LAB 02

CIS4367.01 – Computer Security

Jaleel Rogers

1 Abstract

Lab 02 focuses on improving the security of files using cryptographic techniques. These techniques are important for ensuring confidentiality, integrity, and authenticity (CIA), and non-repudiation of data. The lab is to be conducted on Windows 2019 Server along with the collaboration of another classmate or account.

2 Tasks

2.1 Task 1: Secure e-mails and files using Gpg4win

2.1.1. Steps

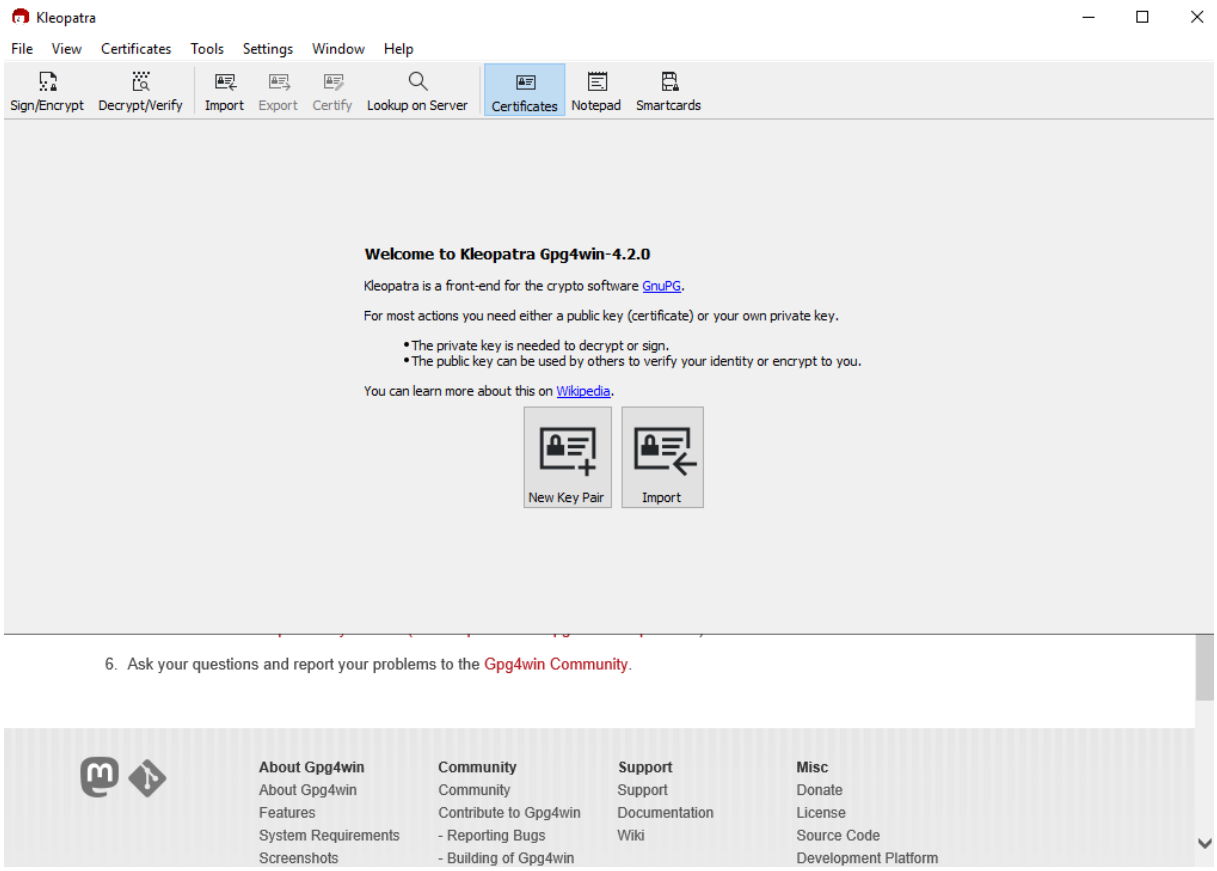

Figure 1 – Kleopatra

I thought the application would be called Gpg4 but instead Gpg4 contains a exe called Kleopatra which is an interface to GnuPG to encrypt/decrpyt and authenticate files.

