

FreeDMR Digital API von SDR-MKK

Anleitung wie man Nutzerdaten von FreeDMR Digital über SDR-MKK bezieht.

Um die API nutzen zu können muss der Code von OpenWebRX etwas bearbeitet werden, ich werde hier zeigen was geändert werden muss.

1. Als erstes müssen wir uns per SSH auf den WebSDR schalten, und mit folgendem Befehl die Datei "meta.py" ändern.

```
sudo nano /usr/lib/python3/dist-packages/owrx/meta.py
```

2. Hier suchen wir mit "STRG + W" nach folgendem Text.

```
data = self._downloadRadioidData(id)
```

3. Jetzt müssen wir folgendes ändern und zwar schreiben wir denn unten folgenden Text direkt darunter.

```
data = self._downloadRadioidData(id)
```

```
if data == None:
```

```
    data = self._downloadRadioidDataFreeDMR(id)
```

4. Jetzt müssen wir die neue Funktion noch einbinden und das machen wir wie folgend. Über der Funktion "def _downloadRadioidData(self, id):" schreiben wir diesen Code.

```
def _downloadRadioidDataFreeDMR(self, id):
```

```
    try:
```

```
        logger.debug("requesting radioid metadata for mode=%s and id=%s", self.mode, id)
```

```
        res = request.urlopen("http://sdr-mkk.de/api.php?id={0}".format(id), timeout=30)
```

```
        if res.status != 200:
```

```
            logger.warning("returned with error %i id=%s", res.status, id)
```

```
            return None
```

```
        data = json.loads(res.read())
```

```
        if "results" in data:
```

```
            for item in data["results"]:
```

```
                return item
```

```
        except json.JSONDecodeError:
```

```
            logger.warning("unable to parse response JSON")
```

```
        except HTTPError as e:
```

```
            logger.warning("responded with error: %s", str(e))
```

```
        return None
```

5. Jetzt müssen wir nur noch einmal OpenWebRX neustarten und wir sind fertig

```
sudo systemctl restart openwebrx
```