

## RX8 ECU Dump Tool in Python

Ce guide ce veut être simplement une aide a l'utilisation de l'outil de DUMP en python trouvable ici:  
<https://github.com/xcorbel/RX8-E85-Vtune/tree/main/DumpTools>

Version Exécutable (Windows Only) :

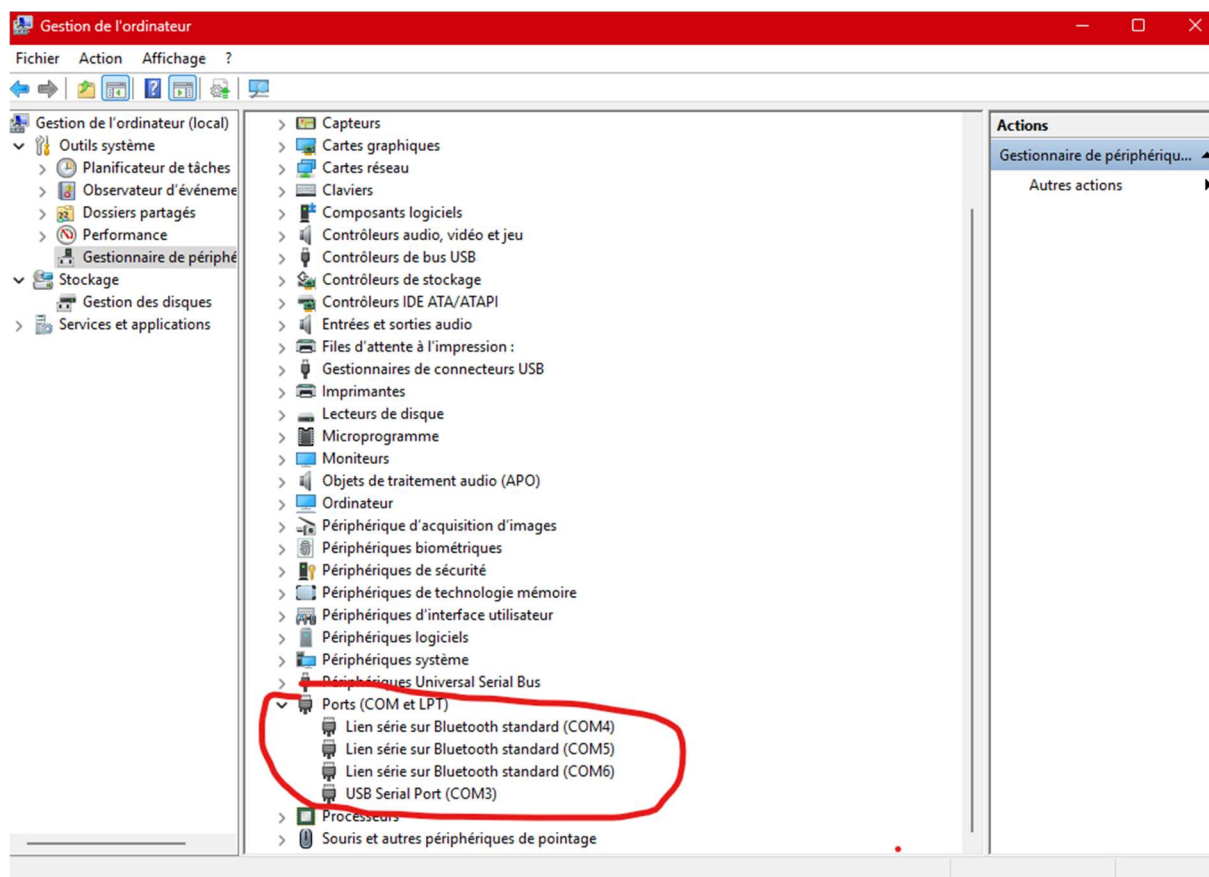
Etape 1 :

Créer un répertoire et télécharger le script de dump et le copier dans le répertoire :

lien : <https://github.com/xcorbel/RX8-E85-Vtune/raw/refs/heads/main/DumpTools/xco-rx8dump.exe>

Etape2 :

Connecter votre câble ou ELM en BT à votre PC et identifier le port COM associé dans le gestionnaire de périphérique.