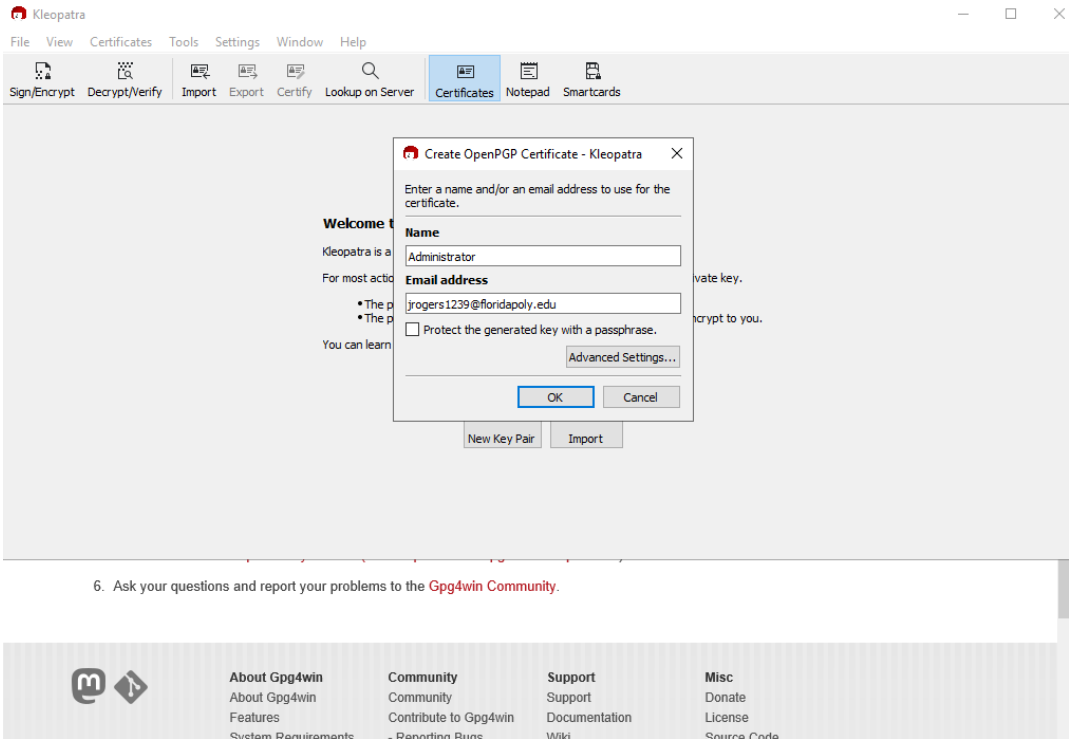

Figure 2 - Creating a Certificate

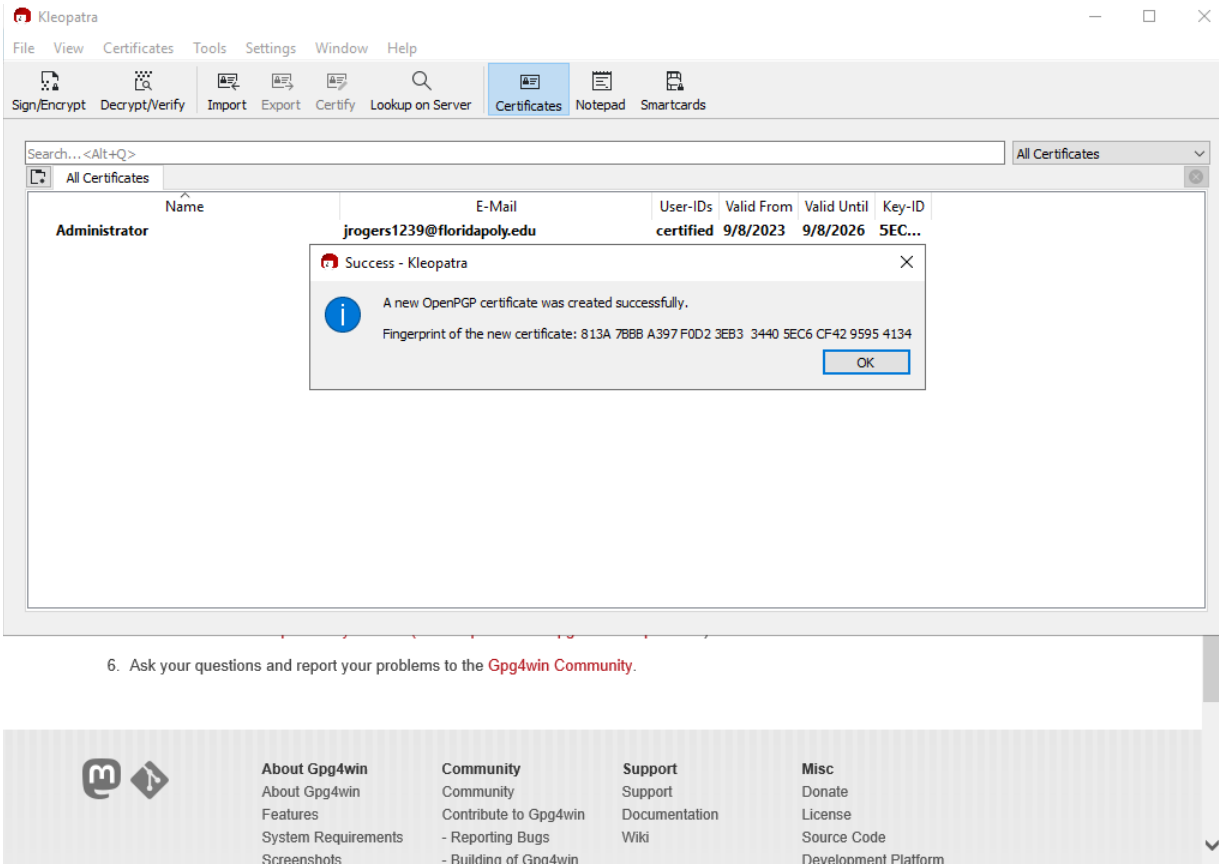

Figure 3 - Successful Creation of the Certificate

Figures 2 & 3 show the creation of the certificate. It was easier than I expected. I thought the email would receive a notification that its email address is being used in a certificate, but I guess not. The host is called the Administrator.

