МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №1

з дисципліни
«МЕТОДИ РЕАЛІЗАЦІЇ КРИПТОГРАФІЧНИХ МЕХАНІЗМІВ»
на тему: «ВИБІР БІБЛІОТЕКИ РЕАЛІЗАЦІЇ ОСНОВНИХ КРИПТОГРАФІЧНИХ ПРИМІТИВІВ З ТОЧКИ ЗОРУ ЇХ ЕФЕКТИВНОСТІ ЗА ЧАСОМ ТА ПАМ'ЯТТЮ ДЛЯ РІЗНИХ ПРОГРАМНИХ ПЛАТФОРМ»

Виконала:

студентка 6 курсу ФТІ
група ФБ-11мн
Чуракова Єкатєріна

Перевірив:
Байденко П.В.

Київ 2022

Варіант 2А: ПОРІВНЯННЯ БІБЛІОТЕК OPENSSL, CRYPTO++, CRYPTOLIB, PYCRYPTO ДЛЯ РОЗРОБКИ ГІБРІДНОЇ КРИПТОСИСТЕМИ ПІД WINDOWS ПЛАТФОРМУ

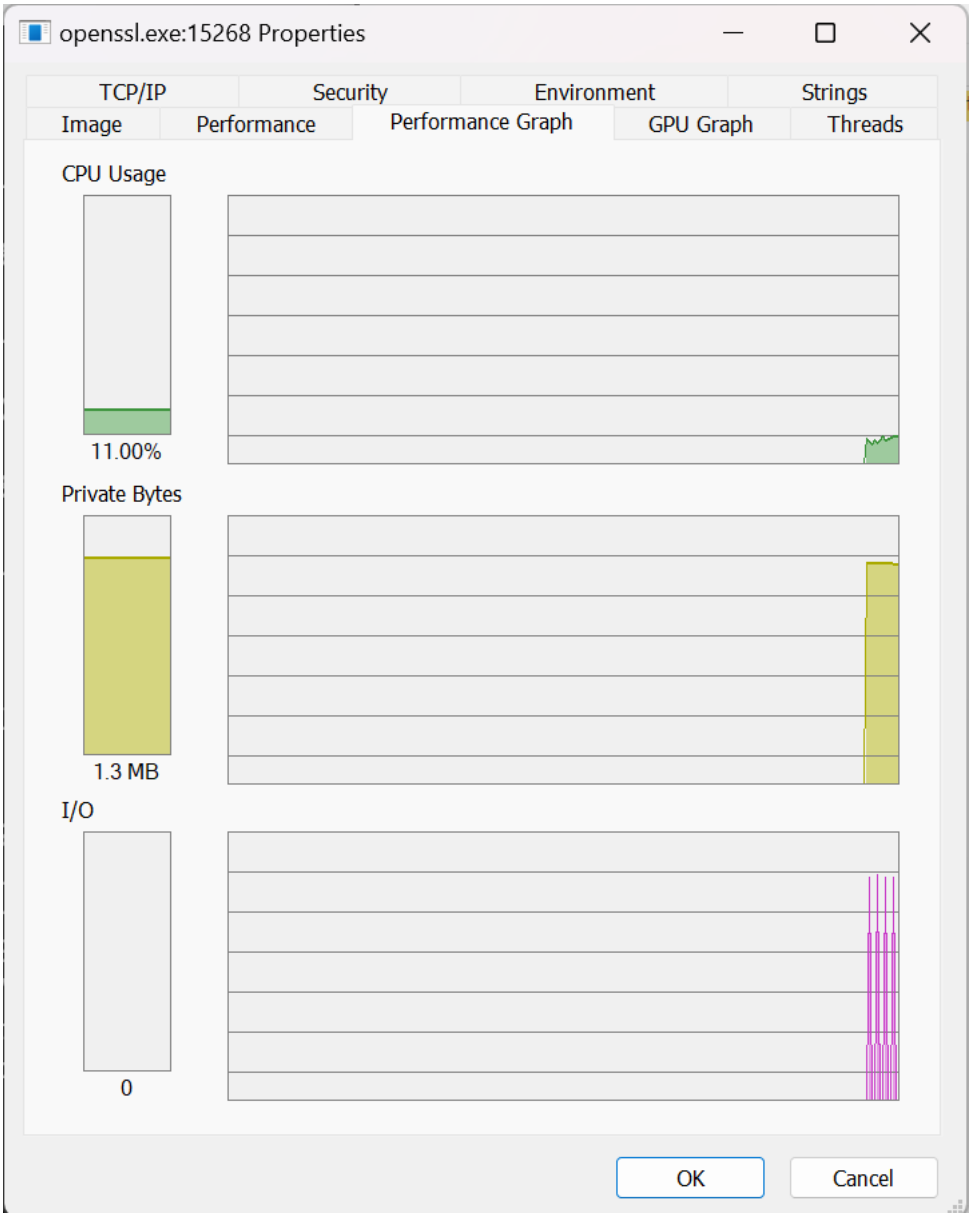
Процесор, на якому проводилось тестування: AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx

```
PS C:\Users\KDolp> wmic path win32_Processor get Name,NumberOfCores,NumberOfLogicalProcessors
Name                                  NumberOfCores  NumberOfLogicalProcessors
AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx  4              8
```

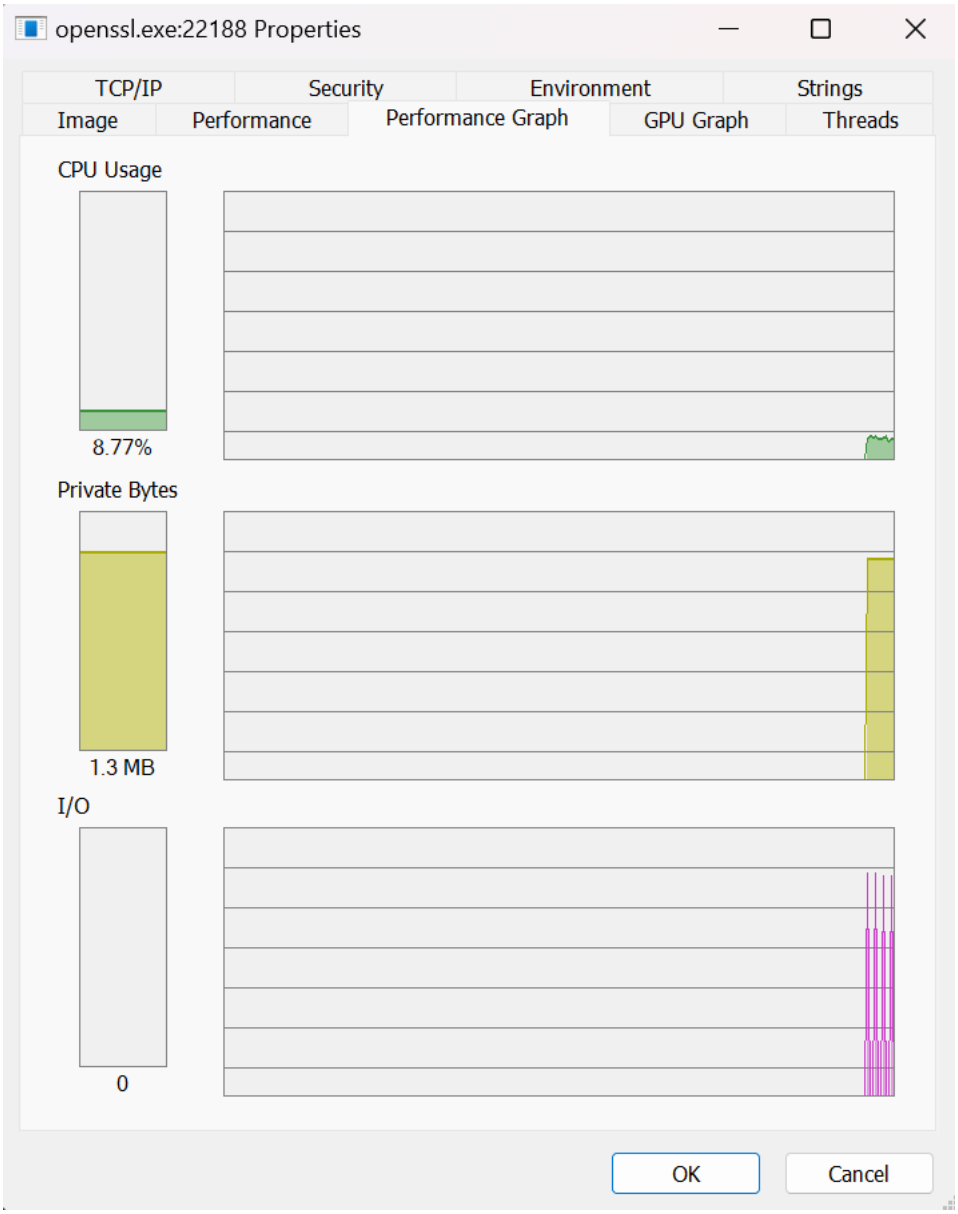
Для тестування були обрані алгоритми AES-256-CBC, AES-128-CBC, AES-256-GCM, AES-128-GCM (швидкість операцій шифрування і розшифрування на різних довжинах вхідних даних) і асиметричний алгоритм RSA (швидкість підпису). Для моніторингу стану ресурсів комп'ютера використовувалася утиліта ProcessExplorer.

