



土屋 健

画像の保存に オブジェクト・ストレージ

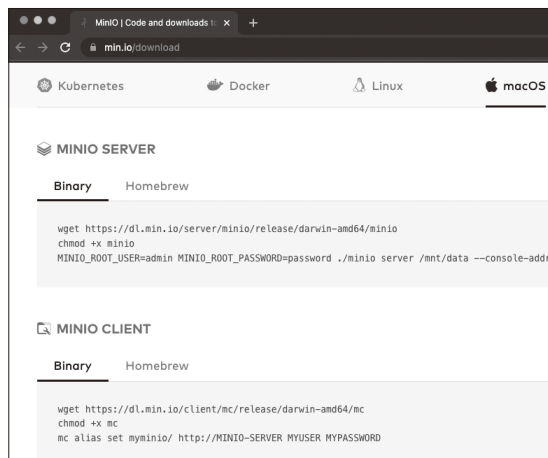


図1 MinIO ダウンロード・ページ

● カギはMinIO…Amazonにもスケールできる オープンなストレージ

IoTでも画像データを扱うことがあります。画像はデータ・サイズが大きく、また、保存や参照が頻繁に行われるので、InfluxDBやEthereumで保存するには適しません。そこでオブジェクト・ストレージを用意してデータの保存場所にします。

オブジェクト・ストレージと言えばAmazon S3が有名ですが、S3はクラウドの利用が前提です。そこで今回は、S3互換のオープンソース・ストレージであるMinIO (<https://min.io/>)を使用します。

MinIOはGo言語で実装されており、Amazon S3と互換性のあるストレージです。S3クライアント・ライブラリを通してアクセス可能なので利用しやすいというメリットもあります。また、接続先を変更すれば、そのままS3を利用できる点もシステム開発においてはメリットとなります。

● 導入ステップ

MinIOを使用する準備を整えます。作業は以下の手順で進めます。

- 1) MinIOをインストール
- 2) データ・サーバでファイルの保存場所としてMinIOを使えるようにする
- 3) データ・サーバにファイル・アクセスAPIを追加する

ステップ1…MinIOのインストール

● **ダウンロードしたファイルに実行権を付与する**
MinIOのダウンロード・ページからバイナリをダウンロードします(図1)。

<https://min.io/download>

macOS、Linux、Windows向けのバイナリが用意されています。

MinIOはシングル・バイナリでダウンロードしたコマンドを配置するだけで利用できるのでセットアップは手間がかかりません。以降は、macOS環境で説明しますが基本的な手順はどの環境でも同じです。また、各環境のダウンロード・ページに詳しい手順が書かれています。

▶ MinIOサーバのインストール

macOSの場合は以下のファイルをダウンロードします。

[https://dl.min.io/server/minio/
release/darwin-amd64/minio](https://dl.min.io/server/minio/release/darwin-amd64/minio)

ダウンロードしたファイルは、\$HOME/CQ/MyDataServer/MinIOに配置します。ターミナルで以下のコマンドを実行し、ダウンロードしたファイルに実行権を付与します。

```
$ chmod +x minio
```

▶ MinIOクライアントのインストール

macOSの場合は以下のファイルをダウンロードします。

[https://dl.min.io/client/mc/
release/darwin-amd64/mc](https://dl.min.io/client/mc/release/darwin-amd64/mc)

ダウンロード・ファイルは、\$HOME/CQ/MyDataServer/MinIOに配置します。ターミナルで次のコマンドを実行しダウンロードしたファイルに実行権