

GakuNinRDM ストレージ マウンター使用手引書

0.1 版

2024.07.09

データ管理システム開発ユニット) 平岡 千明

目次

1	はじめに.....	4
2	環境	4
3	マウンターのインストール.....	4
4	環境変数ファイル.env について	5
5	マウンター (rdm_mount コマンド) の使用方法.....	7
6	マウンター (rdm_mount コマンド) の終了方法.....	8

1 はじめに

GakuNinRDM で使用しているアドオンストレージを Linux 上のディレクトリにマウントし、Linux 上でファイルのコピー等をできるようにするツールです。

2 環境

Linux(Ubuntu 22.04 等)の OS で python3, wget, zip, unzip がインストールされていることが前提条件となります。

3 マウンターのインストール

① インストーラーの取得

以下のコマンドにより取得します。

```
wget https://dmsutil.riken.jp/tool/cs-rdmfs\_install.gip
```

例)

```
$ cd /tmp
```

```
$ wget https://dmsutil.riken.jp/tool/cs-rdmfs\_install.gip
```

② インストール (スーパーユーザーになって行ってください。)

- 1) インストーラーおよびマウンターコマンドを展開するディレクトリを作成し①のファイルを展開します。

例)

```
$ sudo su -
```

```
# mkdir -p /usr/local/rdmfs/bin
```

```
# cd /usr/local/rdmfs/bin
```

```
# cp /tmp/cs-rdmfs_install.gip .
```

```
# unzip cs-rdmfs_install.gip
```

```
Archive: cs-rdmfs_install.zip
```

```
[cs-rdmfs_install.zip] install.sh password: ←パスワード入力
```

※ユーザーが全員利用できるディレクトリを推奨

```
パスワード: riken_r2dms_202406
```

- 2) マウンターアプリをインストールするディレクトリを作成します。

例)

```
# mkdir -p /opt/rdmfs
```

- 3) インストーラーの実行

2) で作成したディレクトリに移動し、フルパスで1) のインストーラーを起動します。

例)

```
# cd /opt/rdmfs
# /usr/local/rdmfs/bin/install.sh
```

※実行後は、/opt/rdmfs/CS-rdmfs が作成される。

install.sh のディレクトリ配下には環境変数ファイルの雛形が /usr/local/rdmfs/bin/.env_template(ディレクトリは例)がありますのでユーザー毎に利用するディレクトリにコピーし必要な値を設定し、.env ファイル名にするか、任意のファイル名にし、任意のファイルの場合には、マウンターコマンド実行時にパラメタで指定する必要があります。

4) インストール後の環境設定

スーパーユーザー以外のユーザーでマウントする場合には、/etc/fuse.conf を修正します。

```
#user_allow_other
の行の#を削除する。(コメントを解除する。)
```

4 環境変数ファイル.env について

デフォルトの.env ファイル

```
GRDM_ACCESS_TOKEN=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
GRDM_PROJECT_ID=XXXXX
GRDM_MOUNT_POINT=/mountpoint/dir
GRDM_OSF_API_BASEURL="https://rdm.riken.jp:8000/v2/"
GRDM_WB_API_BASEURL="https://rdm.riken.jp:7777/"
GRDM_STORAGE_PROVIDER=s3compat
GRDM_TARGET_FOLDER=/
GRDM_UPLOAD_CONCURRENT=10
GRDM_UPLOAD_REPLACE_FLAG=False
GRDM_LOCAL_DELETE_FLAG=False
GRDM_OUT_LOG="/tmp/`date`
+%Y%m%d%H%M%S`_rdm_mount_${USER}_${GRDM_PROJECT_ID}_log.txt"
```