OpenSSL

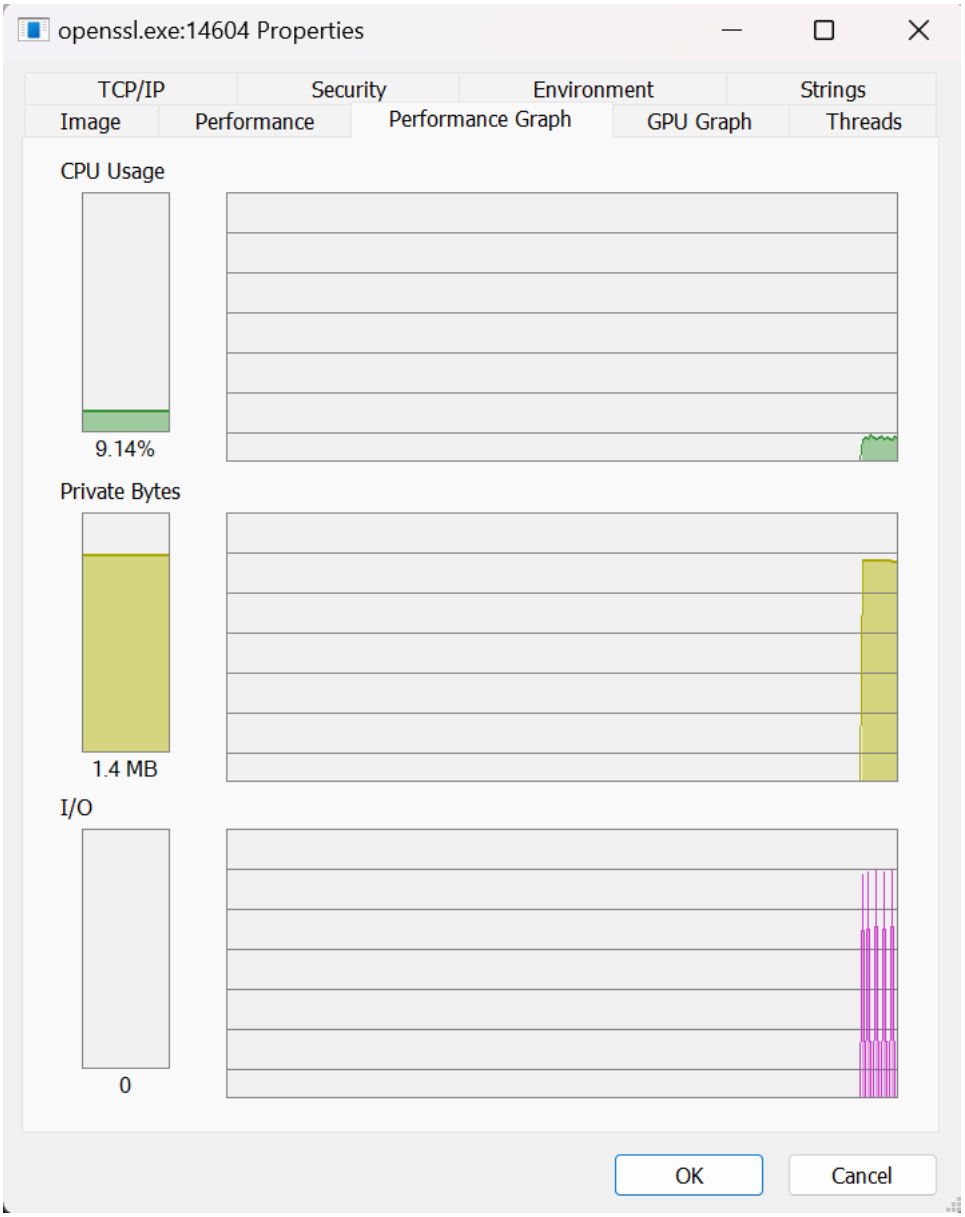
```
PS C:\Users\KDolp> openssl speed aes-256-cbc
Doing aes-256 cbc for 3s on 16 size blocks: 20379221 aes-256 cbc's in 2.14s
Doing aes-256 cbc for 3s on 64 size blocks: 5067774 aes-256 cbc's in 1.52s
Doing aes-256 cbc for 3s on 256 size blocks: 1455264 aes-256 cbc's in 1.64s
Doing aes-256 cbc for 3s on 1024 size blocks: 338096 aes-256 cbc's in 1.16s
Doing aes-256 cbc for 3s on 8192 size blocks: 45523 aes-256 cbc's in 1.84s
Doing aes-256 cbc for 3s on 16384 size blocks: 24370 aes-256 cbc's in 1.98s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes     64 bytes    256 bytes   1024 bytes   8192 bytes  16384 bytes
aes-256 cbc     152323.52k   213995.90k   227076.62k   299425.13k   202264.09k   201211.00k
```



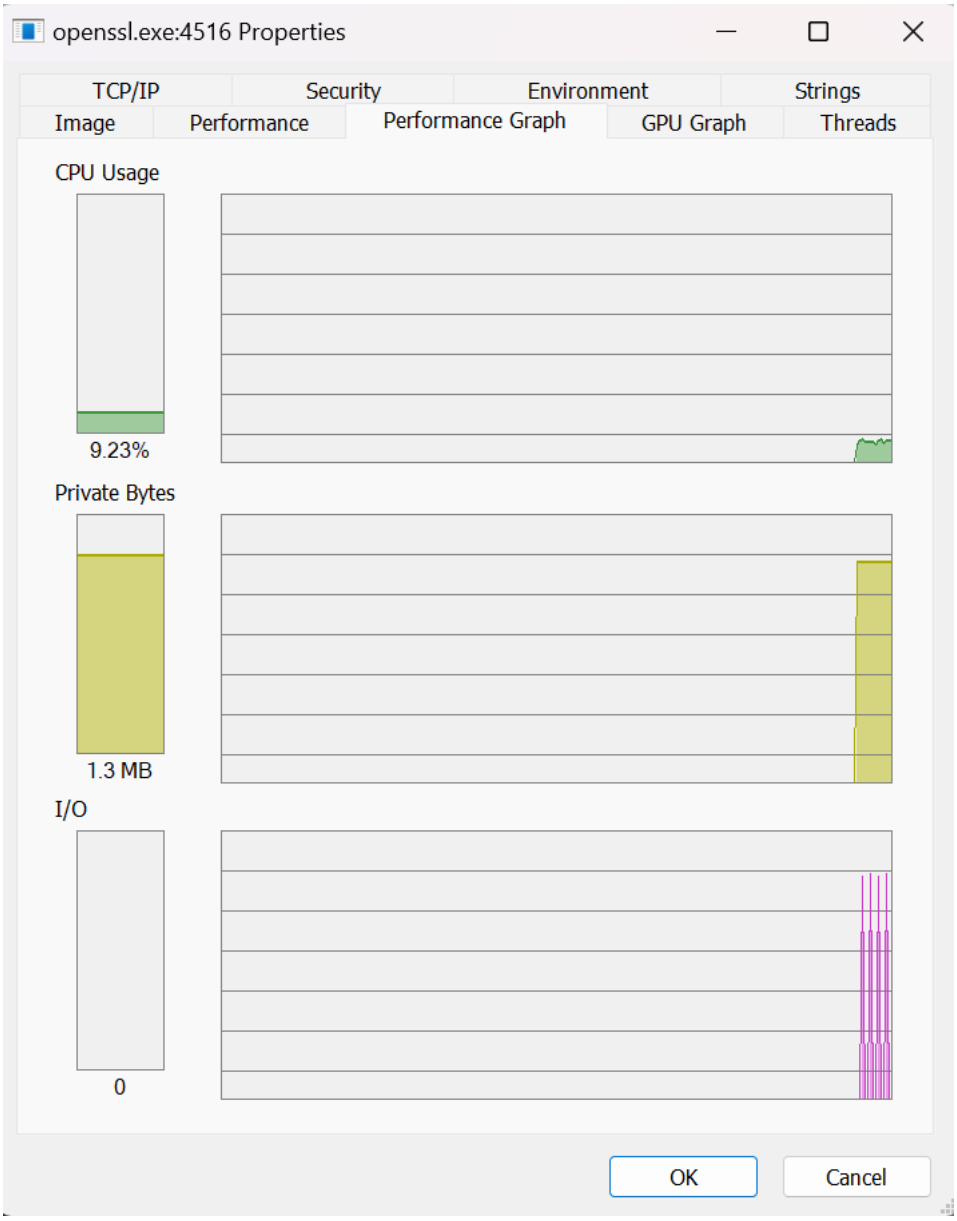

```
PS C:\Users\KDolp> openssl speed aes-128-cbc
Doing aes-128 cbc for 3s on 16 size blocks: 29348040 aes-128 cbc's in 1.78s
Doing aes-128 cbc for 3s on 64 size blocks: 8014195 aes-128 cbc's in 1.89s
Doing aes-128 cbc for 3s on 256 size blocks: 2040577 aes-128 cbc's in 1.73s
Doing aes-128 cbc for 3s on 1024 size blocks: 514368 aes-128 cbc's in 1.70s
Doing aes-128 cbc for 3s on 8192 size blocks: 64370 aes-128 cbc's in 1.77s
Doing aes-128 cbc for 3s on 16384 size blocks: 32045 aes-128 cbc's in 1.63s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes     64 bytes    256 bytes   1024 bytes   8192 bytes  16384 bytes
aes-128 cbc     263617.48k   271290.44k   301196.52k   309262.58k   298658.57k   323092.48k
```



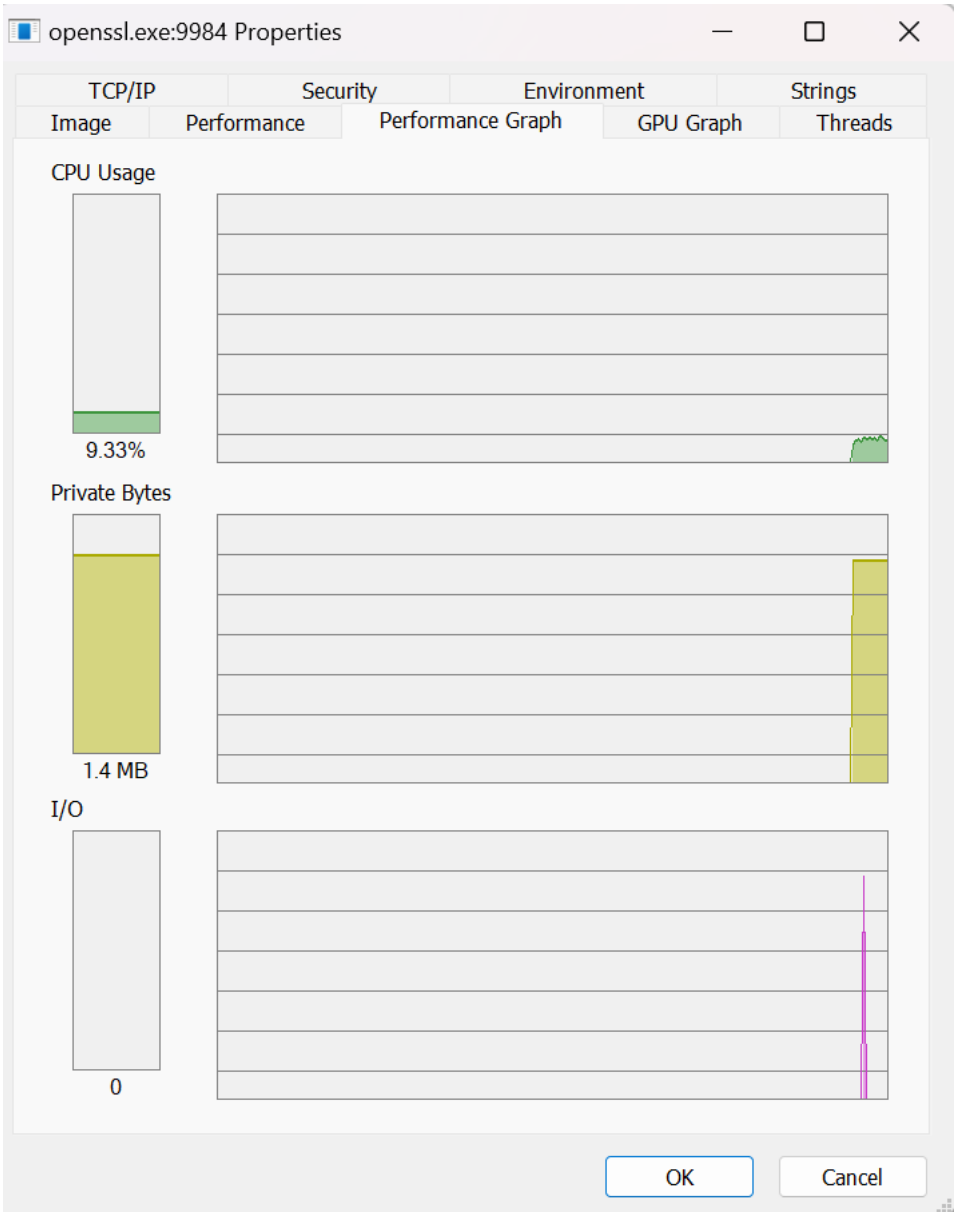

```
PS C:\Users\KDolp> openssl speed -evp aes-256-gcm
Doing aes-256-gcm for 3s on 16 size blocks: 33809721 aes-256-gcm's in 1.69s
Doing aes-256-gcm for 3s on 64 size blocks: 20837098 aes-256-gcm's in 1.61s
Doing aes-256-gcm for 3s on 256 size blocks: 11324599 aes-256-gcm's in 0.95s
Doing aes-256-gcm for 3s on 1024 size blocks: 4601893 aes-256-gcm's in 1.16s
Doing aes-256-gcm for 3s on 8192 size blocks: 857270 aes-256-gcm's in 1.70s
Doing aes-256-gcm for 3s on 16384 size blocks: 414111 aes-256-gcm's in 1.88s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes     64 bytes    256 bytes   1024 bytes   8192 bytes  16384 bytes
aes-256-gcm     320566.24k   828628.67k  3041675.90k  4075535.94k  4123452.97k  3618557.13k
```



