[APPLICATION OF CYRPTOGRAPHY]

CIS 4367.01 Computer Security, Fall 2025

Nickolas Diaz

[Xianping Wang]

Table of Contents

Contents

Abstract

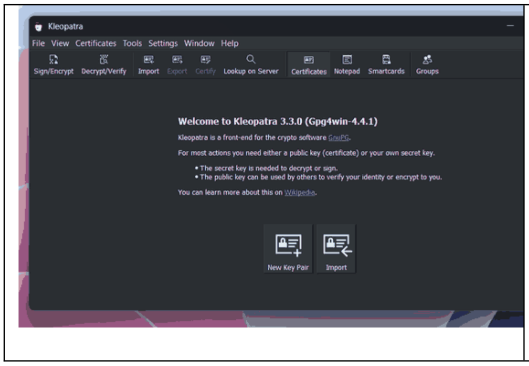
The purpose of this lab is to demonstrate symmetrical/asymmetric cryptography and hash functions. It is a demonstration of using a tool called Gpg4win with manage private/public keys and certificates to secure emails/files. We will also use hash functions to verify whether we got the correct file from a hash. Lastly, we will use Mailvelope with OpenPGP in order to securely send and receive emails from out partner.

Tasks

Task 1: Generate, Import, and Manage OpenPGP Certificates

First install Gpg4win
Command used: winget install -e --id GnuPG.Gpg4win

Generate an OpenPGP certificate using my school email
I uploaded both my personal and school emails


Kleopatra
File View Certificates Tools Settings Window Help
Sign/Encrypt Decrypt/Verify Import Export Certify Lookup on Server Certificates Notepad Smartcards Groups
New Key Pair
Import

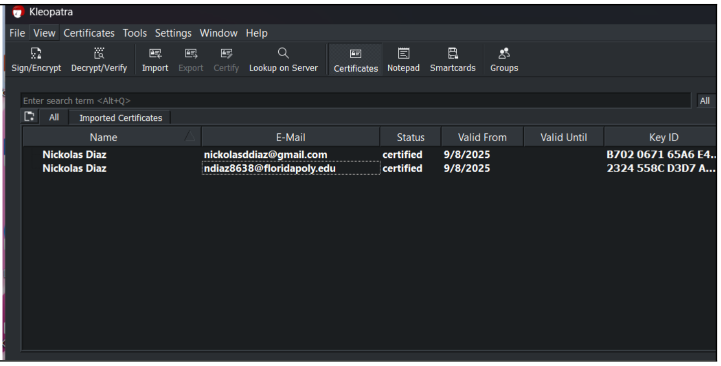
Welcome to Kleopatra 3.3.0 (Gpg4win-4.4.1)

Kleopatra is a front-end for the crypto software GnuPG.

For most actions you need either a public key (certificate) or your own secret key.

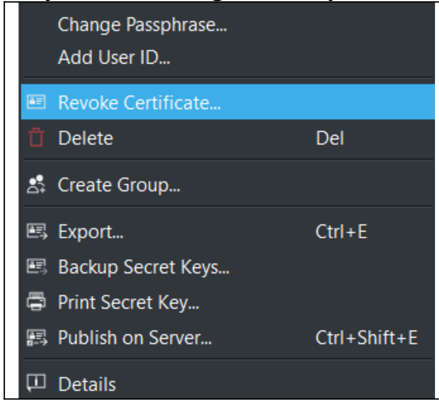
- The secret key is needed to decrypt or sign.
- The public key can be used by others to verify your identity or encrypt to you.

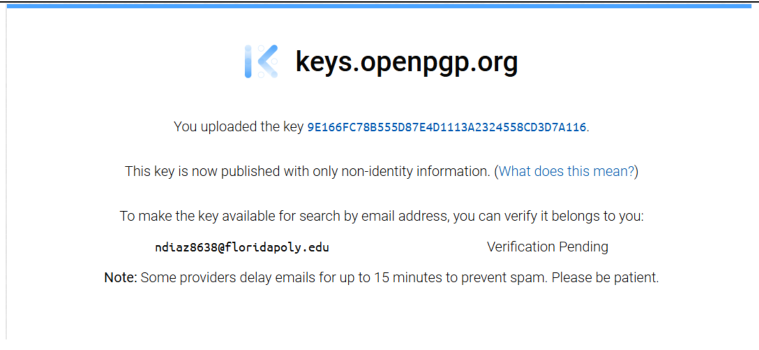
You can learn more about this on Wikipedia.


