teilen können muss zunächst ein neues Netzwerk hinzugefügt werden, welches einem Host-Only entspricht. Zusätzlich muss noch eine Partition auf beiden Systemen erstellt werden, damit darin der gemeinsame Speicherbereich liegt. Bei beiden VMs ist die Partition, welche mittels fdisk erstellt wurde ein Gigabyte groß.
All diese Einstellungen müssen im Vagrantfile eingestellt werden.
Damit die gerade eben erstellten Festplatten verwendet werden können, müssen diese noch mit einem Dateisystem beschrieben werden. XFS wird als Dateisystem gewählt, da es alle Änderungen speichert und somit eine spätere Rekonstruierung möglich ist. Im nächsten Listing werden die Befehle für die Vorbereitung der Fesplatten vorgezeigt.

```
1   mkfs.xfs -i size=1024 /dev/sdb1
    mkdir -p /data/brick1
3   echo '/dev/sdb1 /data/brick1 xfs defaults 1 2' >> /etc/fstab
    mount -a && mount
```

Listing 2: Vorbereitung der Festplatten

Um die Verbindung zwischen den beiden Maschinen über GlusterFS möglich zu machen, muss bei beiden VMs jeweils der Hostname der anderen Maschine in die /etc/hosts- Datei eingetragen werden. Jetzt muss zwischen den Maschinen ein trusted pool aufgebaut werden, welcher aus mehreren Maschinen bestehen kann. In diesem Beispiel sind aber nur die beiden Testmaschinen inkludiert. Damit so ein pool aufgesetzt werden kann muss ein Package mit dem Namen glusterfs-server installiert werden. Mittels eines probing zwischen den Maschinen wird so ein Pool aufgebaut.

Jetzt müssen noch Ordner für den gemeinsamen Speicherbereich angelegt werden. Dies geschieht über gluster volume create, wie im folgenden Listing vorgeführt wird.

```
gluster volume create shared replica <number of bricks/servers> seidl_test1:shared_folder
    seidl_test2:shared_folder gluster volume start shared_folder
```

Listing 3: Gluster Volume

Mit der fertigen Installation sind bereits 3 der geforderten Anforderungen erfüllt. Durch das Anlegen von Dateien auf einer Maschine in dem geteilten Ordner, ist es jetzt möglich zu testen ob das ganze System funktioniert.

```
1 touch testdatei.txt
  nano testdatei.txt
```

Listing 4: Erstellen einer Datei im Shared folder

4 OriFS

4.1 Installation

Die Installation erfolgt ebenfalls über zwei virtuelle Maschinen, welche mit Vagrant aufgesetzt und konfiguriert werden. Da OriFs nicht im offiziellen apt Repository vorhanden ist, muss es manuell aus einem anderen Repository hinzugefügt werden. Hierbei stellen die Entwickler ein Package in einem third-party Packagerepository (PPA)

```
  sudo add-apt-repository ppa:ezyang/ppa
2 sudo apt update
  sudo apt install ori
```

Listing 5: Erstellen einer Datei im Shared folder

Die Dateiübertragung bei OriFS funktioniert über SSH und somit muss zwischen den beiden Hosts ein SSH-Zugriff ermöglicht werden. Hierbei muss ein gemeinsamer SSH-Key erstellt werden und bei den jeweiligen Hosts in die .ssh/authorized_keys eingetragen werden.

4.2 Konfiguration und Funktionsweise

OriFS erstellt Daten nicht auf Partitionen sondern in Repositories, diese müssen dann bei dem System gemountet werden. Das darauffolgende Listing beschreibt die Erstellung eines solchen Repositories und das mounten über die benötigten Befehle.

```
1 ori newfs testRepo
  mkdir testRepo
3 orifs testRepo
```