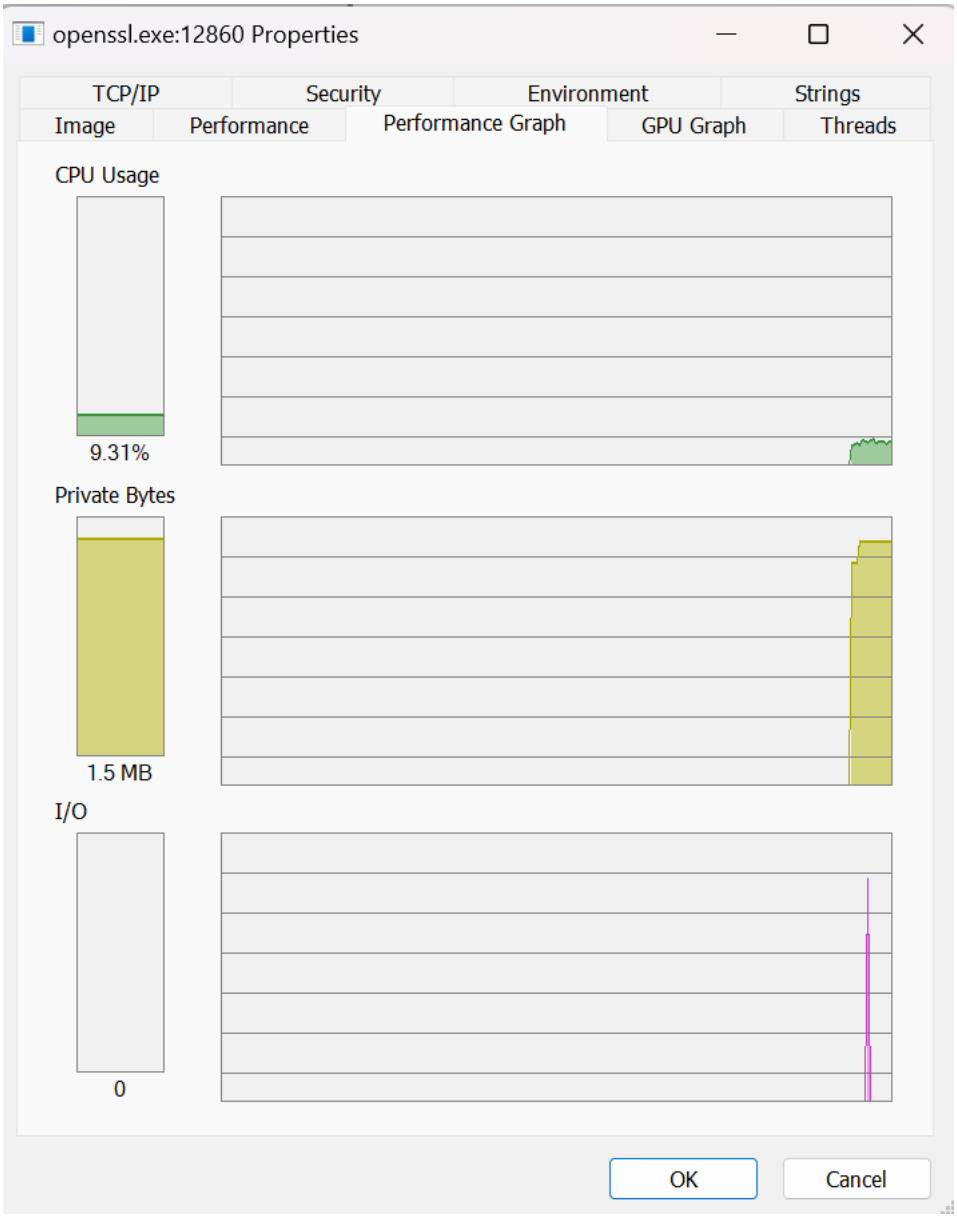

```
PS C:\Users\kDolp> openssl speed -evp aes-128-gcm
Doing aes-128-gcm for 3s on 16 size blocks: 34923544 aes-128-gcm's in 1.75s
Doing aes-128-gcm for 3s on 64 size blocks: 22075710 aes-128-gcm's in 1.88s
Doing aes-128-gcm for 3s on 256 size blocks: 14288536 aes-128-gcm's in 1.55s
Doing aes-128-gcm for 3s on 1024 size blocks: 5938535 aes-128-gcm's in 1.78s
Doing aes-128-gcm for 3s on 8192 size blocks: 957677 aes-128-gcm's in 1.84s
Doing aes-128-gcm for 3s on 16384 size blocks: 481976 aes-128-gcm's in 1.83s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes     64 bytes    256 bytes   1024 bytes   8192 bytes  16384 bytes
aes-128-gcm     319300.97k   753517.57k  2364680.54k  3413928.33k  4255072.53k  4319559.54k
```




```
PS C:\Users\KDolp> openssl speed rsa2048
Doing 2048 bits private rsa's for 10s: 7934 2048 bits private RSA's in 4.34s
Doing 2048 bits public rsa's for 10s: 231602 2048 bits public RSA's in 4.08s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
                  sign    verify    sign/s verify/s
rsa 2048 bits 0.000547s 0.000018s   1826.5  56791.3
```

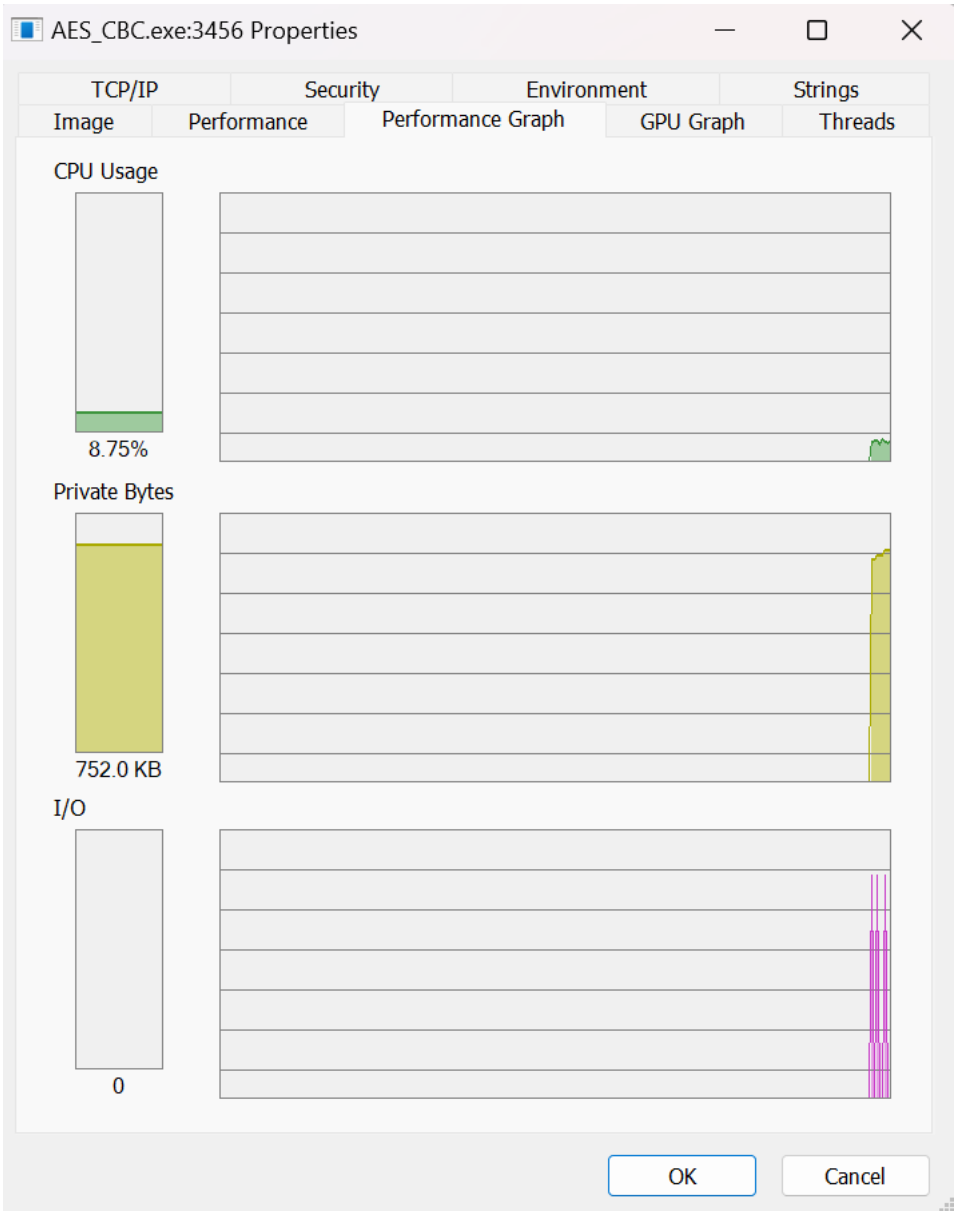



```
PS C:\Users\KDolp> openssl speed rsa4096
Doing 4096 bits private rsa's for 10s: 1119 4096 bits private RSA's in 5.98s
Doing 4096 bits public rsa's for 10s: 69736 4096 bits public RSA's in 6.14s
OpenSSL 1.1.1n  15 Mar 2022
built on: Mon Mar 21 08:22:10 2022 UTC
options:bn(64,64) rc4(8x,int) des(long) aes(partial) idea(int) blowfish(ptr)
compiler: cl.exe /Zi /Fdossl_static.pdb /Gs0 /GF /Gy /MD /W3 /wd4090 /nologo /O2 -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUI
D_OBJ -DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DS
HA512_ASM -DKECCAK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOL
Y1305_ASM
                  sign    verify    sign/s verify/s
rsa 4096 bits 0.005348s 0.000088s    187.0  11356.5
```