Kleopatra
File View Certificates Tools Settings Window Help
Sign/Encrypt Decrypt/Verify Import Export Certify Lookup on Server Certificates Notepad Smartcards Groups
Enter search term <Alt+Q>
All
All
Imported Certificates

Name	E-Mail	Status	Valid From	Valid Until	Key ID
Nickolas Diaz	nickolasddiaz@gmail.com	certified	9/8/2025		B702 0671 65A6 E4..
Nickolas Diaz	ndiaz8638@floridapoly.edu	certified	9/8/2025		2324 558C D3D7 A...

Publish your OpenPGP certificate on a public key server.
There are two options of uploading to a public server I did both, however the website was the only one wanting to verify the email





You uploaded the key 9E166FC78B555D87E4D1113A2324558CD3D7A116.

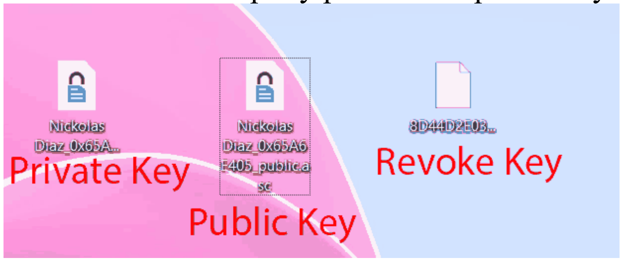
This key is now published with only non-identity information. (What does this mean?)

To make the key available for search by email address, you can verify it belongs to you:

ndiaz8638@floridapoly.edu Verification Pending

Note: Some providers delay emails for up to 15 minutes to prevent spam. Please be patient.

Backup your private (secret) key
Export your public key from your OpenPGP certificate
I was able to backup my private and public key in addition to the revoke key.



Note I could not verify my school email address, however my personal computer was able to be verified

Task 2: Data/Program Integrity Assurance

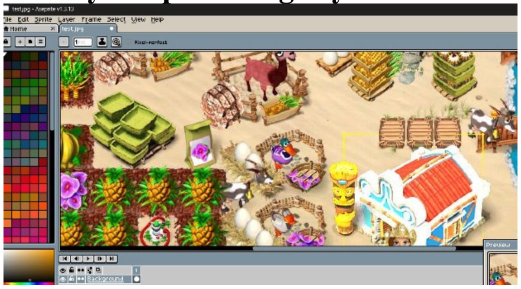
Take a photo with your smartphone and transfer it to your computer.
Create a checksum for this photo.

Image used it is used from game Ranch rush 2

Verify that the checksum is valid.

Command used: Get-FileHash -Path "C:\Users\nicko\Desktop\test.jpg" -Algorithm SHA256
Command used: Get-FileHash -Path "C:\Users\nicko\Desktop\test.jpg" -Algorithm MD5
SHA256:
2B29C2E1346EC102DEDE88D11A203D7136B1705D820D6738AA93182B1920B0E1
MD5: 9FEE8D0D36F96FACA0F9413D7BAFE700

Modify the photo slightly and confirm that the old checksum is now invalid.


test.jpg - Aseprite v1.3.13
File Edit Sprite Layer Frame Select View Help
Home
test.jpg
Pixel-perfect

One pixel was changed
New hashes
SHA256
D4BE450550F0DAB2D5D179B51013DBA5DB743B6305995CFF7E8E6BDE922D8426
MD5 5A935BFB950B70B3CEC40817878C7853

Answer the following questions:

What checksum algorithm did you use?

I used SHA256 and MD5 to hash the picture

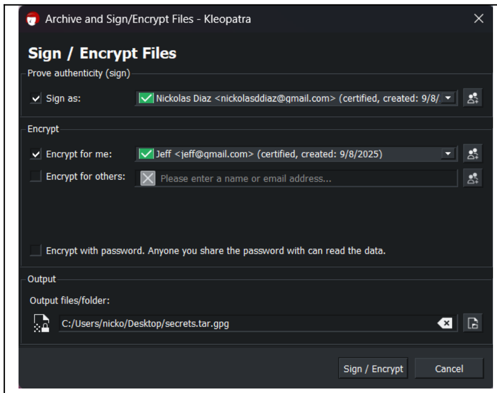
What is the length of this checksum in bits, bytes, and hexadecimal digits?

