Padding Oracle Attack

The goal of this exercise is to get familiar with the MATLAB environment[1] and to develop code for the last-byte and last-block padding oracle attacks. The provided code (`padding oracle attack exercise.zip`) simulates a very simple server that can encrypt and decrypt using the DES cipher in CBC mode of operation. The server CBC encryption and decryption is specified in the files `server_encrypt.m`, `server_decrypt.m` and `DES.m`.

- Starting at the `main.m` file, the attacker intercepts an initialization vector IV, together with ciphertexts C_1 and C_2
- The attacker's goal is to decrypt ciphertext C_2 and recover the respective plaintext P_2
- The attacker has oracle access to this simple CBC decryption server. Thus the attacker can send the IV together with C_1 and C_2 and observe if there is any padding error or not. The attacker's oracle access is specified in the file `oracle.m`.

- ▶ Write code in the file `main.m` that recovers the last byte of P_2 i.e. code the last-byte oracle attack
- ▶ To develop the last-byte oracle attack you can call the `oracle.m` function from `main.m`, since the attacker has oracle access to the server. However, you cannot call the functions `server_encrypt.m`, `server_decrypt.m` or `DES.m`, since the server internals are not accessible. Still, try to explore them to get more familiar with the MATLAB coding language and environment.
- ▶ Optional: Write code that recovers the full plaintext block P_2 i.e. code the last-block oracle attack. Naturally, to develop the last-block oracle attack you can again call the `oracle.m` function from `main.m`.
- ▶ Optional: Perform the extra check that finds the exact value of the correct padding.

Deliverables: Submit the `main.m` file

[1] check https://datanose.nl/#byod to use the UvA MATLAB licence