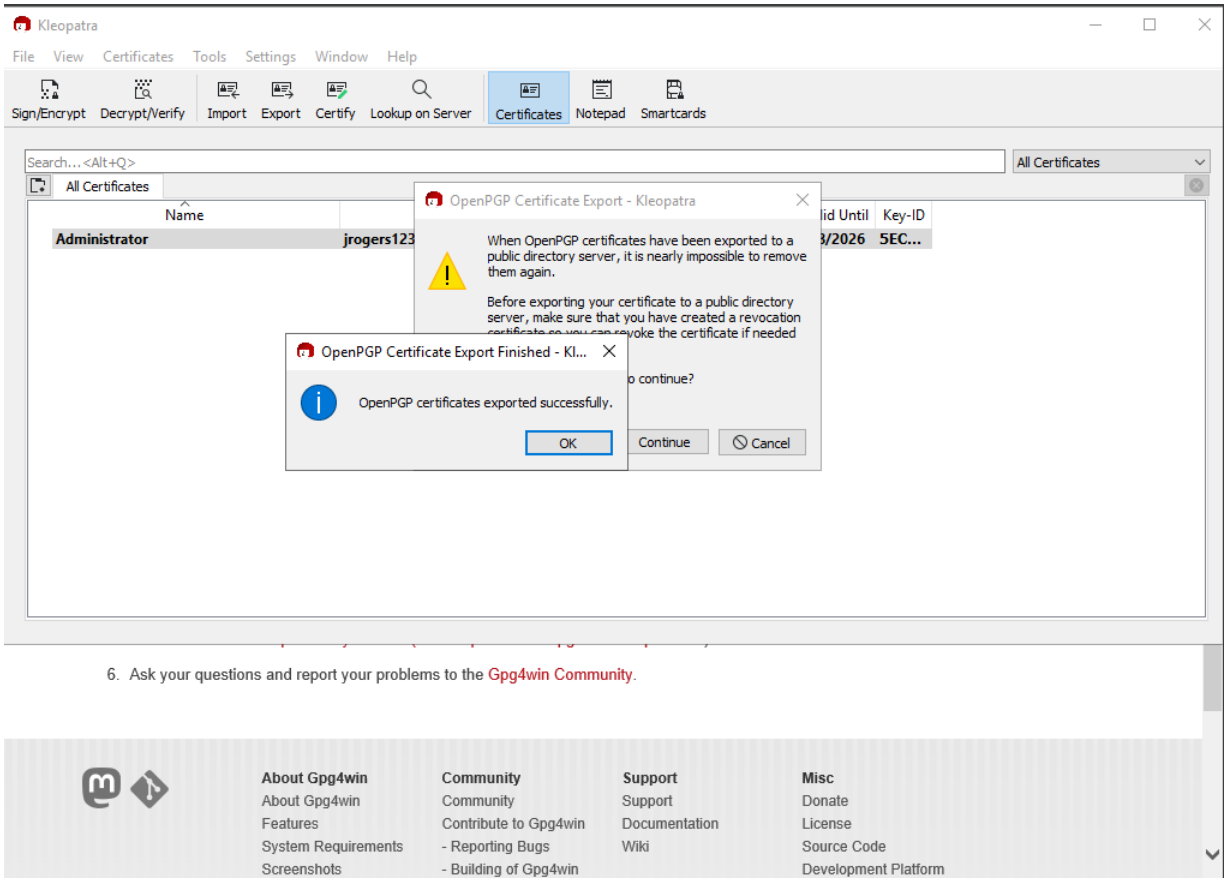

Figure 4 - Successful Publication of Certificate

Made my certificate publicly available.