For sha256 there are 64 characters/hexadecimal digits, each hex character is 4 bits each so 256 bits and 8 bytes

For md5 there are 32 characters/hexadecimal digits, each hex character is 4 bits each so 128 bits and 4 bytes

Task 3: Privacy Assurance

Put a couple of pictures inside a folder named secrets
Using my private key and my friend's public key encrypt the folder and send it to my friend
Receive secrets from a friend by decrypting the received file
Decrypt the secrets file received from my friend



Archive and Sign/Encrypt Files - Kleopatra

Sign / Encrypt Files

Prove authenticity (sign)

☑ Sign as: Nickolas Diaz <nickolasddiaz@gmail.com> (certified, created: 9/8/

Encrypt

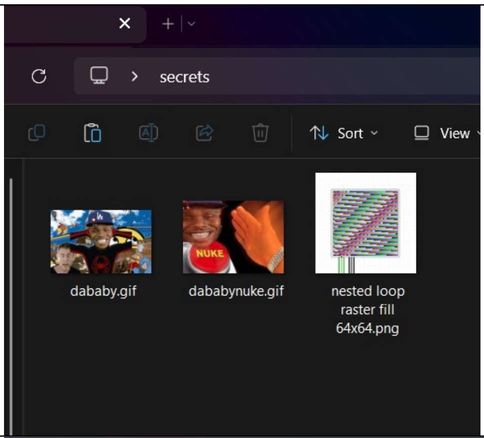
☑ Encrypt for me: Jeff <jeff@gmail.com> (certified, created: 9/8/2025)

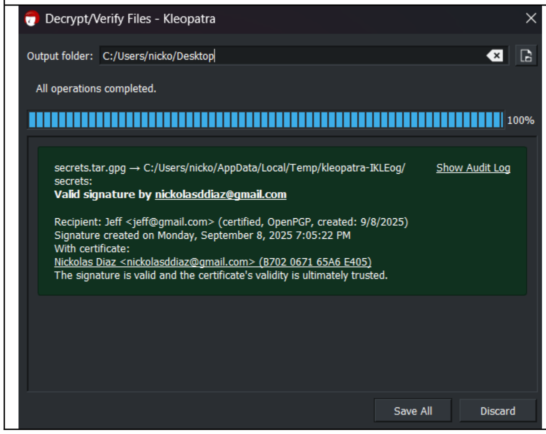
□ Encrypt for others: Please enter a name or email address...

□ Encrypt with password. Anyone you share the password with can read the data.

Output

Output files/folder: C:/Users/nicko/Desktop/secrets.tar.gpg





Decrypt/Verify Files - Kleopatra

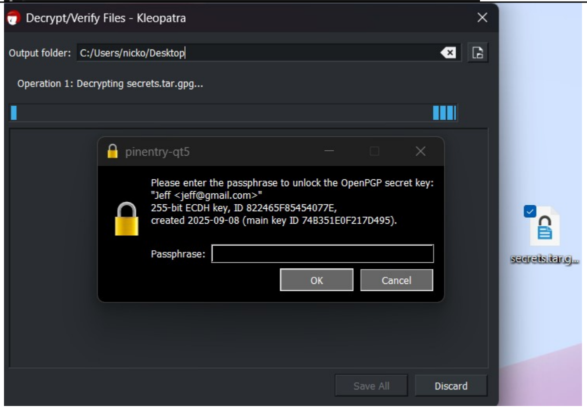
Output folder: C:/Users/nicko/Desktop

All operations completed.

100%

secrets.tar.gpg → C:/Users/nicko/AppData/Local/Temp/kleopatra-IKLEog/secrets:
Valid signature by nickolasddiaz@gmail.com

Recipient: Jeff <jeff@gmail.com> (certified, OpenPGP, created: 9/8/2025)
Signature created on Monday, September 8, 2025 7:05:22 PM
With certificate:
Nickolas Diaz <nickolasddiaz@gmail.com> (B702 0671 65A6 E405)
The signature is valid and the certificate's validity is ultimately trusted.



Decrypt/Verify Files - Kleopatra

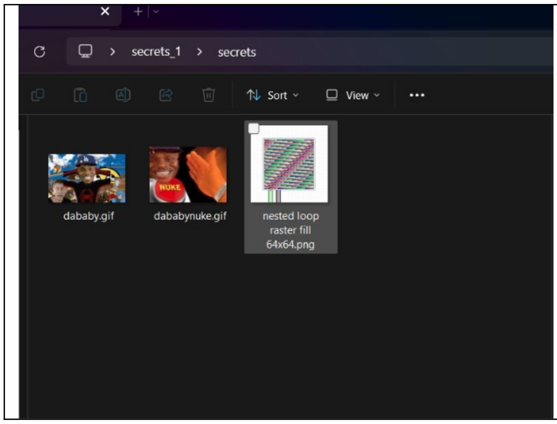
Output folder: C:/Users/nicko/Desktop

Operation 1: Decrypting secrets.tar.gpg...

pinentry-qt5

Please enter the passphrase to unlock the OpenPGP secret key:
"Jeff <jeff@gmail.com>"
255-bit ECDH key, ID 822465F85454077E,
created 2025-09-08 (main key ID 74B351E0F217D495).