Ouvrir une fenêtre de ligne de commande et se déplacer dans le répertoire où le script se situe dans l'exemple 'C:\LOW\RX8\_Python\_ECU'

puis exécuter le script en précisant le port COM en argument COM3 dans l'exemple

```
C:\LOW\RX8_Python_ECU>xco-rx8dump.exe COM3
*****
Mazda RX-8 ROM Dump
*****

Mazda ROM dumper
Connecting with COM3...
ELM327 v1.4b @ 500000 bps

Connecting to vehicle...
ERROR: No response from vehicle (is ignition on?)
Traceback (most recent call last):
  File "xco-rx8dump.py", line 397, in <module>
    args = parser.parse_args()
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "xco-rx8dump.py", line 100, in dump_rom
    # response = sendData("220000")
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "xco-rx8dump.py", line 163, in sendData
    def sendData(ser, string, end=">", blocking=True):
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "xco-rx8dump.py", line 228, in receiveResponse
    response = buffer.splitlines()
    ^^^^^
NameError: name 'exit' is not defined
[PYI-20468:ERROR] Failed to execute script 'xco-rx8dump' due to unhandled exception!

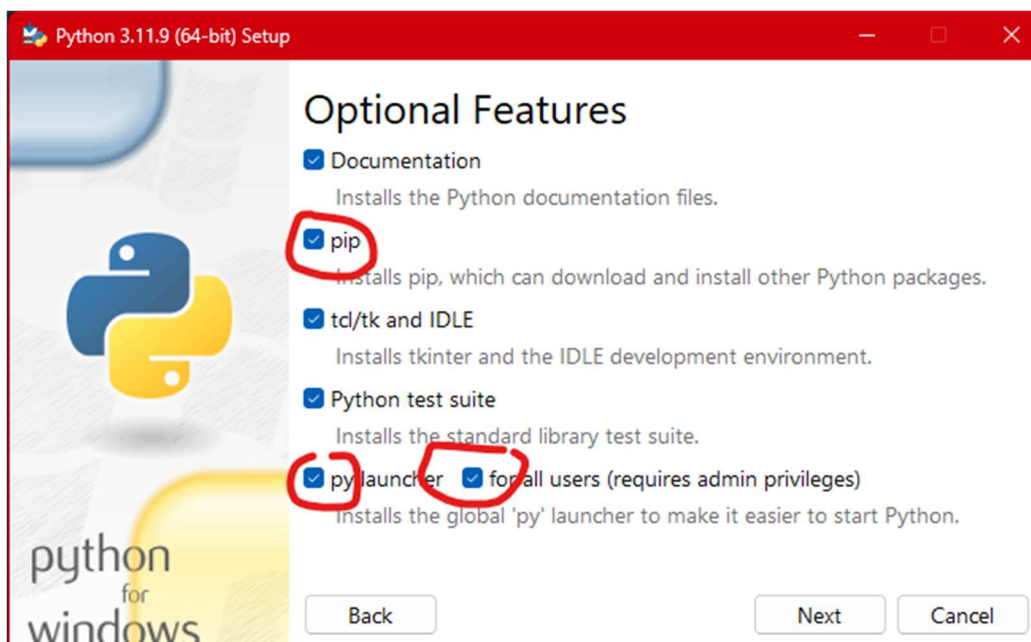
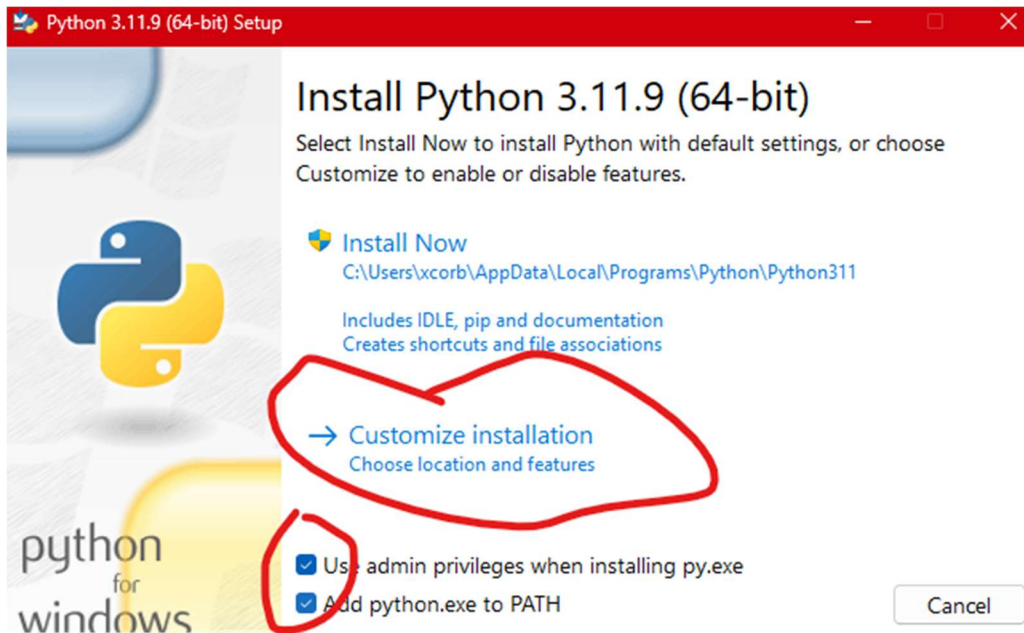
C:\LOW\RX8_Python_ECU>
```