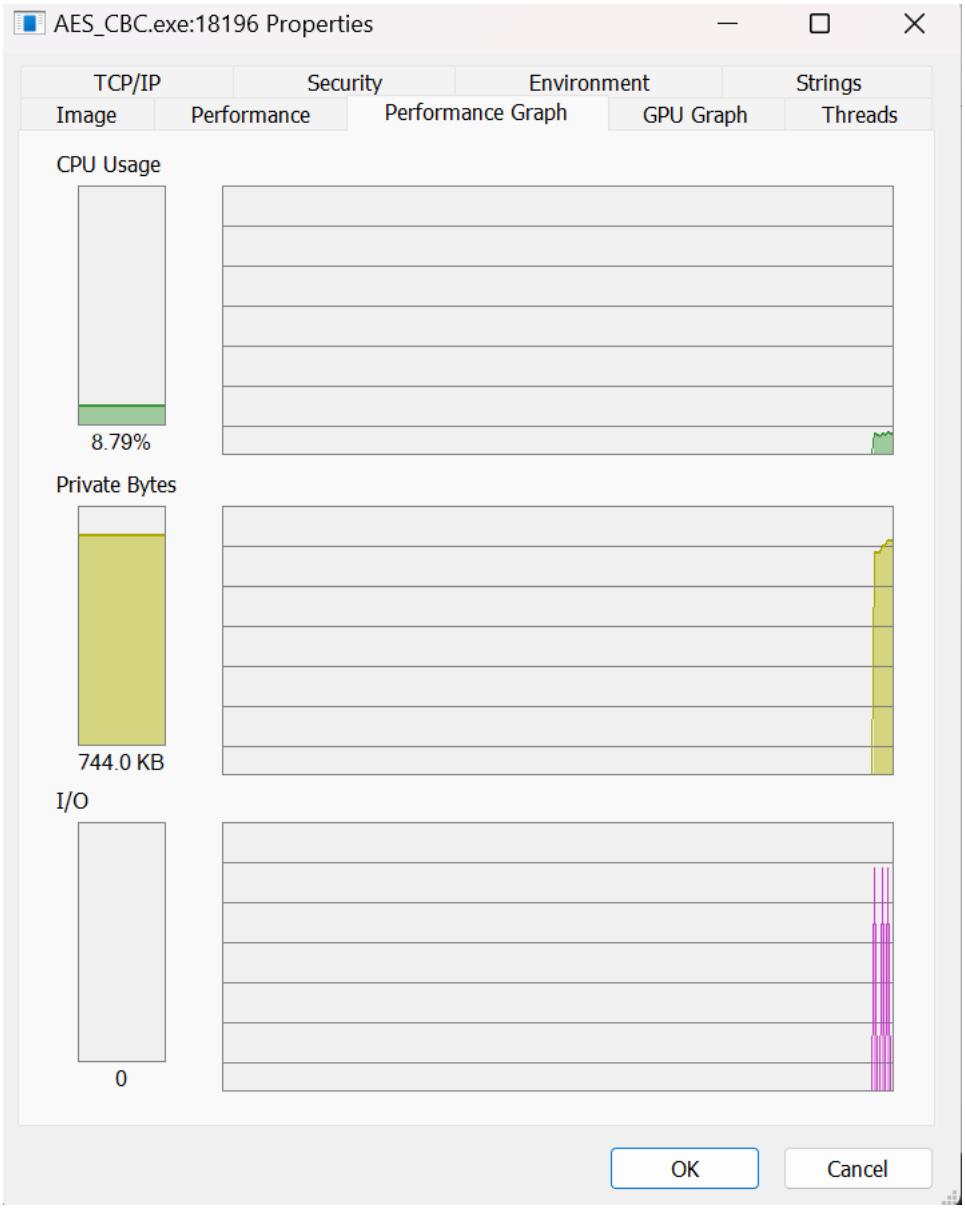

openssl.exe:12860 Properties
TCP/IP
Security
Environment
Strings
Image
Performance
Performance Graph
GPU Graph
Threads
CPU Usage
9.31%
Private Bytes
1.5 MB
I/O
0
OK
Cancel

Crypto++

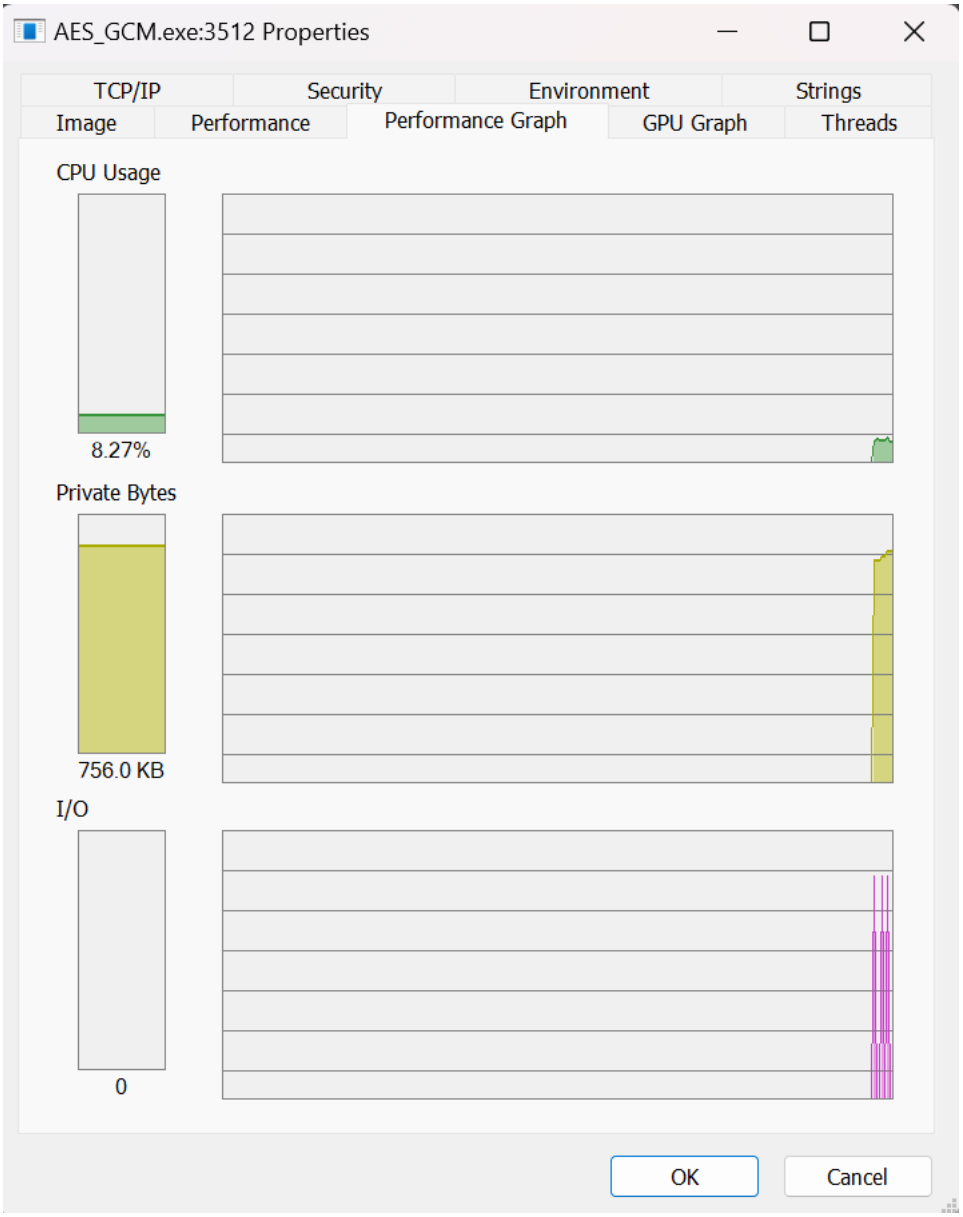
```
Doing aes-256 cbc for 3s on 16 size blocks: 1112270 aes-256 cbc in 3.00's
Doing aes-256 cbc for 3s on 64 size blocks: 1368830 aes-256 cbc in 3.00's
Doing aes-256 cbc for 3s on 256 size blocks: 1249428 aes-256 cbc in 3.00's
Doing aes-256 cbc for 3s on 1024 size blocks: 816338 aes-256 cbc in 3.00's
Doing aes-256 cbc for 3s on 8192 size blocks: 191896 aes-256 cbc in 3.00's
Doing aes-256 cbc for 3s on 16384 size blocks: 102824 aes-256 cbc in 3.00's
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes        64 bytes        256 bytes       1024 bytes      8192 bytes      16384 bytes
aes-256 cbc      5932.11k        29201.71k       106617.86k      278643.38k      524004.00k      561556.19k
```