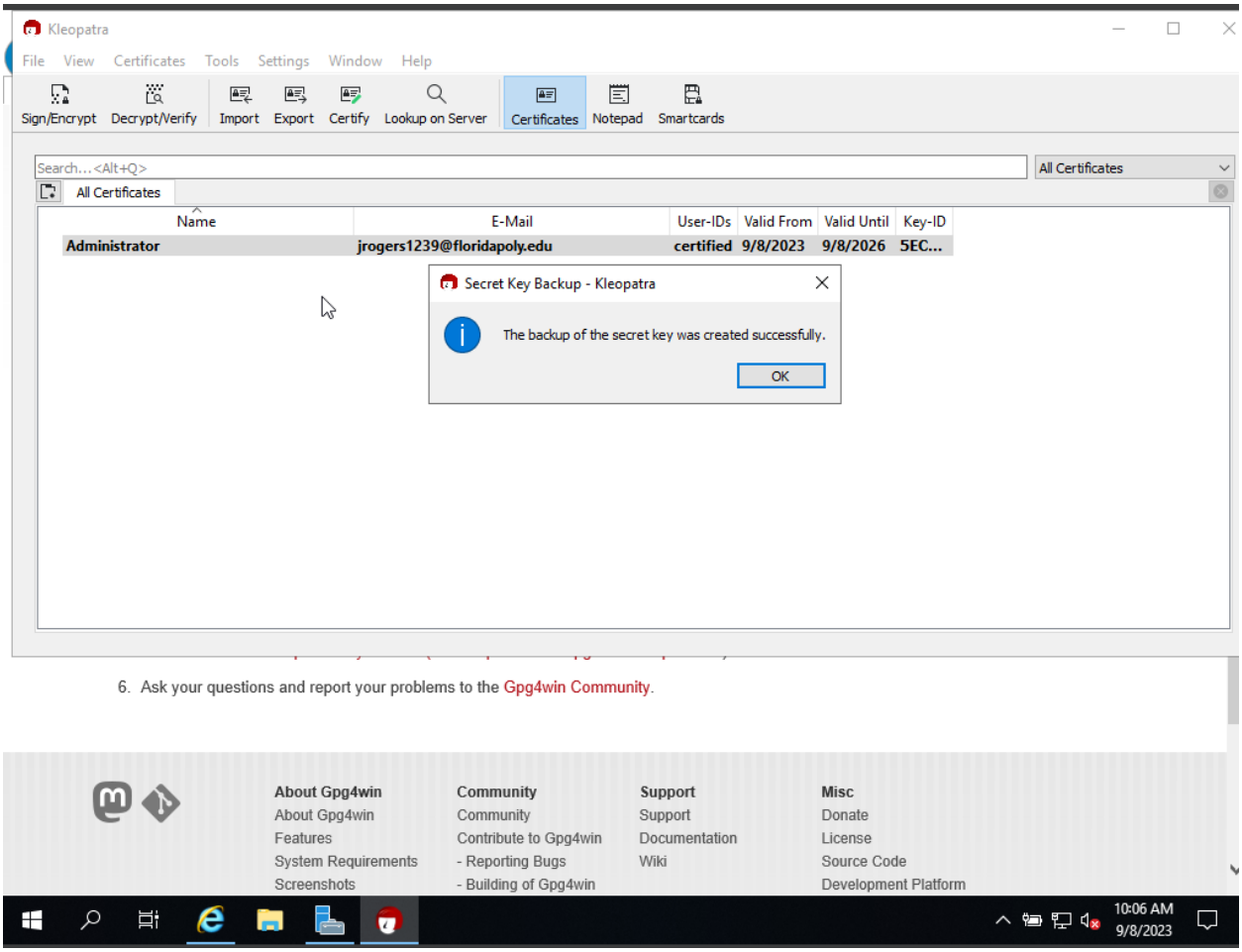

Figure 5 - Back Up Private Key

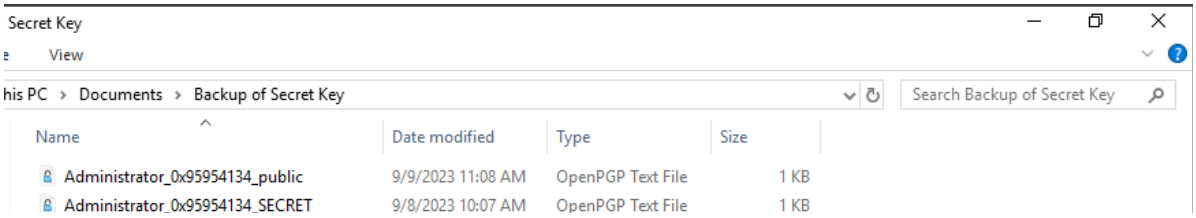

Figure 6 - Exportation of Public & Private Keys

For Figure 5 & 6, I backed up both my public and private keys just in case. I left them in a folder together called ‘Backup of Secret Key’.

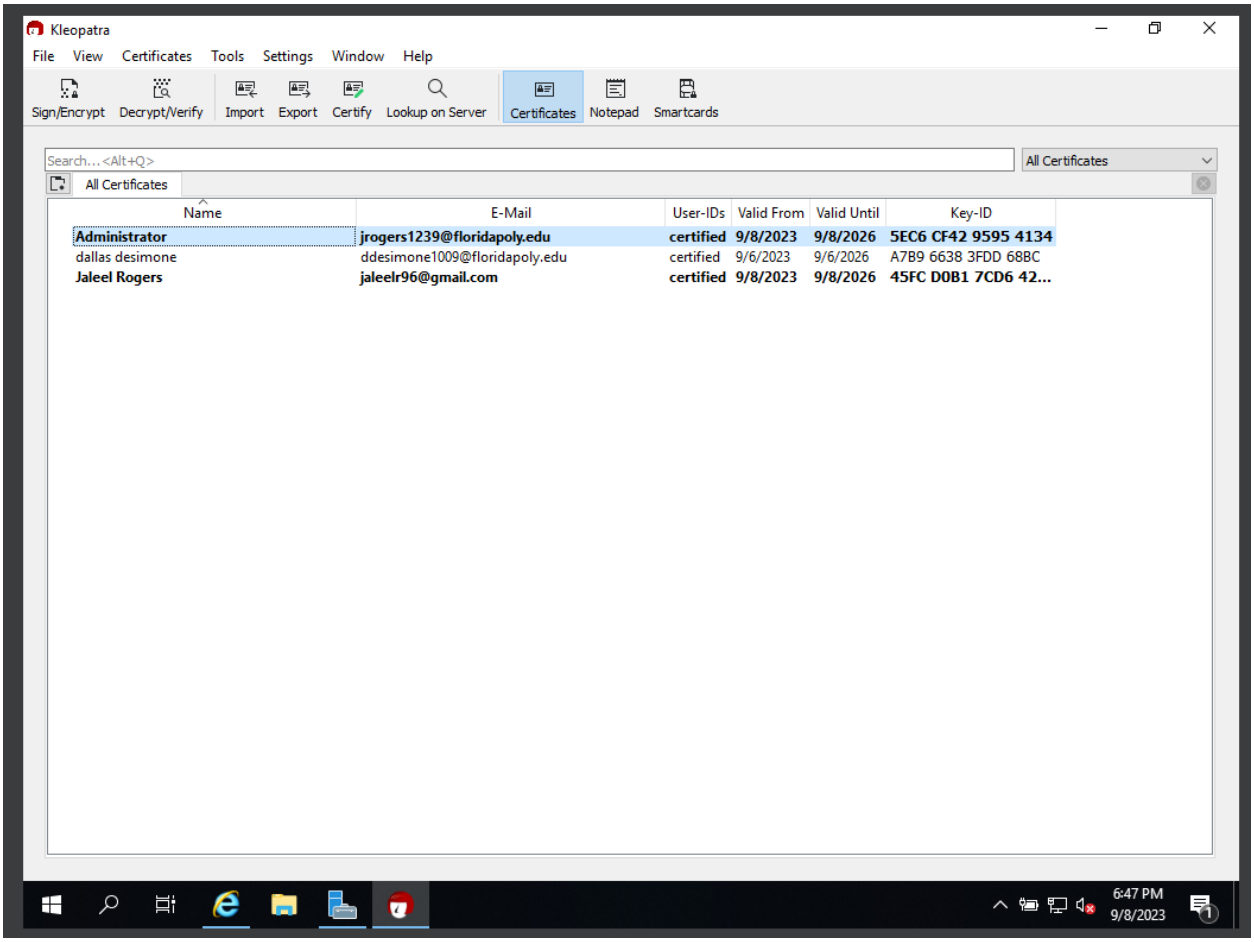

Figure 7 - Current Certificates

I found a friend's certificate; however, it expired before I could use it. So, an alternative I had was I created another certificate using my personal email address to act as my friend.

