

Konsumering af REST i Python

Formål: At lave en REST consumer i Python

Baggrund:

- [Opgave i rest consumer I C#](#)
- Hav 'PyCharm' installeret (se installations guide)
- Hav Wireshark installeret
- Dit gamle REST-API

Opgave 1.1: Lav et nyt Python - projekt

Lav et nyt python-projekt og tilføj en 'RestClient.py' – fil til projektet, hvor REST metoderne skal ligge.

Hvis ikke du har installeret requests pakken – så gør det nu:

Du skal åbne Terminal – fanen i bunden af PyCharm og skrive:

```
python -m pip install requests
```

Opgave 1.2: Lav CRUD-metoder i RestClient'en

Du skal i RestClient.py :

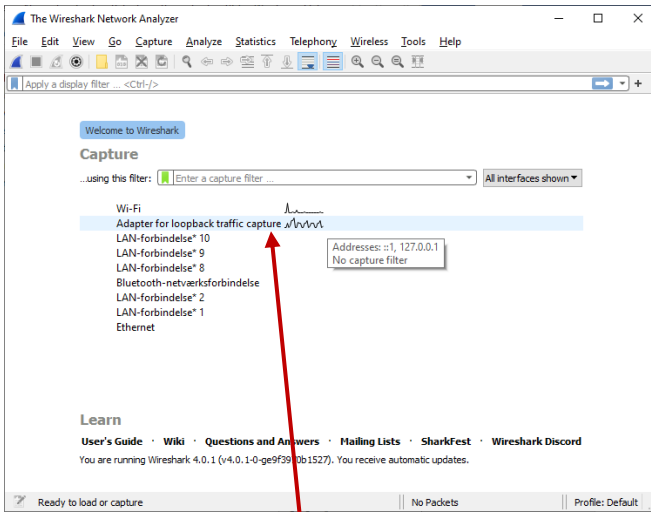
- importere requests – modulet (du installerede i opgave 1.1)
- (ligesom ved C#) definere en konstant (variable) med URL'en til din REST-Service.
dvs: URL = 'http://...'
- definere 5 metoder (GetAll, GetById, Add, Update og Delete).

Opgave 2.1: Brug din RestClient i main.py

Du skal importere din RestClient og prøve at kalde de forskellige metoder.

Opgave 2.2: Prøv din applikation

Du skal starte Wireshark – til at opfange din kommunikation med REST-servicesen:

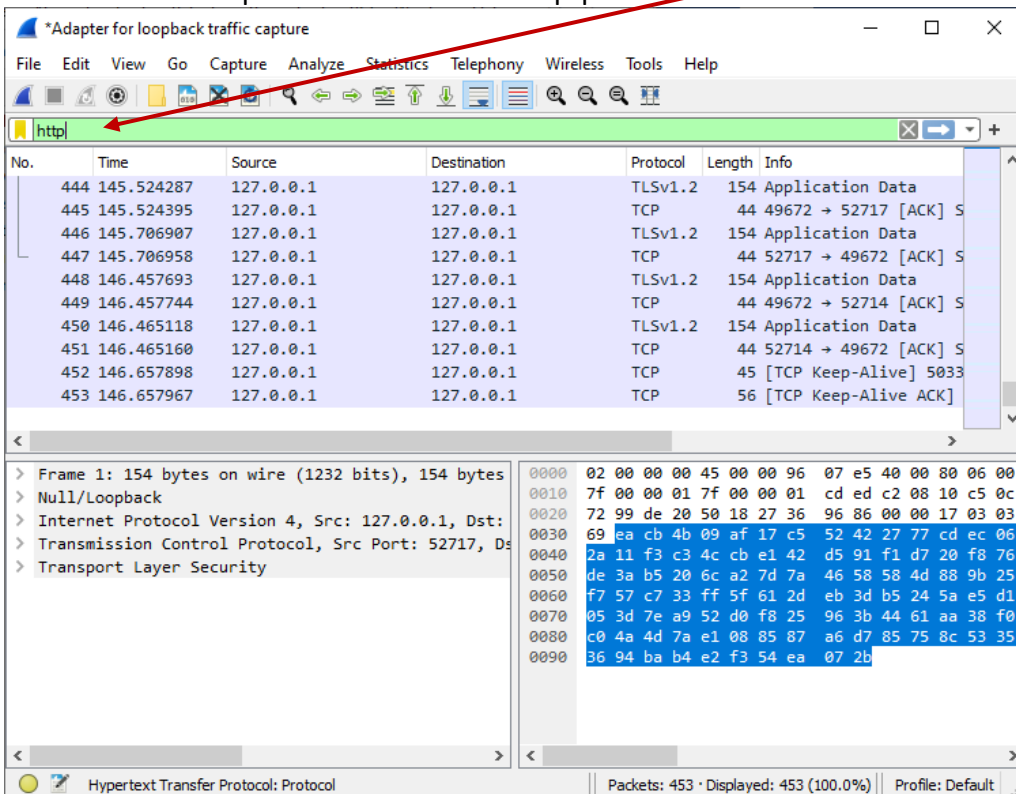


Klik på adapter for loopback til at starte packet-snifferen (programmet).

Kør dit Python program.

Hvis det gik godt stop wireshark (den røde firkant øverst til venstre)

Filtrer de sniffede pakker så du kun ser http-pakker:



Undersøg de forskellige http-request og http-responses

Opgave 2.3: Prøv med din REST-Service i Azure

Sørg for at din REST-service er publish'ede i Azure.

Du skal nu ændre din RestClient så den benytter REST-tjenesten i Azure i stedet for den på din lokale PC.