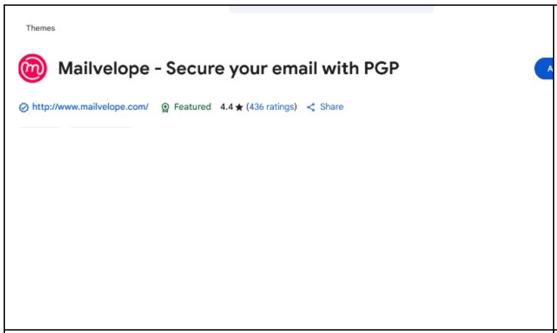
Passphrase:

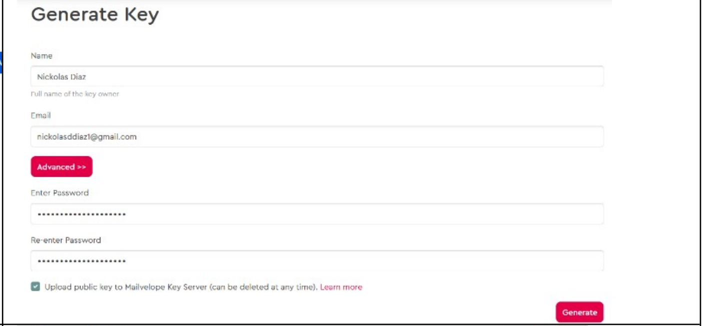


I imported Jeff’s public key and signed my folder using my private key then sent it to my friend which was able to decrypt it using his private key jeff was able to see my photos.

Task 4: Secure Emails with Mailvelope

Import your private OpenPGP key exported from Kleopatra.
Import your partner's OpenPGP certificate exported from Kleopatra or a key server.
Send an encrypted and signed email to your partner.
Decrypt and verify your partner's encrypted and signed email.


Themes
Mailvelope - Secure your email with PGP
http://www.mailvelope.com/
Featured
4.4 ★ (436 ratings)
Share


Generate Key

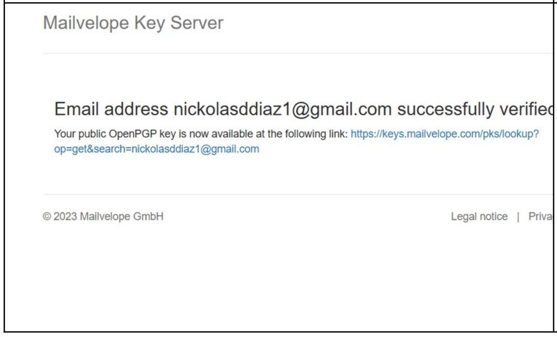
Name: Nickolas Diaz
Full name of the key owner

Email: nickolasddiaz1@gmail.com

Enter Password: ••••••••••••••••••••

Re-enter Password: ••••••••••••••••••••

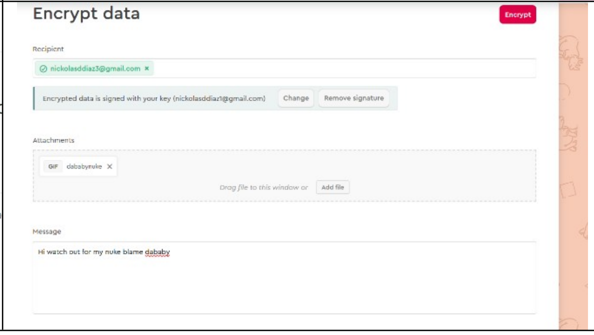
☑ Upload public key to Mailvelope Key Server (can be deleted at any time). Learn more


Mailvelope Key Server

Email address nickolasddiaz1@gmail.com successfully verified

Your public OpenPGP key is now available at the following link: https://keys.mailvelope.com/pks/lookup?op=get&search=nickolasddiaz1@gmail.com

© 2023 Mailvelope GmbH

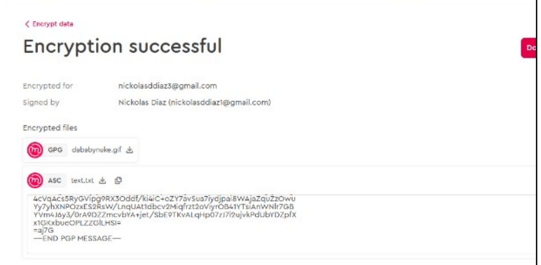

Encrypt data

Recipient: nickolasddiaz3@gmail.com ×

Encrypted data is signed with your key (nickolasddiaz1@gmail.com)