2.2 Task 2: Data/program integrity assurance

2.2.1 Steps

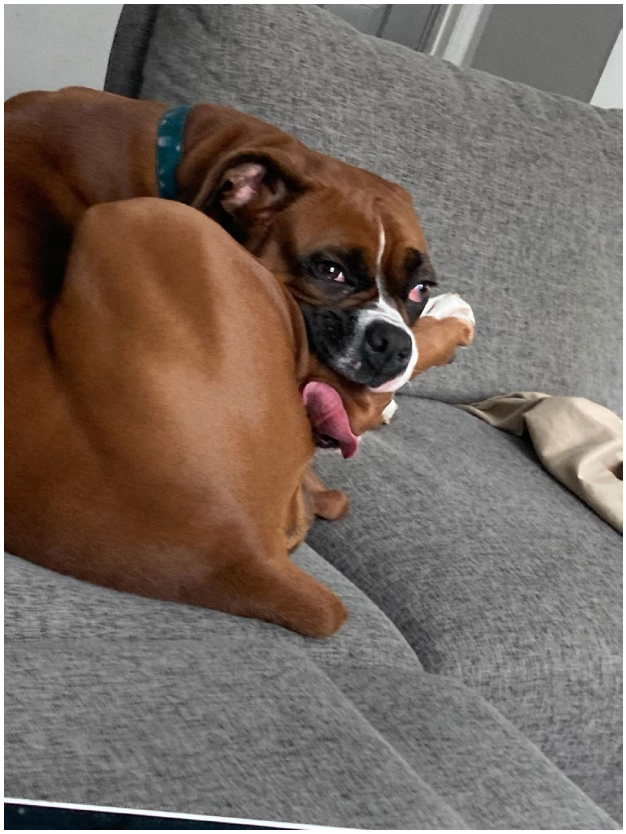

Figure 8 - image5222.jpg

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\jalee> Get-FileHash -Path "C:\Users\jalee\Pictures\image5222.jpg" -Algorithm MD5

Algorithm       Hash                                                                   Path
---------       ----                                                                   ----
MD5             AC8B963E9B74A811AD6EB6E310F6B9D0                                       C:\Users\jalee\Pictures\image...


PS C:\Users\jalee> ac8b963e9b74a811ad6eb6e310f6b9d0
```

Figure 9 - Creation of the Checksum

Figure 8 is that contents of the image5222.jpg which I will use to make the checksum for. In Figure 9 I initially created the checksum using the MD5 hashing algorithm on PowerShell. The Command ‘Get-FileHash’ will get the hash of the file within the path while ‘Algorithm’ will use one of multiple algorithms like SHA256, MD5, etc. for the checksum.

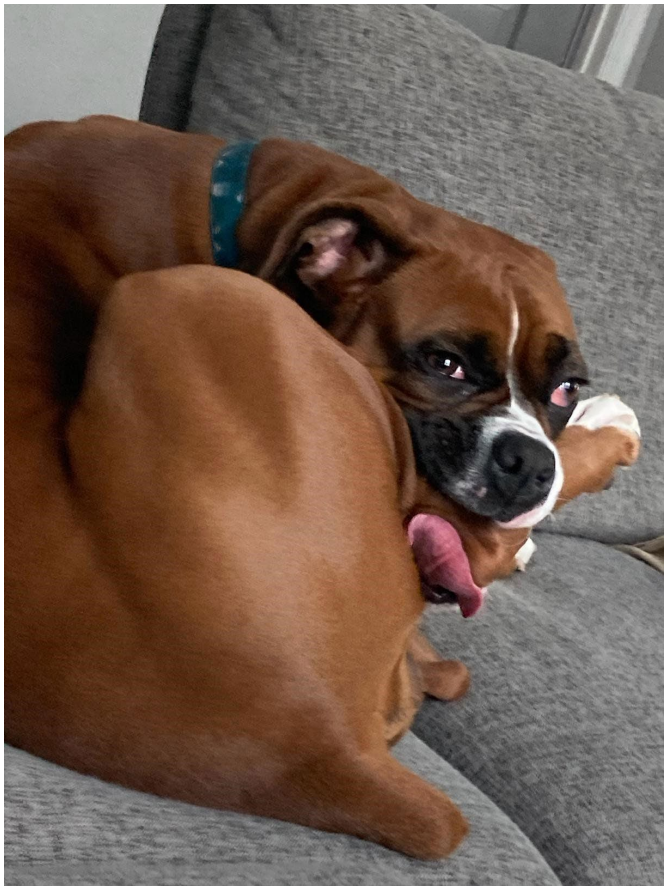

Figure 10 - Modified image 5222.jpg

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\jalee> Get-FileHash -Path "C:\Users\jalee\Pictures\image5222.jpg" -Algorithm MD5

Algorithm       Hash                                                                   Path
---------       ----                                                                   ----
MD5             AC8B963E9B74A811AD6EB6E310F6B9D0                                       C:\Users\jalee\Pictures\image...


PS C:\Users\jalee> Get-FileHash -Path "C:\Users\jalee\Pictures\image5222.jpg" -Algorithm MD5

Algorithm       Hash                                                                   Path
---------       ----                                                                   ----
MD5             AC767C2FE38EE4E6E66172D137A0DFA4                                       C:\Users\jalee\Pictures\image...


PS C:\Users\jalee>
```

Figure 11 - Old image5222 Compared to Modified image5222.jpg

In Figure 10, I modified image5222.jpg by cropping out a portion of the image. I ran the same command as Figure 9 and got a new checksum. In Figure 11, I just manually looked at both the old and new hashes to compare whether they are different. I could have made a small program in notepad and run it to compare each character within each hash to each other, but I felt that would be too convoluted. But the old checksum would be invalid as the hash varies by the third character in the string ‘8’.

```
PS C:\Users\jalee\Pictures> $checksum = Get-FileHash -Path "C:\Users\jalee\Pictures\image5222.jpg" -Algorithm MD5
PS C:\Users\jalee\Pictures> $checksum.Hash
AC767C2FE38EE4E6E66172D137A0DFA4
PS C:\Users\jalee\Pictures> notepad C:\path\to\checksum_file.txt
PS C:\Users\jalee\Pictures> notepad C:\path\to\checksum_file.txt
PS C:\Users\jalee\Pictures>
```

Figure 12 - Attempt to Open MD5 file in Notepad

I attempted to open the file with the notepad however I got a message saying that the file did not exist. I went to the directory to check if the MD5 file was there, and it was not. So, I could not really open to check its content. I can only assume based on the command I used in Figure 9 & 11 that the file would contain the type of algorithm, the hash of said algorithm, and the path that hash is used for. Since there are 32 digits in the hash of the MD5 algorithm and 1 digit is equal to 4 bits, 32 digits * 4 bits = 128 bits for the hash. 128 bits / 8 bits = 16 bytes and if you do 128 bits / 4 = 32 hex digits or 0x80.