Listing 6: Konfiguration eines OriFS Repositories

Jetzt ist auf einer der beiden Testmaschinen ein Repository erstellt worden. Damit aber die zweite virtuelle Maschine ebenfalls auf diesen Datenbereich zugreifen kann, hat OriFS eine ähnliche Funktionsweise wie Git. Um eine lokale Änderung im Repository zu übernehmen muss ein sogenanter snapshot gemacht werden, welcher wie ein commit von Git funktioniert. Auf einem anderen System

kann jetzt mittels dem Befehl pull die gesammelten snapshots lokal auf die Maschine geladen werden. Bei gleichzeitigen Änderungen im Repository von verschiedenen Standorten, muss ein merch ausgeführt werden um die unterschiedlichen snapshots zusammen zu führen. Im nächsten Listing werden die gerade genannten Befehle dargestellt.

```
ori replicate seidl@192.168.66.12:testRepo
mkdir testRepo
orifs testRepo
```

Listing 7: OriFS Befehle

```
ori pull
Pulled up to edaf575111225a8a120cb98081a649a82a8915a7f01cfb04848f82456b5012f5

ori checkout edaf575111225a8a120cb98081a649a82a8915a7f01cfb04848f82456b5012f5
Checkout success!

ori snapshot
Committed edaf575111225a8a120cb98081a649a82a8915a7f01cfb04848f82456b5012f5
```

Listing 8: OriFS Befehl pull und snapshot

4.3 Synchronisierung

Mit den oben beschriebenen Befehlen erhält man keine automatische Synchronisierung zwischen den Maschinen, sondern es muss jede Änderung manuell von der anderen Maschine geholt werden. OriFS bietet hierfür den orisync-Dienst an, welcher das snapshoten und pullen automatisiert, da er Änderungen von selber erkennt.

Theoretisch sollte dies einwandfrei nach Aktivierung dieses Dienstes funktionieren, jedoch hat es bei mir nach mehreren Versuchen nicht funktioniert. Das erstellen eines Clusters mittels dem Begfehl orisync init, sollte die Maschinen zusammenführen, doch hierbei traten immer wieder Fehler auf.

4.4 Logging

In OriFS ist ein Einblick in log-Dateien sehr einfach gestaltet und funktioniert über den Befehl ori log. Dieser gibt einen Einblick in die Meta-Daten eines Repositorys.

```
Name File                   System ID
testRepo                    be52494d-78f7-423c-9347-083e6299a1f8
seidl@vagrant-ubuntu-trusty-64:~/testRepo$ ori log
Commit:     134608697e8308ca056c87dda42ba7e8c2cc1ef3749a57384201220cd75a6921
Parents:    edaf575111225a8a120cb98081a649a82a8915a7f01cfb04848f82456b5012f5
Tree:       850eb6fab4785c419a30b6619d78f590d80e46a85e348f4cbfb241606fc2436e
Author: Date: Wed Apr 05 17:12:33 2017
Created snapshot 'firstSnap'
Commit:     edaf575111225a8a120cb98081a649a82a8915a7f01cfb04848f82456b5012f5
Parents:
Tree:       52869c32902bf5245a41a0028b48f7768d1900d403ec12f8302c56e91d50539f
Author:
Date: Wed Apr 05 16:46:04 2017

No message.
```

Listing 9: ori log

5 Fazit - Gegenüberstellung

Aufgrund der Probleme bei der Installation und Konfiguration von OriFS war es mir relative schwer einen sinnvollen Zusmamenhang zwischen den beiden verteilten Dateisystemen zu finden. Abgesehen von der Funktionsweise unterscheiden sich die beiden DFS lediglich im Einsatzbereich. OriFS zum Beispiel wäre keine gute Lösung für ein verteiltes Dateisystem in einem Unternehmen, da es seit 2014 keine weiteren Updates gegeben hat und die Dokumentation nur spärlich vorhanden ist. GlusterFS dagegen ist nach einigen problemlosen Schritten voll einsatzfähig und ist deswegen auch sehr weit verbreitet. Der native Support von verschiedenen Linux-Distiributionen ist ebenfalls ein weiterer Punkt, welcher GlusterFS über OriFS stellt. Der einzige Nachteil den Schulkollegen im Laufe der Durchführung der Laborübung entdeckt haben, ist die Vernachlässigung von Linux-Paketen, welche auf nicht RedHat-basierten Distributionen vorkommt.

Tabellenverzeichnis

Listings

Abbildungsverzeichnis