Version Python :

Etape 1 : installation de python (testé avec python 3.11) :

-télécharger une version de python ici : <https://www.python.org/downloads/>

-lancé l'installation de python en mettant python dans le chemin système et avec l'installation de PIP.

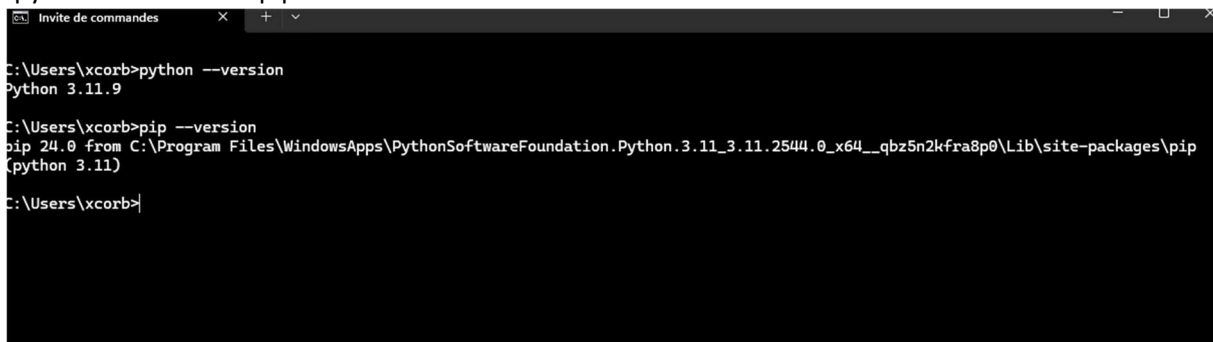


Finaliser l'installation

## Etape2 : Vérification de python et installation des dépendances

Après installation s'assurer que le programme a été bien installé en ouvrant une fenêtre de ligne de commande et exécuter les commandes suivantes :

'python --version' et 'pip --version'