2.3 Task 3: Privacy assurance

2.3.1 Steps

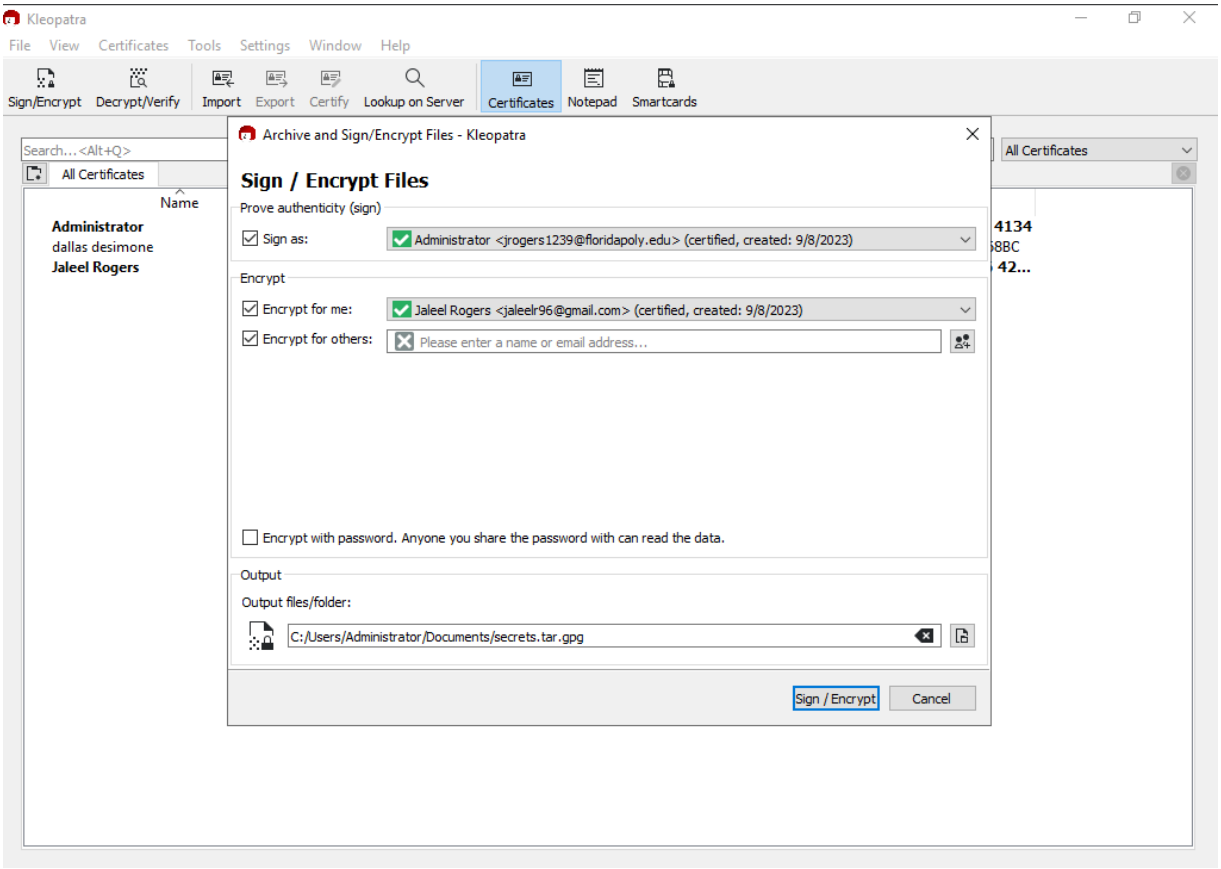

Figure 13 - Signing File with Public Key

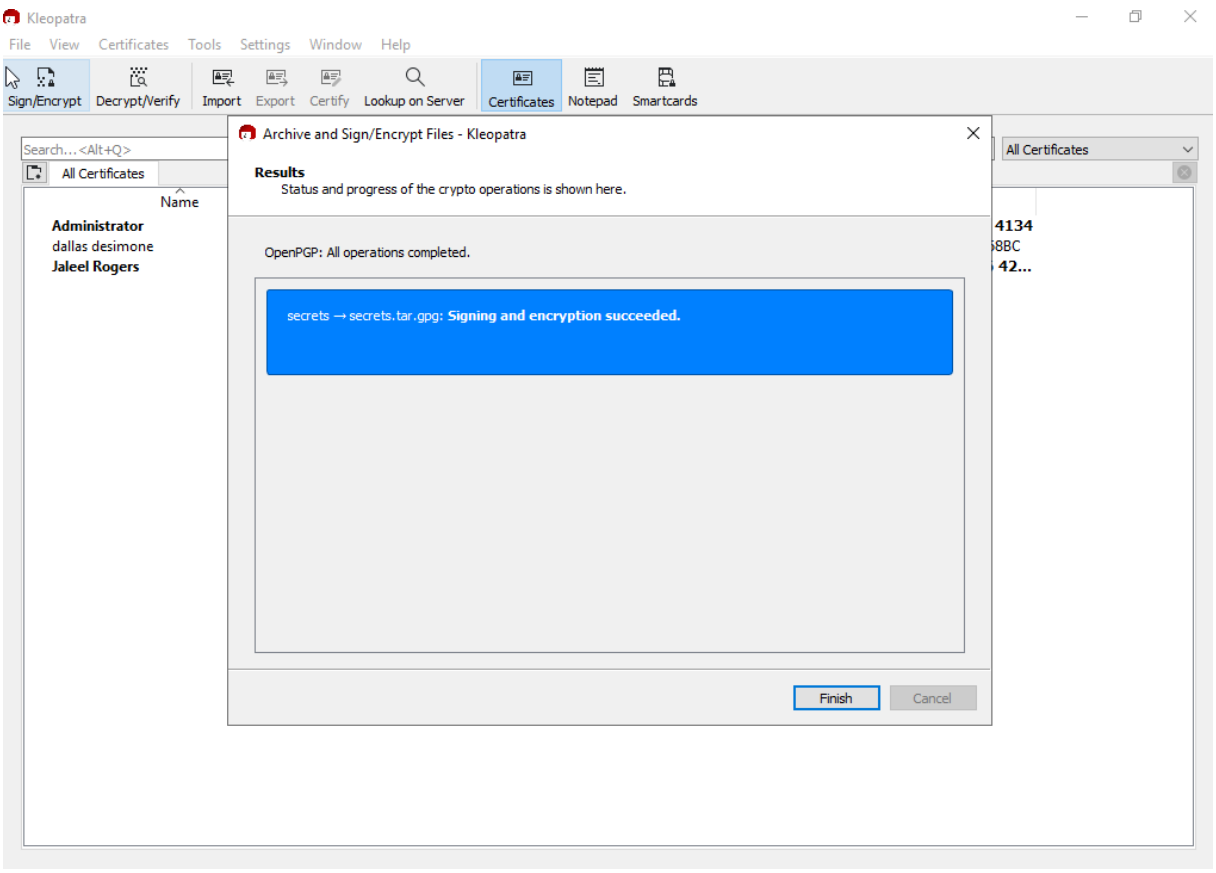

Figure 14 - Successful Signing of File

For Figures 13 & 14 I created a file and signed it with my certificate, allowing