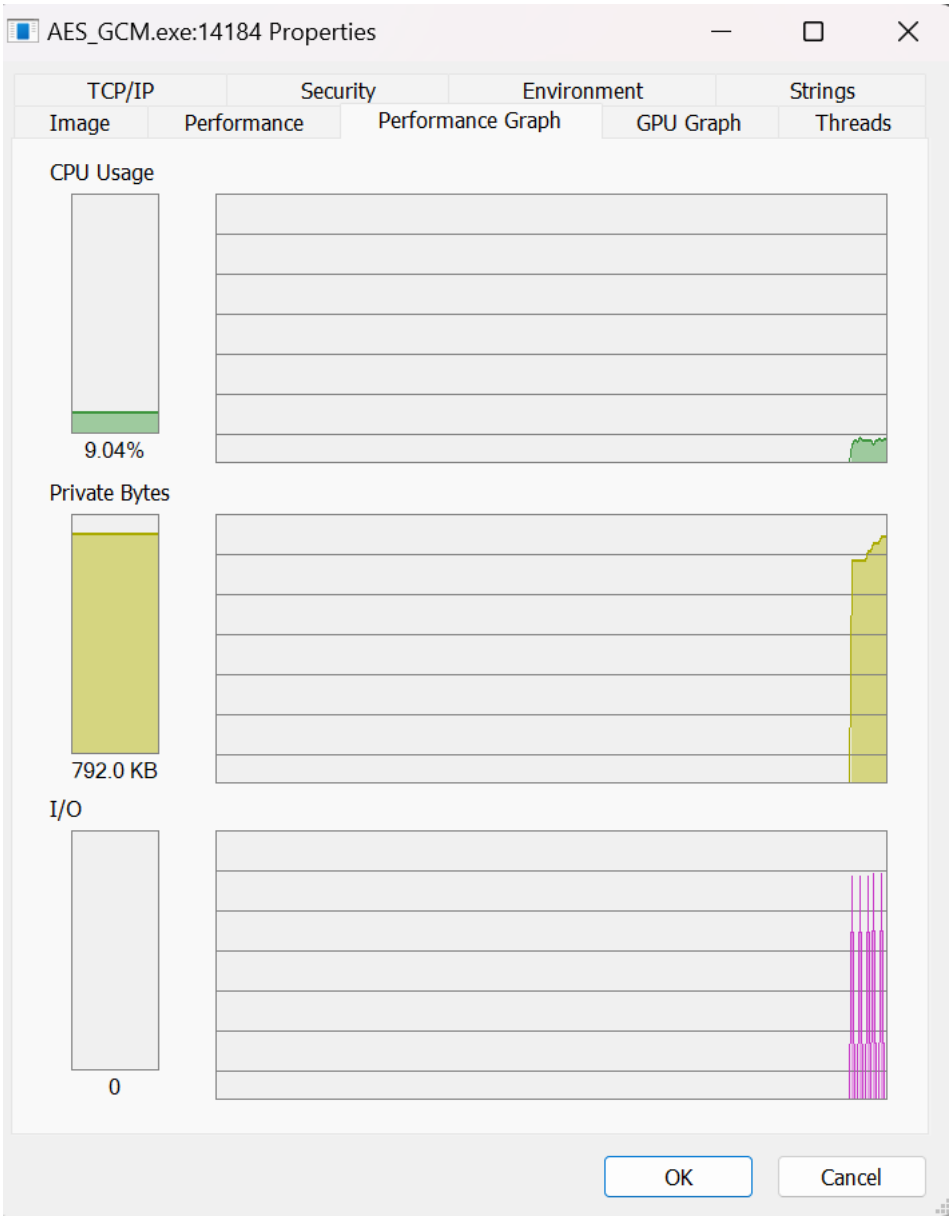

```
Doing aes-128 cbc for 3s on 16 size blocks: 1424182 aes-128 cbc in 3.00's
Doing aes-128 cbc for 3s on 64 size blocks: 1458251 aes-128 cbc in 3.00's
Doing aes-128 cbc for 3s on 256 size blocks: 1293686 aes-128 cbc in 3.00's
Doing aes-128 cbc for 3s on 1024 size blocks: 916394 aes-128 cbc in 3.00's
Doing aes-128 cbc for 3s on 8192 size blocks: 221652 aes-128 cbc in 3.00's
Doing aes-128 cbc for 3s on 16384 size blocks: 121056 aes-128 cbc in 3.00's
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes        64 bytes        256 bytes       1024 bytes      8192 bytes      16384 bytes
aes-128 cbc      7595.64k        31109.35k       110394.54k      312795.81k      605257.75k      661127.19k
```



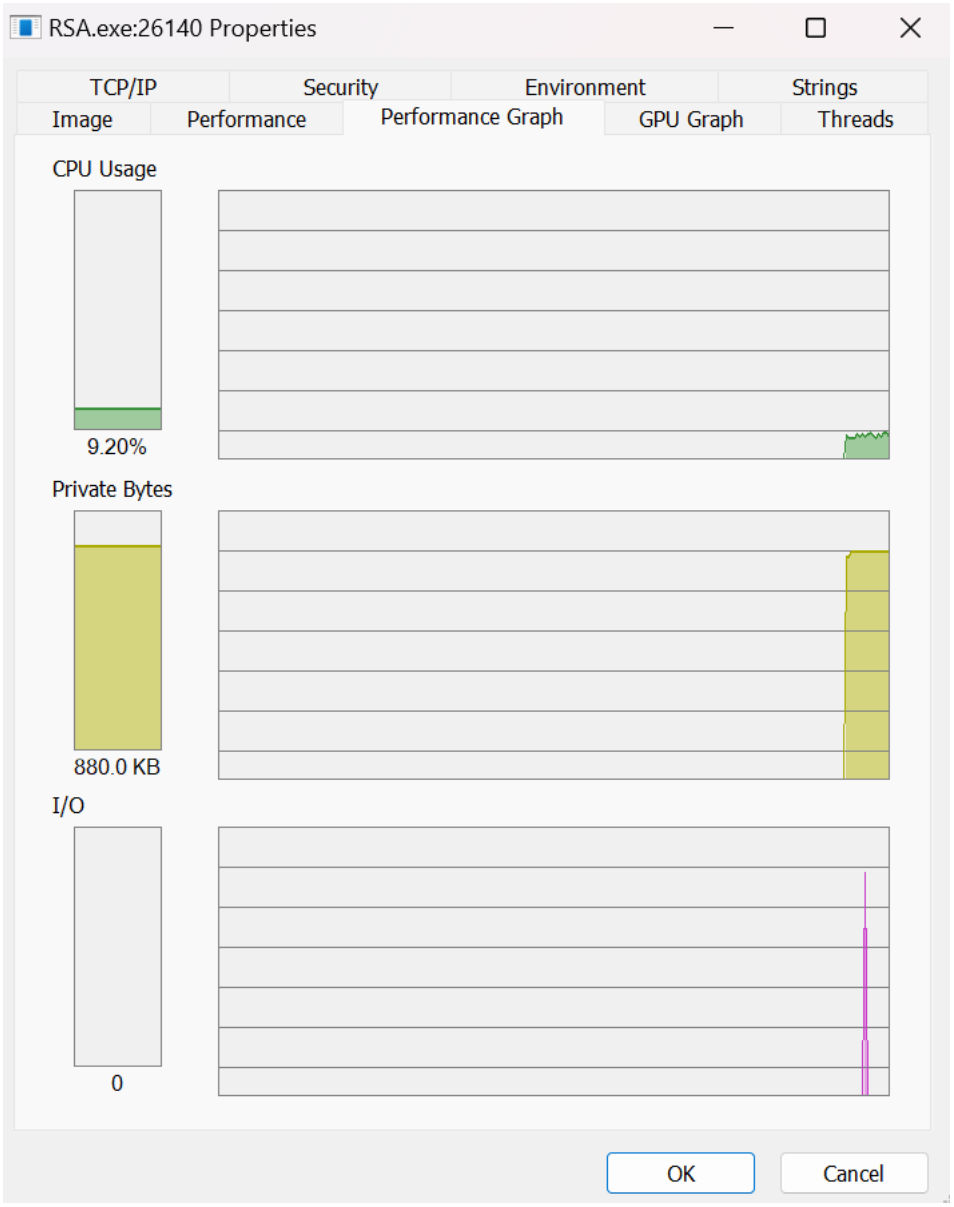

```
Doing aes-256 gcm for 3s on 16 size blocks: 902658 aes-256 gcm in 3.00's
Doing aes-256 gcm for 3s on 64 size blocks: 1152928 aes-256 gcm in 3.00's
Doing aes-256 gcm for 3s on 256 size blocks: 1105892 aes-256 gcm in 3.00's
Doing aes-256 gcm for 3s on 1024 size blocks: 862814 aes-256 gcm in 3.00's
Doing aes-256 gcm for 3s on 8192 size blocks: 282661 aes-256 gcm in 3.00's
Doing aes-256 gcm for 3s on 16384 size blocks: 157151 aes-256 gcm in 3.00's
The 'numbers' are in 1000s of bytes per second processed.
type                     16 bytes       64 bytes       256 bytes      1024 bytes     8192 bytes     16384 bytes
aes-256 gcm              4814.18k       24595.80k      94369.45k      294507.16k     771853.00k     858254.00k
```