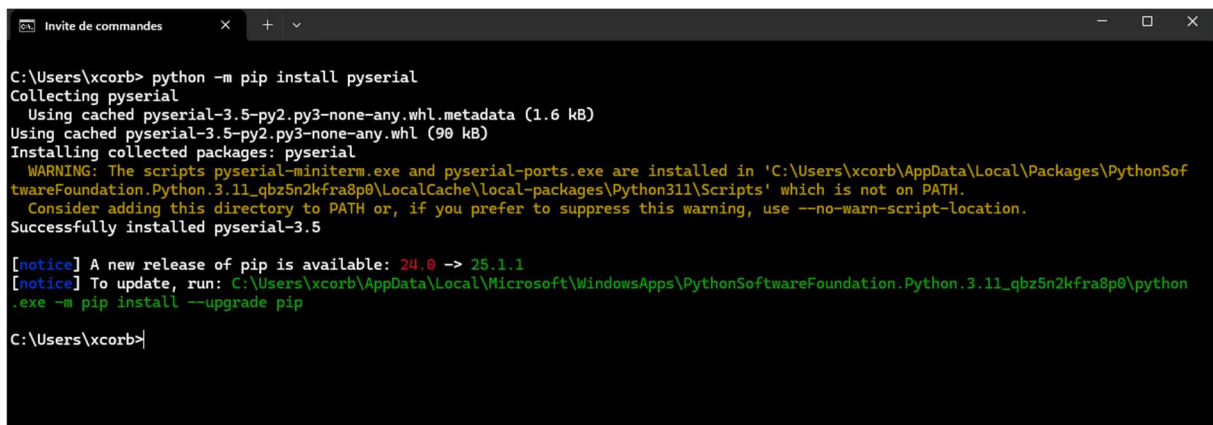


```
Invite de commandes
C:\Users\xcorb>python --version
Python 3.11.9

C:\Users\xcorb>pip --version
pip 24.0 from C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.2544.0_x64__qbz5n2kfra8p0\Lib\site-packages\pip
(python 3.11)

C:\Users\xcorb>
```

Installer pyserial via pip en utilisant la commande 'python -m pip install pyserial'



```
Invite de commandes
C:\Users\xcorb> python -m pip install pyserial
Collecting pyserial
  Using cached pyserial-3.5-py2.py3-none-any.whl.metadata (1.6 kB)
  Using cached pyserial-3.5-py2.py3-none-any.whl (90 kB)
Installing collected packages: pyserial
  WARNING: The scripts pyserial-miniterm.exe and pyserial-ports.exe are installed in 'C:\Users\xcorb\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed pyserial-3.5

[notice] A new release of pip is available: 24.0 -> 25.1.1
[notice] To update, run: C:\Users\xcorb\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\python
.exe -m pip install --upgrade pip

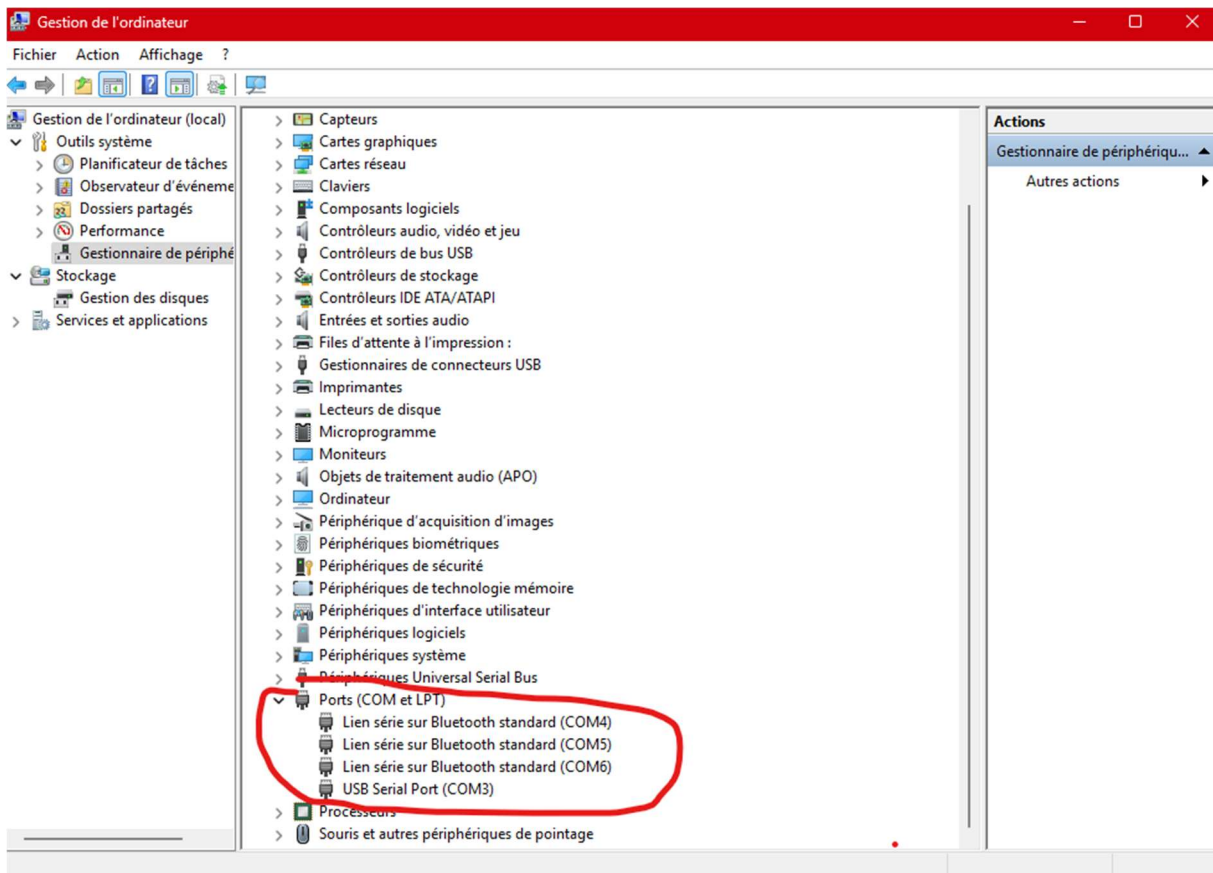
C:\Users\xcorb>
```

