



アクセス情報の登録と アクセス・ライブラリの準備

土屋 健

表1 Ethereumへアクセスできるように接続情報を登録する

項目	設定値
id	Ethereum
name	Etherem private net
address	192.168.3.3
port	8545
account	0x70f9e0445a572d62eabf6d6669f69283558d7e2f
password	dataprowider
private_key	c2f7b10487844a8d0f~~~~8557bc71280e28

前章でEthereumによるプライベートなネットワークを構築することができました。本章では、特集第2部で制作したデータ・サーバから、Ethereumへアクセスする準備を行います。

Ethereumアクセス情報の登録

Ethereumへアクセスできるように接続情報を登録します。InfluxDBの場合と同様にMySQLのds_ethereumテーブルに必要な情報を登録し、プログラムから参照できるようにします。登録情報は表1の通りです。

● gethコンソールで登録したアカウントのプライベート・キーを入手する方法

アカウントのプライベート・キー(private_key)の入手にはちょっと手間がかかります。gethで作成したアカウントのプライベート・キー情報は、

```
<DATADIR>/keystore
```

配下に保存されていますが、そのままでは参照できないので、keythereumというライブラリを使用して取り出します。

```
https://github.com/ethereumjs/keythereum
```

取り出しのためにはJavaScriptの実行環境が必要なので、Node.jsをインストールしてリスト1のように取り出します。

● 接続情報の登録

Ethereumへの接続情報の登録もSQL実行またはLIB/setethereum.pyツールで行います。リスト2にLIB/setethereum.pyツールで登録する方法を示します。リスト3で登録内容を確認します。

ds_ethereumテーブルに情報を登録したらデータ・サーバで接続情報を参照できるようになります。

Ethereumアクセス・ライブラリの準備

Ethereumにアクセスしてコントラクトを実行するためのライブラリ(LIB/ethereum.py)を準備します。

PythonでEthereumへアクセスするプログラムを開発するためのライブラリとしてWeb3.py(<https://github.com/ethereum/web3.py>)が提供されており、それを利用して実装します。Web3.pyを使うと簡単な記述でEthereumへアクセスしスマート・コントラクトを実行できます。

まずはPython環境にWeb3.pyライブラリをインストールします(pip install web3実行)。リスト4に実行例を示します。

● 実装

次項でEthereumアクセス・ライブラリ(LIB/ethereum.py, リスト5)の実装を説明します。Ethereumアクセス・ライブラリはEthereumクラスとして実装しており、機能としては、

- Ethereumへの接続
- スマート・コントラクト呼び出し
 - 更新系…トランザクション系
 - 参照系…ファンクション呼び出し

となります。トランザクション系はスマート・コントラクト呼び出し時、コストのかかる処理の呼び出しに使用します。また、データ・サーバからスマート・コントラクトを呼び出すためにはインターフェース定義が必要です。ここでは、

- 1) センサ・データ記録