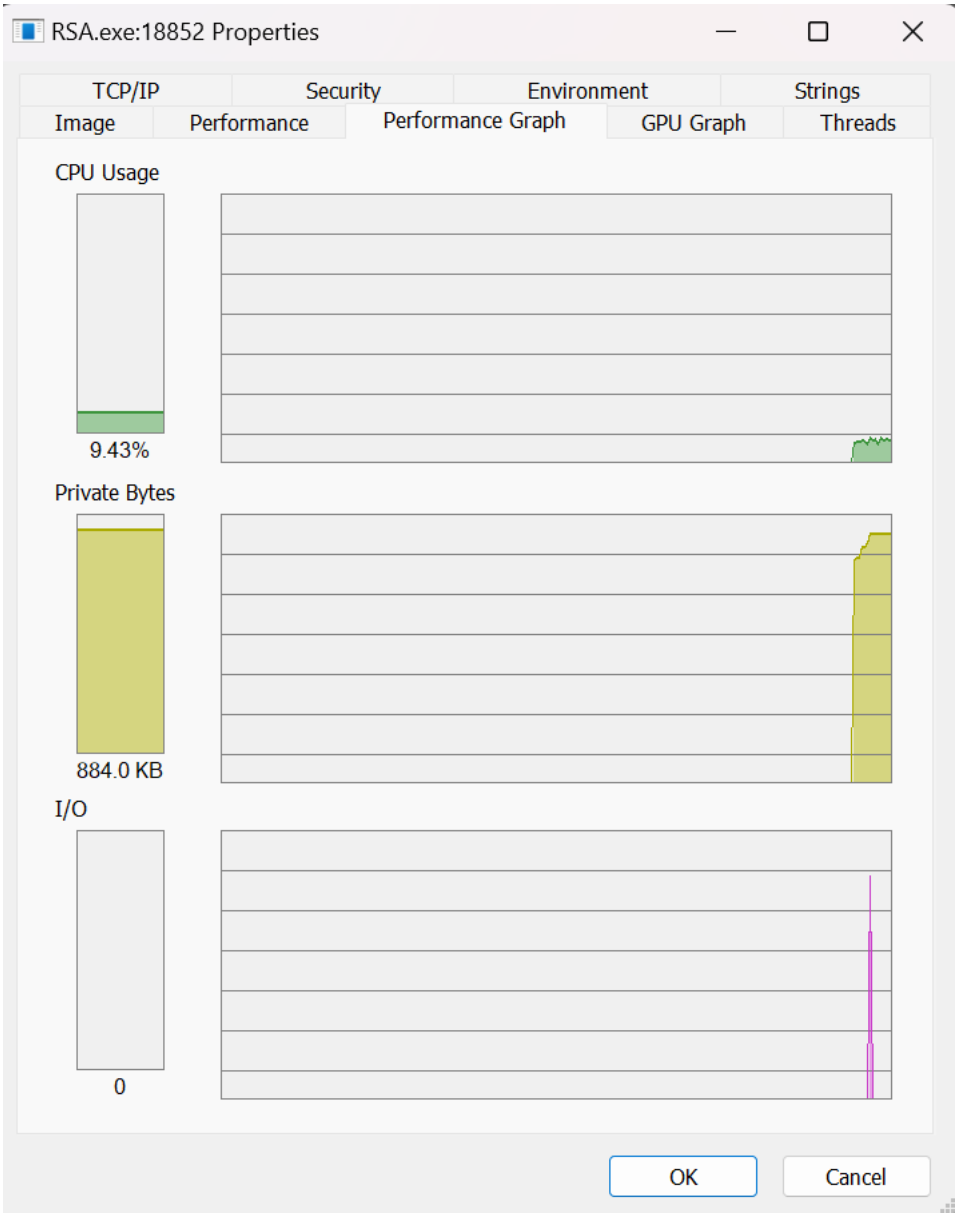

```
Doing aes-128 gcm for 3s on 16 size blocks: 905251 aes-128 gcm in 3.00's
Doing aes-128 gcm for 3s on 64 size blocks: 1146427 aes-128 gcm in 3.00's
Doing aes-128 gcm for 3s on 256 size blocks: 1094865 aes-128 gcm in 3.00's
Doing aes-128 gcm for 3s on 1024 size blocks: 884953 aes-128 gcm in 3.00's
Doing aes-128 gcm for 3s on 8192 size blocks: 339148 aes-128 gcm in 3.00's
Doing aes-128 gcm for 3s on 16384 size blocks: 200569 aes-128 gcm in 3.00's
The 'numbers' are in 1000s of bytes per second processed.
type                     16 bytes        64 bytes        256 bytes       1024 bytes      8192 bytes      16384 bytes
aes-128 gcm              4828.01k        24457.11k       93428.48k       302063.97k      926100.19k      1095374.25k
```




```
Doing 2048 bits private rsa's for 10s: 4828 2048 bits private RSA's in 10.00s
Doing 2048 bits public rsa's for 10s: 275623 2048 bits public RSA's in 10.00s
                  sign             verify          sign/s          verify/s
rsa 2048 bits   0.002071s       0.000036s       482.80          27562.300781
```

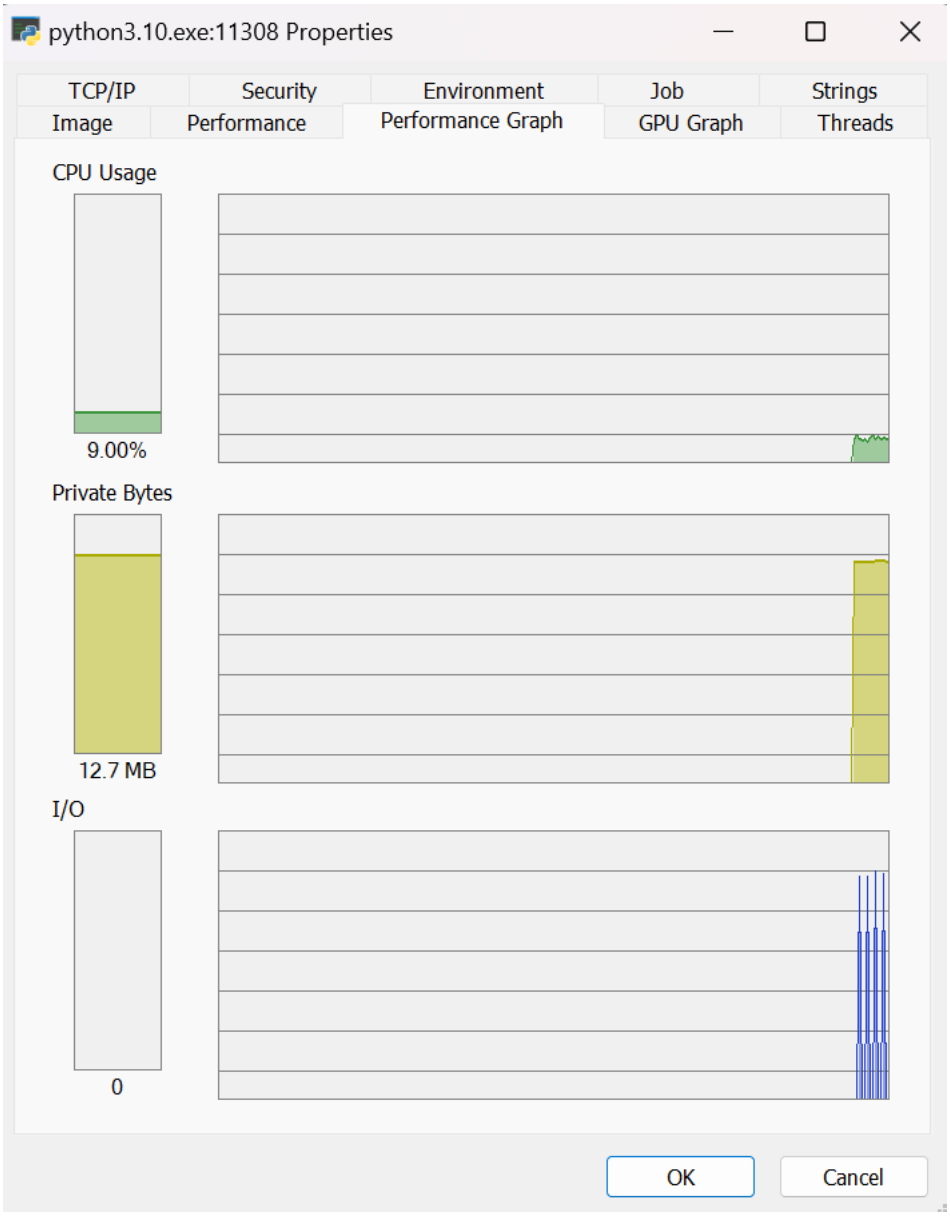



```
Doing 4096 bits private rsa's for 10s: 808 4096 bits private RSA's in 10.01s
Doing 4096 bits public rsa's for 10s: 103757 4096 bits public RSA's in 10.00s
                  sign            verify          sign/s          verify/s
rsa 4096 bits   0.012376s       0.000096s       80.80           10375.700195
```

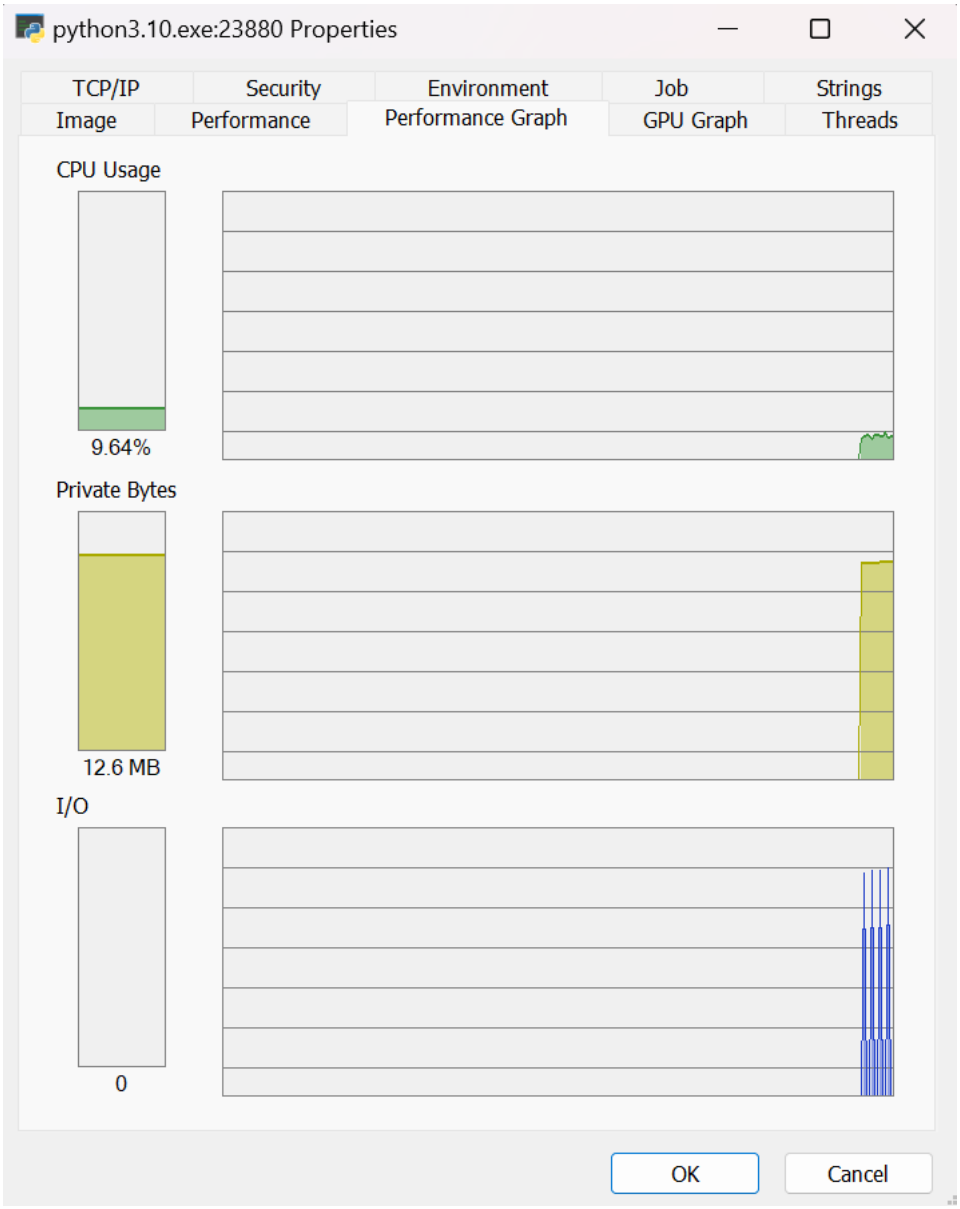

RSA.exe:18852 Properties
TCP/IP
Security
Environment
Strings
Image
Performance
Performance Graph
GPU Graph
Threads
CPU Usage
9.43%
Private Bytes
884.0 KB
I/O
0
OK
Cancel

