# Computer Security
# HW 2

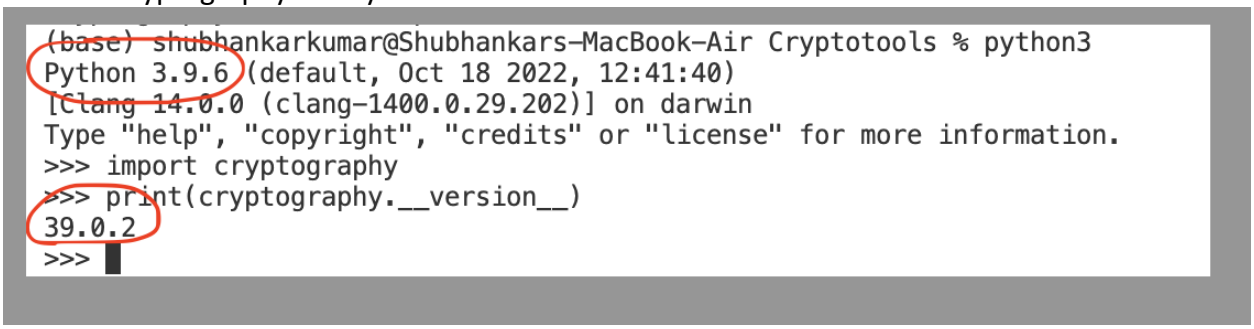
Shubhankar Kumar, skumar45

March 2023

Words: 1436
Pages: 6

1. **The chosen programming language and library option.**

   I chose Python as the programming language and pyca/Cryptography as the library option.
   For Python I chose version 3.9.6
   And the Cryptography library version is 39.0.2




2. **The timing results that your program measured as specified above.**

   Running AES Experiments

   AES with key size: 128-bit, mode : CBC and file size: 1024
   - Key generation time: 0.000017709 s
   - Encryption time: 0.125273834 s
   - Encryption speed: 8174.093242808 bytes/s
   - Decryption time: 0.000041208 s
   - Decryption speed: 24849543.777915023 bytes/s

   AES with key size: 128-bit, mode : CBC and file size: 10485760
   - Key generation time: 0.000012791 s
   - Encryption time: 0.030365416 s
   - Encryption speed: 345319161.772722185 bytes/s

Decryption time: 0.007622917 s
Decryption speed: 1375557414.569775581 bytes/s

AES with key size: 128-bit, mode : CTR and file size: 1024
Key generation time: 0.000011542 s
Encryption time: 0.000403292 s
Encryption speed: 2539103.180821839 bytes/s
Decryption time: 0.000035083 s
Decryption speed: 29187925.776024099 bytes/s

AES with key size: 128-bit, mode : CTR and file size: 10485760
Key generation time: 0.000005500 s
Encryption time: 0.005515458 s
Encryption speed: 1901158525.728950500 bytes/s
Decryption time: 0.005893791 s
Decryption speed: 1779119755.009981394 bytes/s

AES with key size: 256-bit, mode : CTR and file size: 1024
Key generation time: 0.000017042 s
Encryption time: 0.000170041 s
Encryption speed: 6022077.028481024 bytes/s
Decryption time: 0.000038458 s
Decryption speed: 26626449.633353133 bytes/s

AES with key size: 256-bit, mode : CTR and file size: 10485760
Key generation time: 0.000004834 s
Encryption time: 0.005030334 s
Encryption speed: 2084505720.693707943 bytes/s
Decryption time: 0.005213459 s
Decryption speed: 2011286556.583651781 bytes/s

Running RSA Experiments

RSA with key size: 2048-bit and File Size :1024
Key generation time: 0.065666042 s
Encryption time 0.001348876 s
Decryption time: 0.018388166 s
Encryption speed per byte: 759150.581669479 s
Decryption speed per byte: 55687.989764722 s

RSA with key size: 2048-bit and File Size :1048576
Key generation time: 0.125073208 s
Encryption time 0.702493163 s
Decryption time: 17.643177908 s
Encryption speed per byte: 1492649.402482572 s
Decryption speed per byte: 59432.376948630 s

RSA with key size: 3072-bit and File Size :1024
Key generation time: 0.337641875 s
Encryption time 0.001599832 s
Decryption time: 0.057937752 s
Encryption speed per byte: 640067.207058109 s
Decryption speed per byte: 17674.141033294 s

RSA with key size: 3072-bit and File Size :1048576
Key generation time: 0.602799833 s
Encryption time 1.393966696 s
Decryption time: 57.014375788 s
Encryption speed per byte: 752224.571081432 s
Decryption speed per byte: 18391.431731165 s

Running HASH Experiments

Computing timings for hash: sha256
Hash Value for small file:
b'\xdb\xd6\xa7\xfc\xab\xb10|\xe3\xdc[}\xc3+\xc1\xe5F\xbf\xb0B\xfa\x80\xa9\xd9v\xfb
\xbf[\x82\xbeH\x92'
Hash Value for large file:
b'\xa8=\x84!i\x1d\x82,\x97l\xe6\x89\x08\x07\xd5\xe6H\xbd\x95\xf2\xcae/!\x01
47\xcb\r\xebB'
Total time taken for small file: 0.003103750 s
Total time taken for large file: 0.063583125 s
Total time taken for both the files: 0.066686875 s
Per-byte time for small file: 0.000065124 s
Per-byte time for large file: 0.000000006 s

Computing timings for hash: sha512

Hash Value for small file:
b'0+\x03\xf4\xb3E\xe5\x12\n#F\xb2\x1aS\xb1[\x14\xb3|,\xbc|\xbex\xf8M\x0c\x0cG\x8f\xc4[C\x1f\xbe\x04\xde\x80cp\x0c\x02x\xe5\xd6K\xf3\xa8O\xe0\xdff\xb5+a\x04\xd4\x99\x91l\xb2\x16\xf4\xa9'