only my friend to decrypt it with his key.

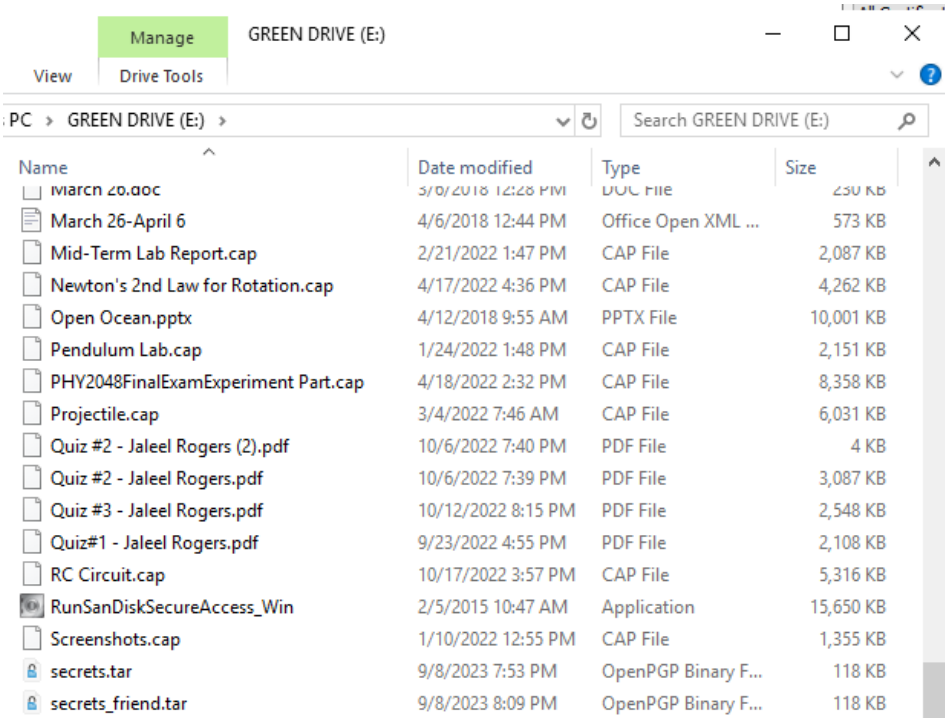

Figure 15 - Sending & Receiving Private Keys

Since I am using two of my own email addresses, I just stored the encrypted files for my friend and myself in the same location, a thumb stick.