Attachments: GIF dababynuke ×

Message: Hi watch out for my nuke blame dababy



Encryption successful

Encrypted for nickolasddiaz3@gmail.com

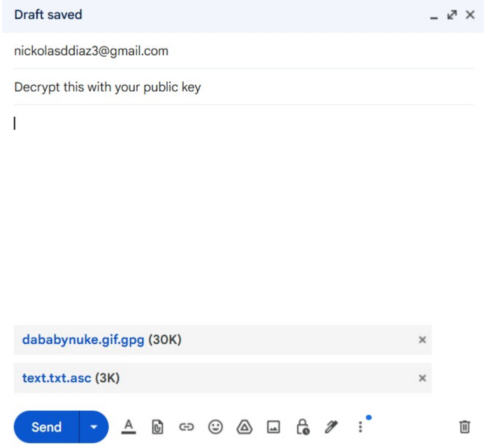
Signed by Nickolas Diaz (nickolasddiaz1@gmail.com)

Encrypted files

dababynuke.gif

text.txt

```
4cVqAcs5RyGVipg9RX3Oddf/ki4iC+oZY7avSua7iydjpai8WAjaZquZzOwu
Yy7yhXNPOzxES2RsW/LnqUAt1dbcv2Miqfrzt2oViyrOB41YTsiAnWNlr7GB
YVm4J6y3/0rA9DZZmcvbYA+jet/SbE9TKvALqHp07zJ7i2ujvkPdUbYDZpfX
x1GKxbueOPLZZGlLHSI=
=aj7G
—END PGP MESSAGE—
```

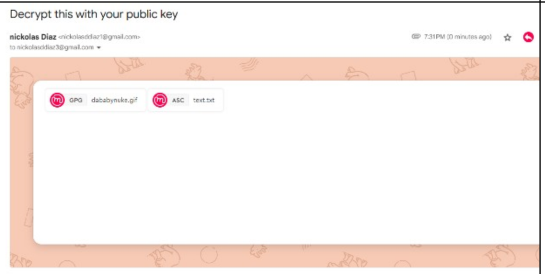


nickolasddiaz3@gmail.com

Decrypt this with your public key

dababynuke.gif.gpg (30K)

text.txt.asc (3K)



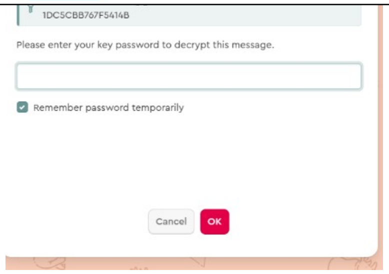
Decrypt this with your public key

nickolas Diaz <nickolasddiaz1@gmail.com> 7:31 PM (0 minutes ago)

to nickolasddiaz3@gmail.com

dababynuke.gif

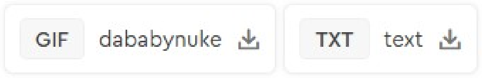
text.txt

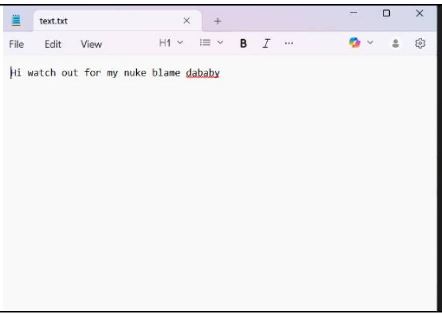


1DC5CBB767F5414B

Please enter your key password to decrypt this message.

☑ Remember password temporarily





```
Hi watch out for my nuke blame dababy
```

First step download Mailvelope on both ends, next generate key input email and password then verify your email, next send each other your public keys, then click encrypt and input the friend's public key and your message/image, next send the encrypted message though email, then the friend decrypts the message using their private key. Next read the email.

Conclusions

Detail the output and results of the laboratory exercises. Answer the question: "What did you learn during this lab?"

I did many exercises encrypting and decrypting using public and private keys, first was using OpenPGP to encrypt my private key and my fiends public key to decrypt my message and send it to my friend which was able to decrypt it using his private key. Next challenge was using hash on two different pictures, the original and the original with one pixel change. I used sha256 and md5 to hash and found out that the hashes from the two files were drastically different. Next, I used GnuPG with my OpenPGP key to send and decrypt a file with a couple of pictures. The last challenge was using Mailvelope to encrypt and decrypt a message and sending it through email.

References

https://github.com/ufidon/comsec/tree/main/labs/lab02