PyCrypto

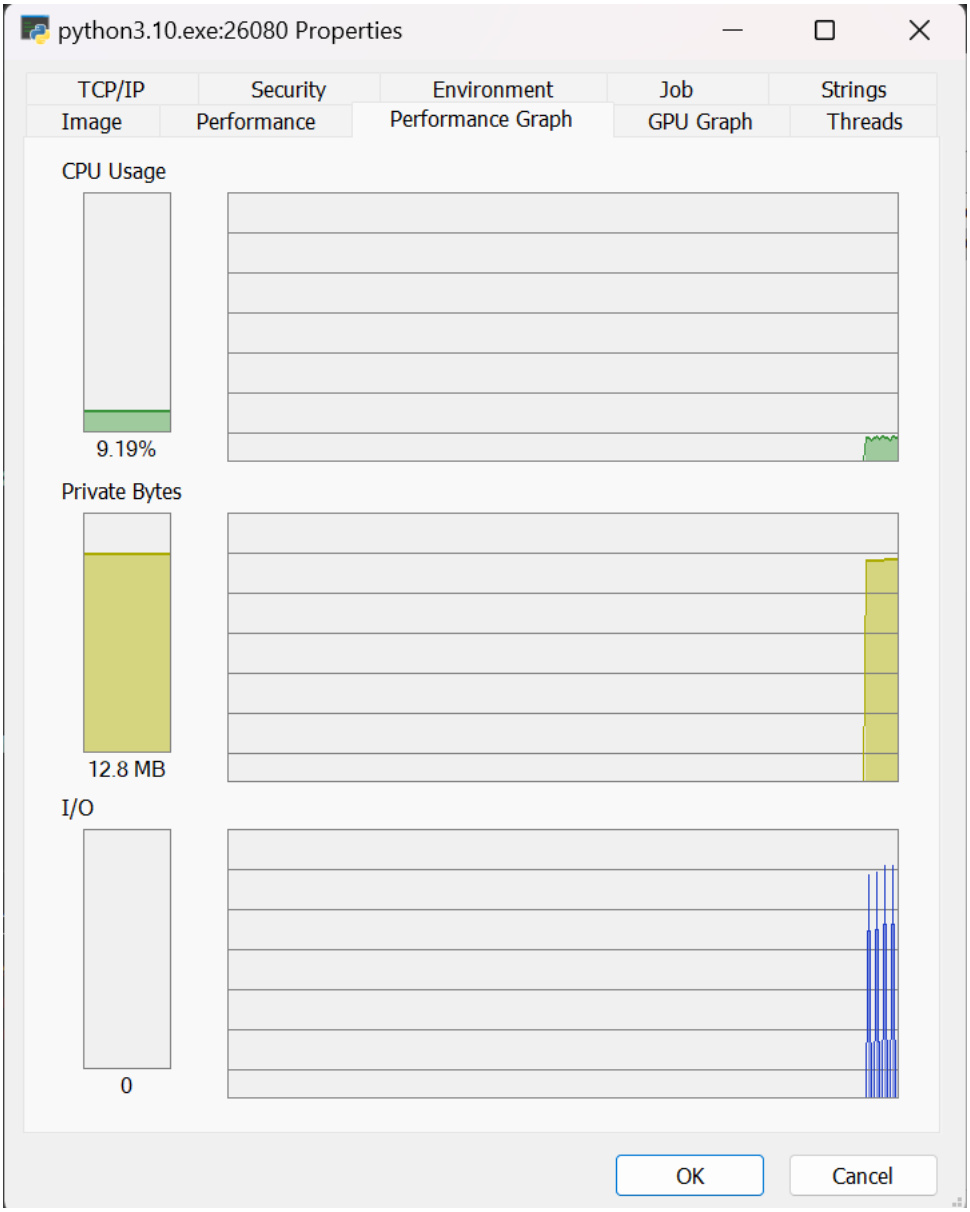
```
PS C:\Users\kDolp> & C:/Users/kDolp/AppData/Local/Microsoft/WindowsApps/python3.10.exe d:/PhisTech/Cryptography/AES-256
Doing aes-256 cbc for 3s on 16 size blocks: 501253 aes-256 cbc's in 3.01's
Doing aes-256 cbc for 3s on 64 size blocks: 451781 aes-256 cbc's in 3.0's
Doing aes-256 cbc for 3s on 256 size blocks: 372478 aes-256 cbc's in 3.0's
Doing aes-256 cbc for 3s on 1024 size blocks: 263138 aes-256 cbc's in 3.01's
Doing aes-256 cbc for 3s on 8192 size blocks: 79161 aes-256 cbc's in 3.0's
Doing aes-256 cbc for 3s on 16384 size blocks: 46743 aes-256 cbc's in 3.0's
The 'numbers' are in 1000s of bytes per second processed.
type                 16 bytes       64 bytes       256 bytes      1024 bytes     8192 bytes     16384 bytes
aes-256 cbc          2673.35k       9637.99k       31784.79k      89817.77k      216162.3k      255279.1k
```



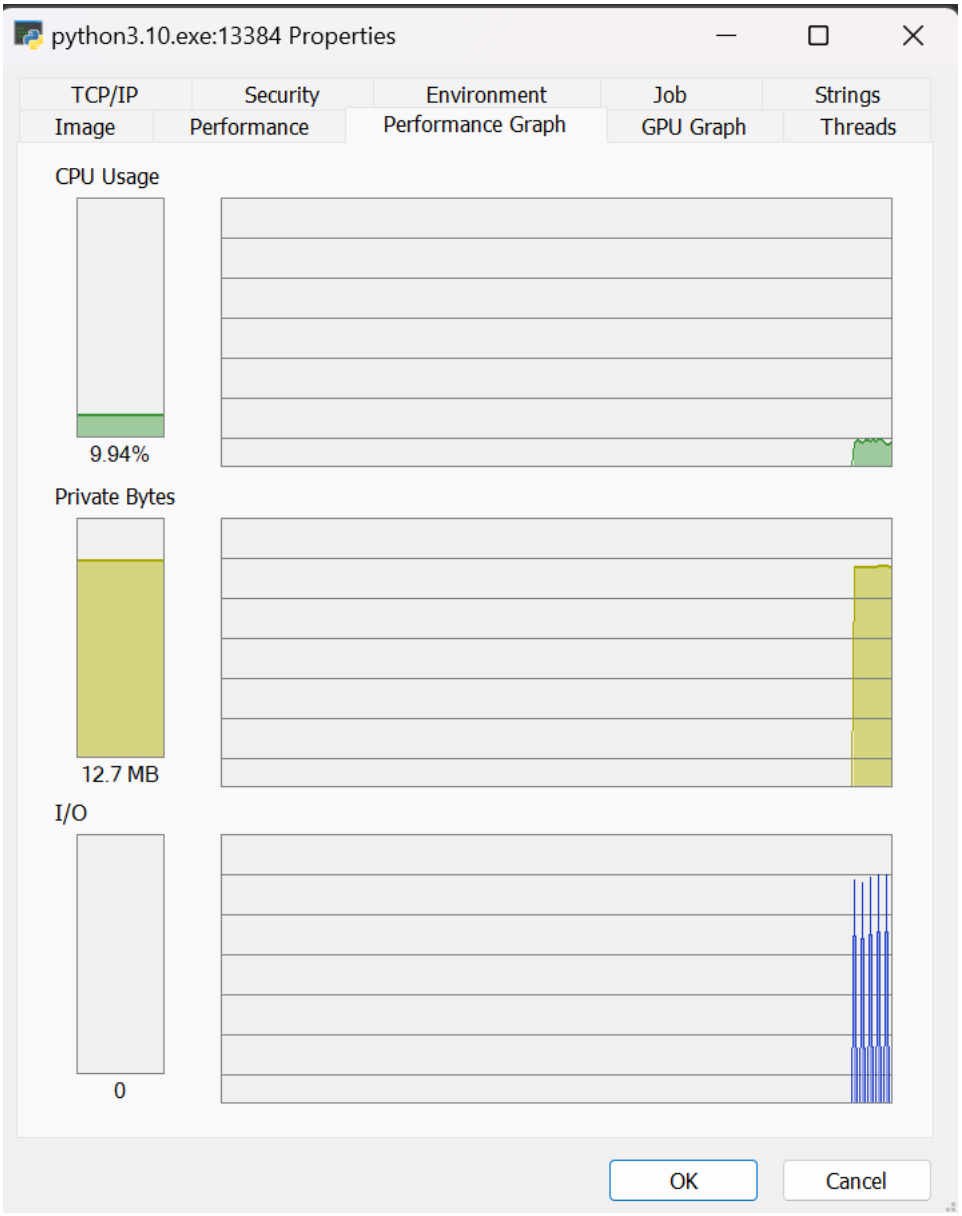

```
Doing aes-128 cbc for 3s on 16 size blocks: 415910 aes-128 cbc's in 3.01's
Doing aes-128 cbc for 3s on 64 size blocks: 461208 aes-128 cbc's in 3.01's
Doing aes-128 cbc for 3s on 256 size blocks: 250280 aes-128 cbc's in 3.0's
Doing aes-128 cbc for 3s on 1024 size blocks: 258284 aes-128 cbc's in 3.01's
Doing aes-128 cbc for 3s on 8192 size blocks: 80314 aes-128 cbc's in 3.0's
Doing aes-128 cbc for 3s on 16384 size blocks: 43205 aes-128 cbc's in 3.0's
The 'numbers' are in 1000s of bytes per second processed.
type                 16 bytes       64 bytes       256 bytes      1024 bytes     8192 bytes     16384 bytes
aes-128 cbc          2218.19k       9839.1k        21357.23k      88160.94k      219310.76k     235956.91k
```




```
Doing aes-256 gcm for 3s on 16 size blocks: 183695 aes-256 gcm's in 3.01's
Doing aes-256 gcm for 3s on 64 size blocks: 213493 aes-256 gcm's in 3.01's
Doing aes-256 gcm for 3s on 256 size blocks: 230901 aes-256 gcm's in 3.01's
Doing aes-256 gcm for 3s on 1024 size blocks: 202902 aes-256 gcm's in 3.01's
Doing aes-256 gcm for 3s on 8192 size blocks: 121281 aes-256 gcm's in 3.01's
Doing aes-256 gcm for 3s on 16384 size blocks: 101893 aes-256 gcm's in 3.01's
The 'numbers' are in 1000s of bytes per second processed.
type                     16 bytes       64 bytes       256 bytes      1024 bytes     8192 bytes     16384 bytes
aes-256 gcm              979.71k        4554.52k       19703.55k      69257.22k      331177.98k     556471.64k
```

