High Availability (HA) Distributed File Storage

Document: Developers Documentation

Version 1.0

Publication date: 2-MAY-2016

Group name: Gryffindor

Group Members:

- ARJITH CHALASANI

- JITENDRA NEELAM

- URMILA JYOTHULA

- CHINNA BALAJI YALLA

- NITISH NAGABHAIRAVA

- PRANEEL REDDY PADALA

- SURYA TEJA BOLLIMPALLI

- SRI GANESH SAI GUNNAM

- SRI KASYAP KAPPAGANTULA

- SIRISHA MANASWINI BHAMIDI

- RAGHUVINAYAK RAO MEDISETTI

- MAHAMMAD SUHAIL ATCHUKATLA

1. Preface

The main concept of this project is to develop a secure file storage to the company SecureFile in the form of a distributed file storage system with high availability to the customers.

When a user uploads a file, the file is stored in a randomly chosen server. In this we are creating replicas for the file uploaded by the user and we use file transfer protocol for the transfer of data.

Service Developer: Gryffindor

Customer: Dragos llie

In this document we defined the technical terms and a short note on them, Organisation of source code, Monitoring process Set up a control node and references.

- **Release v1.0 on 2016-04-18**
  Initial Release

2. Glossary and abbreviations

- **HTTP: Hyper Text Transfer Protocol**

  It is a transfer of version data formats between server and client

  EX: plain txt, hyper txt, video and sound

- **FTPS: File Transfer Protocol Security**

  It is an extension for commonly used file transfer protocol(FTP) that adds support for the transfer layer security(TLS) and secure sockets layer (SSL)

- **Message digest: SHA-1**

  IT is a crypto graphic hash function which is consider practically impossible to invert that is to recreate the input data from its hash value alone.

SHA-1: secure Hash algorithm.SHA-1 produces a 160bit (20 byte) hash value known as a message digest. SHA-1 advancements are SHA-2 and SHA-3

- **GUI: Graphical User Interface**

  It is a type of interface which helps in interaction with electronic devices through graphical icon and visual indicators.

- **SQL Server: Structured Query Language Server**

  SQL is used to store, query and manipulate data. It is used for manage data in a relational data base.

- **Restful API: Representation State Transfer**

  An architectural pattern to improve probability and scalability of a system.

INTRODUCTION:

This document gives an idea for the further extension of the tool development. This document gives the idea to the user or a developer on what platforms the product is developed

INTERPRETATION OF TOOL:

In order to develop the product, we used three main aspects they are the front end, back end and data base. we use html, css and php in order to develop the front end they are used by the customers to register, login and observe the home page. The database is created using the sql database all the data is stored in this database. The backend is used to perform the pinging and replication of the files the replication is done in such a way that the file is present in the rest of the servers. When one of the server is down the user can download the file from other servers.

System Architecture
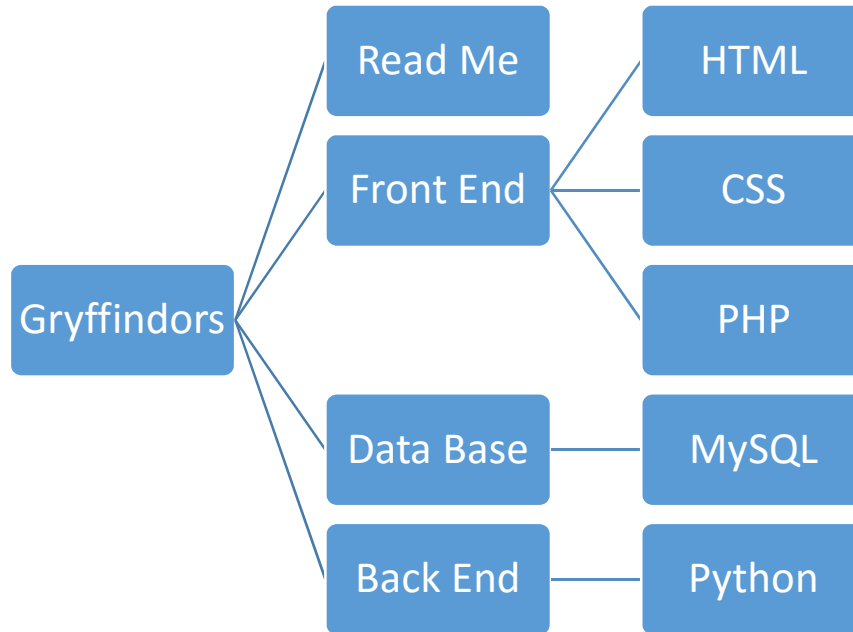
Programming languages used

Python

Html

Php

Css

Mysql

Other requirements

Mysql database

Organisation of source code:

```
                    ┌──────────┐      ┌──────────┐
                    │ Read Me  │      │   HTML   │
                    └──────────┘      └──────────┘
                    ┌──────────┐      ┌──────────┐
                    │Front End │      │   CSS    │
                    └──────────┘      └──────────┘
┌────────────┐                       ┌──────────┐
│ Gryffindors│                       │   PHP    │
└────────────┘                       └──────────┘
                    ┌──────────┐      ┌──────────┐
                    │Data Base │      │  MySQL   │
                    └──────────┘      └──────────┘
                    ┌──────────┐      ┌──────────┐
                    │ Back End │      │  Python  │
                    └──────────┘      └──────────┘
```

Monitoring process:

```
sri@ubuntu:~/db-cluster-utils-master
sri@ubuntu:~$ cd db-cluster-utils-master
sri@ubuntu:~/db-cluster-utils-master$ python db_cluster_utils.py -h
Usage:
db_cluster_utils.py -option [argument]
  -a    --add <slave ip>       add a new slave to the cluster
  -c    --config               view cluster configuration information
  -d    --demote               demote a master to a slave
  -h    --help                 display this help message
  -i    --init                 initialize the cluster based on settings
                               in /etc/db_cluster_utils/cluster_utils.conf
  -m    --move <destination ip> Move the database to another server
  -p    --promote <slave ip>   promote a slave to master
  -r    --remove <slave ip>    remove this node from the cluster
  -s    --start                start replication
  -t    --stop                 stop replication
  -w    --wipe [IP address]    wipe a database from the entire cluster, or
                               from a single server if an IP address
                               is specified

sri@ubuntu:~/db-cluster-utils-master$ python db_cluster_utils.py -i
Database name: suhail_test
Master server IP: 192.168.142.129
Control Host: 192.168.0.198
Slave IP address list:
['192.168.112.129']
Root password: password
User password: password
Slave user password: password
```

The tool retrieves the information about the server being online from the ssh connections

Set up a control node:

Set up a host name and password using the dbi.config. the control node is set up using the mysql and python bindings.to install use the command used is

Sudo apt-get install python-mysqldb mysql-client-y

Run bpi config to expand the root partition and change the hostname and password

References

- Sommerville, Ian. Software Engineering, 9th ed. Addison-Wesley, 2011