各環境変数の説明

環境変数名	説明
GRDM_ACCESS_TOKEN	GakuNinRDM で設定したパーソナルアクセストークン
GRDM_PROJECT_ID	GakuNinRDM 上のプロジェクト ID
GRDM_MOUNT_POINT	GRDM のストレージをマウントするローカルのフォルダのパス(パスはフルパス指定)
GRDM_OSF_API_BASEURL	GakuNinRDM の OSF API のベース URL
GRDM_WB_API_BASEURL	GakuNinRDM の waterbutler API のベース URL
GRDM_STORAGE_PROVIDER	ダウンロード/アップロードするデータを格納するストレージ名
GRDM_TARGET_FOLDER	データのダウンロード元/アップロード先になる GRDM のストレージのフォルダのパス [default はトップ]
GRDM_UPLOAD_CONCURRENT	データをアップロードする際に同時処理する数 [default は 10 処理]
GRDM_UPLOAD_REPLACE_FLAG	データをアップロードする際に元のデータを上書きするかどうかのフラグ [default は上書きしない]
GRDM_LOCAL_DELETE_FLAG	データをアップロード後にローカルのデータを削除するかどうかのフラグ [default は削除しない]
GRDM_OUT_LOG	コマンドのログの出力ファイルのパス

なお、環境変数ファイルは、ダウンロード/アップロードのプログラムと共通化されています。以下にマウンターとダウンロード/アップロードのプログラムで使用する環境変数の表を示します。

環境変数名	マウンターで使用する環境変数	ダウンロード/アップロードで使用する環境変数
GRDM_ACCESS_TOKEN	○	○
GRDM_PROJECT_ID	○	○
GRDM_MOUNT_POINT	○	○
GRDM_OSF_API_BASEURL	○	○
GRDM_WB_API_BASEURL		○
GRDM_STORAGE_PROVIDER		○
GRDM_TARGET_FOLDER		○
GRDM_UPLOAD_CONCURRENT		○
GRDM_UPLOAD_REPLACE_FLAG		○
GRDM_LOCAL_DELETE_FLAG		○
GRDM_OUT_LOG	○	

●マウンターコマンドが使用する環境変数ファイルの優先順位について

- ①マウンターコマンドを実行したディレクトリ（カレント）の.env ファイル（存在していれば）
- ②①が存在しない場合には、マウンターコマンド（rdm_mount）が格納されているディレクトリ配下の.env ファイル（存在していれば）
- ③rdm_mount コマンドに指定した環境変数ファイル名

優先順位： ③ > ② > ①

5 マウンター（rdm_mount コマンド）の使用方法

- ① /usr/local/rdmfs/bin/rdm_mount を実行します。

Usage : rdm_mount [-mp マウントポイント] [環境変数ファイル名]

※環境変数ファイル名は、フルパスでも指定可能

また、マウントポイントはフルパスで指定

- ② ①のマウンター（rdm_mount コマンド）でマウントが成功した場合には、以下のよう
なログが標準出力に出力されます。

実行時ログ（例）

```
ENVIRONMENT_FILE=/home/user/.env
GRDM_MOUNT_POINT=/tmp/test
GRDM_PROJECT_ID=yyyy
GRDM_ACCESS_TOKEN=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
GRDM_OSF_API_BASEURL=https://grdm.xxxx.xxx:8000/v2/
GRDM_OUT_LOG=/tmp/rdm_mount_log.txt
CS_RDMFS_DIR=/opt/rdmfs
rdm_mount: /tmp/test mount Success
```

※ENVIRONMENT_FILE は、どこの環境変数ファイルを使用したのかが分かりません。

また、CS_RDMFS_DIR は、マウンターアプリのパス名が分かります。

mount Success が出力されれば、正常にマウント出来たこととなります。

mount Error の場合には、マウントに失敗したこととなります。

6 マウンター (rdm_mount コマンド) の終了方法

umount **マウントポイント**(環境変数 GRDM_MOUNT_POINT で指定した値)

※マウントポイントにアクセス中の場合にはアンマウントできません。

また、マウントを確認するには、mount コマンドで確認できます。

mount コマンドの結果例)

```
rdmfs_asyncio      on          /tmp/test         type      fuse
(rw,nosuid,nodev,relatime,user_id=1000,group_id=1000,default_permissions,allow_other)
```