Hash Value for large file: b'+\xa4\x97W\xe7\xf9J\x8e\x13w1\xe9y\xd4V-\x89\xef\x0c\x11\x1al\x95\x12]`\xca\xde\x92=\x91\xbf\x07\x1ezU;\xca\xa0aJ\x06\x1d\xe7:\x8c%<\xb6)\xfcV\x82\xaf_P\xad\x06\x1f\x12&x\xd0\xc3'

Total time taken for small file: 0.000452250 s

Total time taken for large file: 0.041288375 s

Total time taken for both the files: 0.041740625 s

Per-byte time for small file: 0.000040762 s

Per-byte time for large file: 0.000000004 s

Computing timings for hash: sha3-256

Hash Value for small file:
b'\xd43\xf7*p{\xaf\x110\xad\xb6Em\x05r\xce\xa3\xc5G\xaa\x18\xa1\xb4\x1d\x10\xcc>\xbe`\x96\xd0r'

Hash Value for large file: b'\xc0(g-\xe5\xc7\xa5\xa4\x84ZP\xab\xa0\xdeb\xd7\xc3k\xf5\xca\xd9\xbaq4\xb5\xbe\x8d\xc5$&&'

Total time taken for small file: 0.000541417 s

Total time taken for large file: 0.057498459 s

Total time taken for both the files: 0.058039876 s

Per-byte time for small file: 0.000056680 s

Per-byte time for large file: 0.000000006 s

Running DSA Experiments

Calculating timings for 2048-bit DSA key

Key generation time: 1.140698250 s

Signature time for small file: 0.001339709 s

Signature time for large file: 0.046615833 s

Verification time for small file: 0.001114416 s

Verification time for large file: 0.046272750 s

Per-byte signature time for small file: 0.000001308 s

Per-byte signature time for large file: 0.000000004 s

Per-byte verification time for small file: 0.000001088 s

Per-byte verification time for large file: 0.000000004 s

Calculating timings for 3072-bit DSA key

Key generation time: 4.608681500 s
Signature time for small file: 0.002430833 s
Signature time for large file: 0.047638875 s
Verification time for small file: 0.002291750 s
Verification time for large file: 0.047517666 s
Per-byte signature time for small file: 0.000002374 s
Per-byte signature time for large file: 0.000000005 s
Per-byte verification time for small file: 0.000002238 s
Per-byte verification time for large file: 0.000000005 s

3. **For each performance aspect below, your comments about, (i) the expected performance, (ii) whether the observed performance followed the expected performance, and (iii) if there was a difference, your justification of the difference:**

i. **How per byte speed changes for different algorithms between small and large files.**

**Answer:**

Generally, we would expect the speed to be same but since we are adding overhead like Initializing vector or a Counter in AES mode, the per byte speed for larger file should be higher than the smaller file. The overhead becomes significant for a small file while for a larger file it is insignificant relative to the file size. Similarly different algorithms have their own overheads, like Hashing requires allocating memory or do some initializations, so again per byte time will be higher for smaller file as compared to a large file.

That is what we observed in the experiment as well. As we can see in the results obtained above, the per byte speed for larger file is much faster than the smaller file in all the algorithms and per byte time is higher for smaller file than large file.

ii. **How encryption and decryption times differ for a given encryption algorithm.**
**Answer:**

**AES CBC**: Being a Symmetric method, the encryption and decryption time should be similar, however with different modes this result can be different. For example, in CBC mode, we can expect decryption to be faster than encryption as we are generating an IV for encryption which is not required while decrypting. Also, there are ways available in modern computing which allows us to parallelize and pipeline CBC decryption which is not possible while encrypting because of the dependency on previous block's output. And as can be seen from the results obtained above, the decryption speed is indeed much higher than the encryption speed.

**AES CTR:** We got very interesting results for AES CTR mode. The encryption and decryption time is similar for the large file while the decryption time is much faster for the small file which was the expected result.

**RSA:** The encryption speed should be much faster than the decryption speed considering d >> e, that is private key is larger than the public key. As given, the private key is 2048 and 3072 bits which is very large and calculating anything to the power of these numbers will take a lot of time. The same can be observed in the obtained results, the encryption speed is almost 10-15 times higher than decryption speed and this value increases to appx 50-60 times for the large file.

iii. How key generation, encryption, and decryption times differ with the increase in the key size.
**Answer:**
**AES:** The time should generally increase with the increase in key size, but for AES it does not increase considerably with the increase in key size. And the same can be seen in the results.
**RSA:** The RSA computations depend on key size as we are calculating power raised to the key size. And with the increase in key size the encryption and decryption time increases especially decryption speed increases a lot with the increase in the key size. The same can be seen in the results.
**DSA:** The key generation time, encryption, and decryption time increases with the increase in the key size.

iv. How hashing time differs between the algorithms and with increase of the hash size.
**Answer:** Generally, sha 256 and sha3-256 should take similar time, while sha 512 should be slower because of the increase in the output size. But in my case the results are not as expected. sha 512 is taking the least time.

v. How performance of symmetric key encryption (AES), hash functions, and public-key encryption (RSA) compared to each other.
**Answer:**
AES is the fastest and RSA is the slowest while Hashing performs in the middle. For the large file RSA is much much slower than AES and Hashing which was expected as RSA is computationally much expensive than AES and Hashing and for a large file, the difference in speed gets added. For smaller file, the Hashing and RSA performs similar. The results are according to the expectation, AES being a symmetric key encryption is supposed to the fastest and it performed accordingly, while RSA is computationally very expensive and is supposed to be slowest.