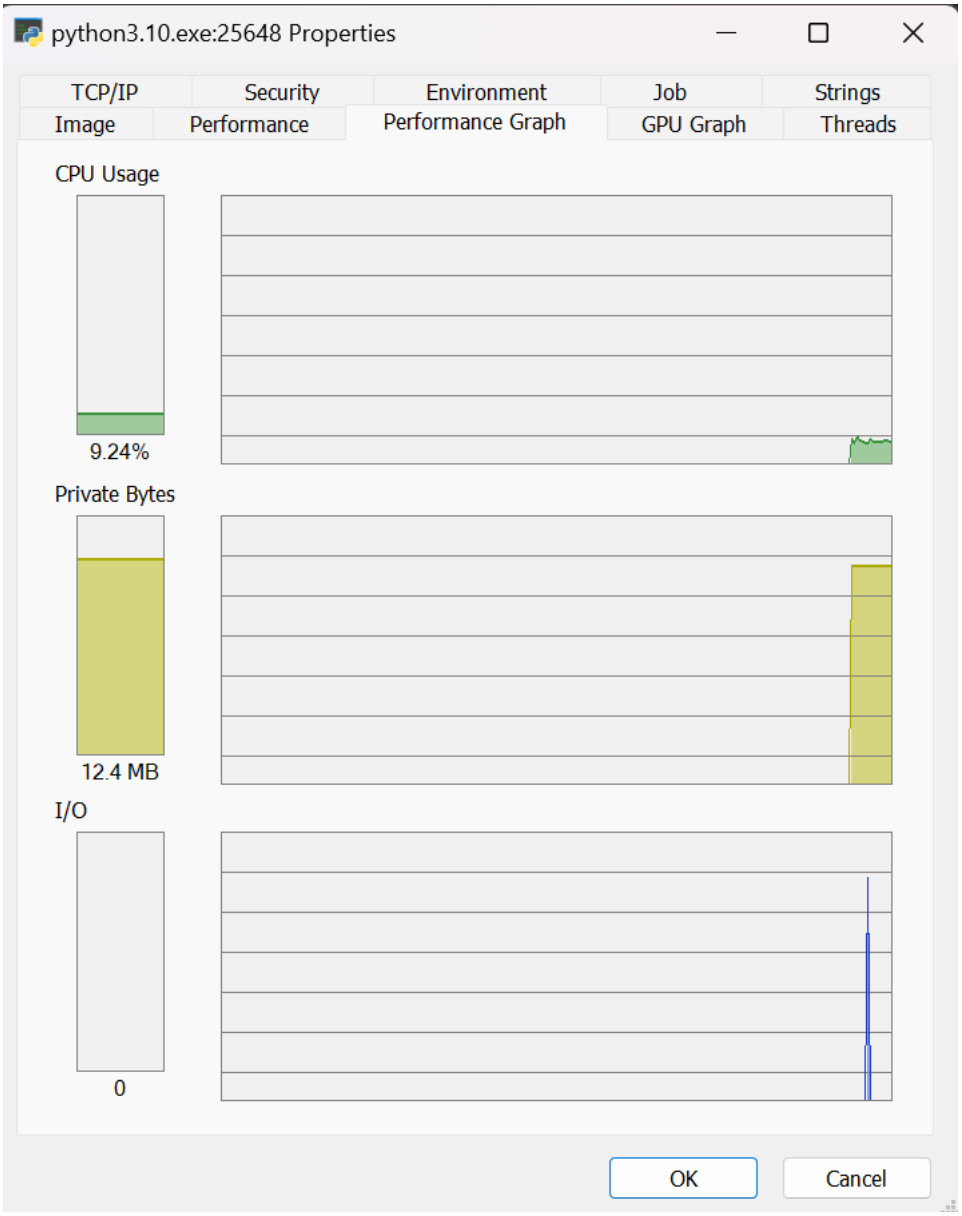



```
Doing aes-128 gcm for 3s on 16 size blocks: 183977 aes-128 gcm's in 3.01's
Doing aes-128 gcm for 3s on 64 size blocks: 212255 aes-128 gcm's in 3.0's
Doing aes-128 gcm for 3s on 256 size blocks: 161604 aes-128 gcm's in 3.01's
Doing aes-128 gcm for 3s on 1024 size blocks: 182444 aes-128 gcm's in 3.0's
Doing aes-128 gcm for 3s on 8192 size blocks: 127886 aes-128 gcm's in 3.01's
Doing aes-128 gcm for 3s on 16384 size blocks: 73342 aes-128 gcm's in 3.01's
The 'numbers' are in 1000s of bytes per second processed.
type                 16 bytes       64 bytes       256 bytes      1024 bytes     8192 bytes     16384 bytes
aes-128 gcm          981.21k        4528.11k       13790.21k      62274.22k      349214.04k     400545.11k
```



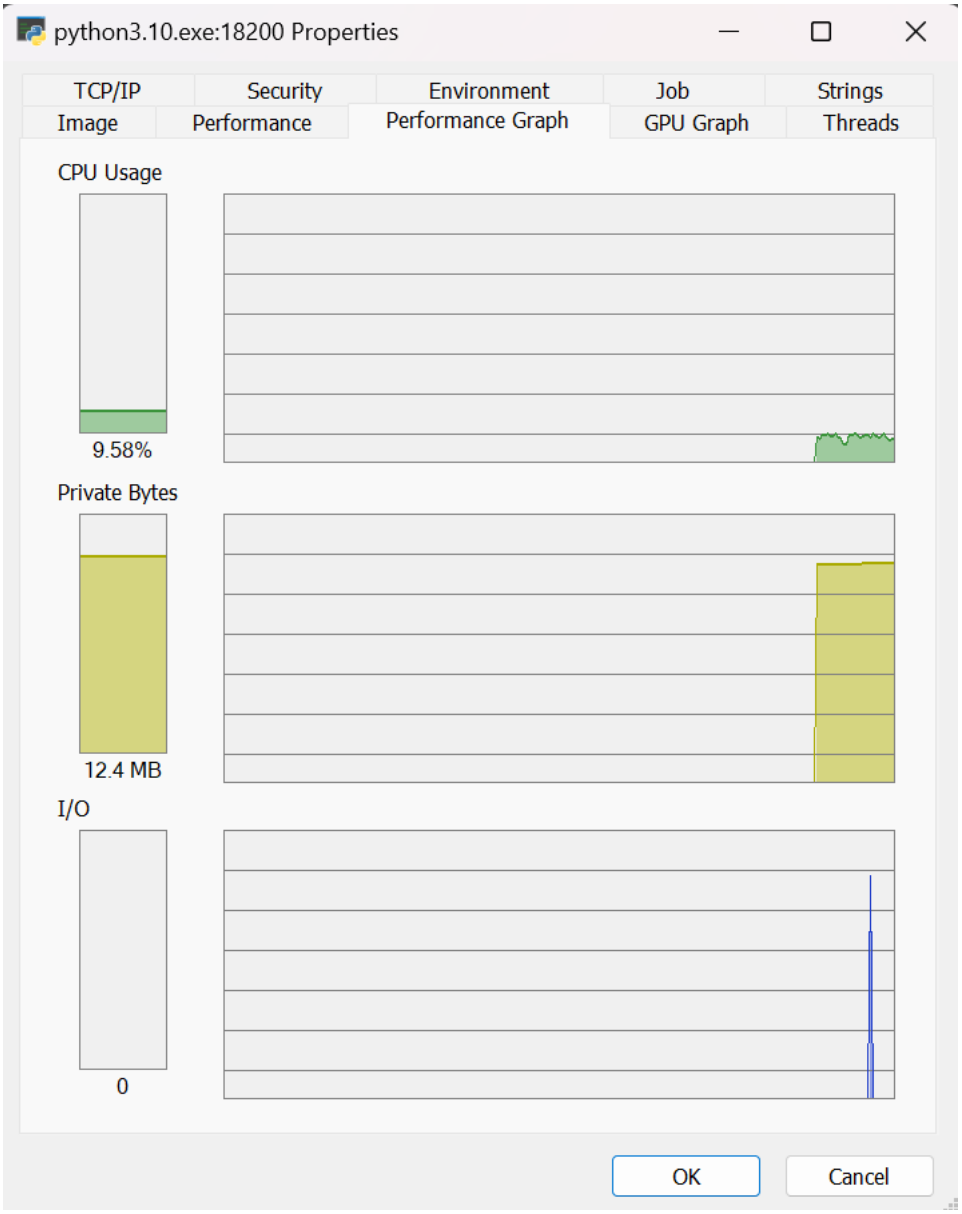

```
Doing 2048 bits private rsa's for 10s: 816 2048 bits private RSA's in 10.0s
Doing 2048 bits private rsa's for 10s: 4285 2048 bits private RSA's in 10.0s

                  sign            verify          sign/s          verify/s
rsa 2048 bits   0.012255s       0.002334s       81.6            428.5
```




```
Doing 4096 bits private rsa's for 10s: 176 4096 bits private RSA's in 10.03s
Doing 4096 bits private rsa's for 10s: 1511 4096 bits private RSA's in 10.0s

                  sign            verify          sign/s          verify/s
rsa 4096 bits     0.056818s       0.006618s       17.6            151.1
```


python3.10.exe:18200 Properties
TCP/IP
Security
Environment
Job
Strings
Image
Performance
Performance Graph
GPU Graph
Threads
CPU Usage
9.58%
Private Bytes
12.4 MB
I/O
0
OK
Cancel

Висновки

У висновку можна побачити, що швидкісний тест OpenSSL показав найкращі результати, ніж Crypto++ та PyCrypto. Crypto++ недалеко відстав від OpenSSL. У свою чергу PyCrypto показав найгірші результати у часі обробки. Це пов’язано з тим, що PyCrypto це бібліотека Python, яка є надбудовою на C-кодом, що у свою чергу призводить до непомітних, але у масштабі великих, затримок. Можна також побачити, що у PyCrypto було помічено найбільше зростання швидкості, ні у інших бібліотек. Це може бути пов’язано із тим, що зі збільшенням об’ємі даних зменшується кількість повільних пітонівських викликів. Найкращу швидкість показав OpenSSL. Трохи повільнішим виявився Crypto++. Це може бути пов’язано із тим, що бібліотека OpenSSL скомпільована з оптимізаціями, а Crypto++ тестувався із дефолтними налаштуваннями компіляції.