

サーバ管理演習のログ収集システムマニュアル

このプロダクトは、Docker がインストールされている PC を前提としている。

このプロダクトでは、サーバの現状を把握するために、コマンド実行履歴、ファイル編集履歴、サーバ状況確認履歴を収集している。以下にそれぞれの履歴収集システムの起動方法を示す。

まず、Github からリポジトリをクローンまたは、ダウンロードする。

https://github.com/thedekopin/Learning_Support_System

次に、ダウンロードした node_mongo というディレクトリに移動する。

移動できたら下記コマンドを実行。

- docker compose up

事前準備

データベース

収集したログを格納するために db コンテナがある。下記コマンドで db コンテナにログインする。

- `docker exec -it db /bin/bash`

データベースは、`mongodb` で作成されており、下記コマンドを実行する。

- `mongo`

`mongo` コマンドを実行すると `mongo` シェルを起動できる。`mongoDB` を利用するために `admin` ユーザを追加するために、`mongo` シェルで下記コマンドを実行する。

```
use admin
```

```
db.createUser({user: "root", pwd: "toor", roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]});
```

```
db.auth("root", "toor")
```

上記のコマンドを入力した結果、1 が端末に表示されたら成功。

次に、履歴を収集するために `test` ユーザを作成する。

```
use test
```

```
db.createUser({user: "teacher", pwd: "rehaet", roles: [ { role: "readWrite", db:
"test" }, { role: "read", db: "reporting" } ] });
db.auth("root", "toor")
```

上記のコマンドを入力した結果、1 が端末に表示されたら成功。

db コンテナから exit コマンドを実行してログアウトする。

コマンド実行履歴とファイル編集履歴

コマンド実行履歴とファイル編集履歴は、学生のコンテナで収集を行うため、事前に学生用コンテナ内でいくつかのコマンドを実行する。

コマンド実行履歴

演習課題用コンテナ student にログインする。

- `docker exec -it student /bin/bash`

student コンテナに用意されている inotify ディレクトリに移動し、中にある

sam_sig.py を下記のコマンドにより実行する。

- `source ~/.bashrc`
- `pip3 install pymongo`
- `python sam_sig.py`

ファイル編集履歴

演習課題用コンテナに用意されている `inotify` ディレクトリに移動し、中にある

`inotify.js` を下記のコマンドにより実行する.

- `node inotify.js`

上記コマンドが全て実行できたら、`student` コンテナからログアウトする.

サーバ状況確認履歴

- `docker exec -it node /bin/bash`

ログインできたら/home/teacher/web/ディレクトリに移動し、下記コマンドを実行する。実行後、ブラウザで localhost:3000 にアクセスすることで学生の進捗状況を確認できる Web サイトを見ることができる。

- npm start