Créer un répertoire et télécharger le script de dump et le copier dans le répertoire :

lien : <https://github.com/xcorbel/RX8-E85-Vtune/raw/refs/heads/main/DumpTools/xco-rx8dump.py>

### 3 : Utilisation de l'outil :

-connecter votre câble ou ELM en BT à votre PC et identifier le port COM associé dans le gestionnaire de périphérique.



Ouvrir une fenêtre de ligne de commande et se déplacer dans le répertoire où le script se situe dans l'exemple 'C:\LOW\RX8\_Python\_ECU'

- cd C:\LOW\RX8\_Python\_ECU

puis exécuter le script en précisant le port COM en argument COM3 dans l'exemple

- python xco-rx8dump.py COM3

```
Invite de commandes
C:\>cd c:\LOW\RX8_Python_ECU
c:\LOW\RX8_Python_ECU>python xco-rx8dump.py COM3
*****
Mazda RX-8 ROM Dump
*****
Mazda ROM dumper
Connecting with COM3...
ELM327 v1.4b @ 500000 bps
Connecting to vehicle...
ERROR: No response from vehicle (is ignition on?)
c:\LOW\RX8_Python_ECU>
```

Débogage :

l'outil utilise le code Stock de Mazda qui est 'MazdA' afin de déverrouiller le PCM.  
ce code est en ascii HEX dans le script Python et doit être passer de la même façon en cas de tentative de dump avec une clé autre que Stock :

« Default\_KEY = "4d 61 7a 64 41" # MazdA »

Voici un test avec un clé erronée

```
C:\LOW\RX8_Python_ECU>python xco-rx8dump.py COM4 [REDACTED]

*****
Mazda RX-8 ROM Dump
*****

Mazda ROM dumper
Connecting with COM4...
ELM327 v2.1 @ 38400 bps

Connecting to vehicle...
Reading VIN...
VIN: ['[REDACTED]', '', '>']
Acquiring securityAccess...
Traceback (most recent call last):
  File "C:\LOW\RX8_Python_ECU\xco-rx8dump.py", line 378, in <module>
    dump_rom(args.port, args.baud, secret, args.output)
  File "C:\LOW\RX8_Python_ECU\xco-rx8dump.py", line 116, in dump_rom
    acquireSecurityAccess(ser, secret)
  File "C:\LOW\RX8_Python_ECU\xco-rx8dump.py", line 298, in acquireSecurityAccess
    raise RuntimeError(f"Unable to access ROM with secret {secret}")
RuntimeError: Unable to access ROM with secret [REDACTED]
```