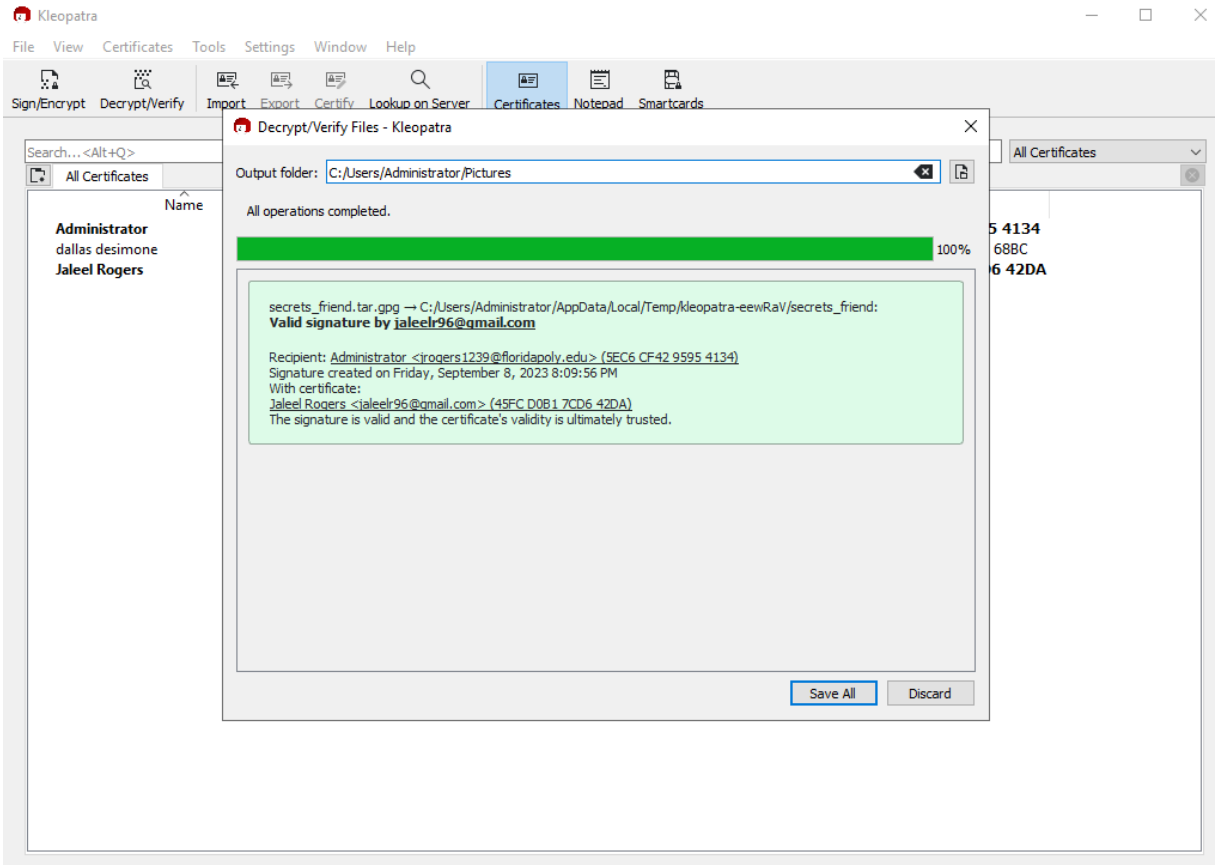

Figure 16 - Decrypting File Sent from Friend

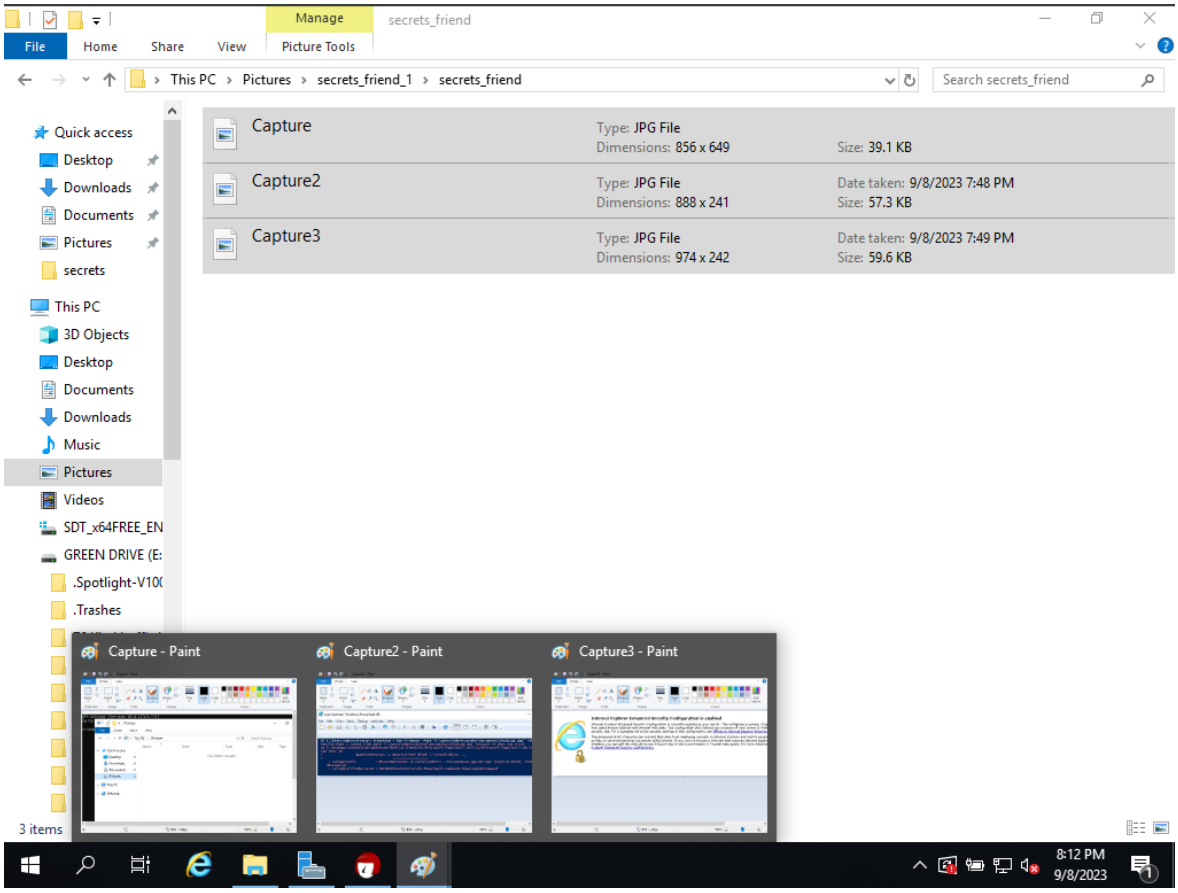

Figure 17 - Opening Friend's File

In Figure 16, I decrypted the file encrypted by my friend for me. In Figure 17, I opened the file and have all of them in view in Microsoft Paint.

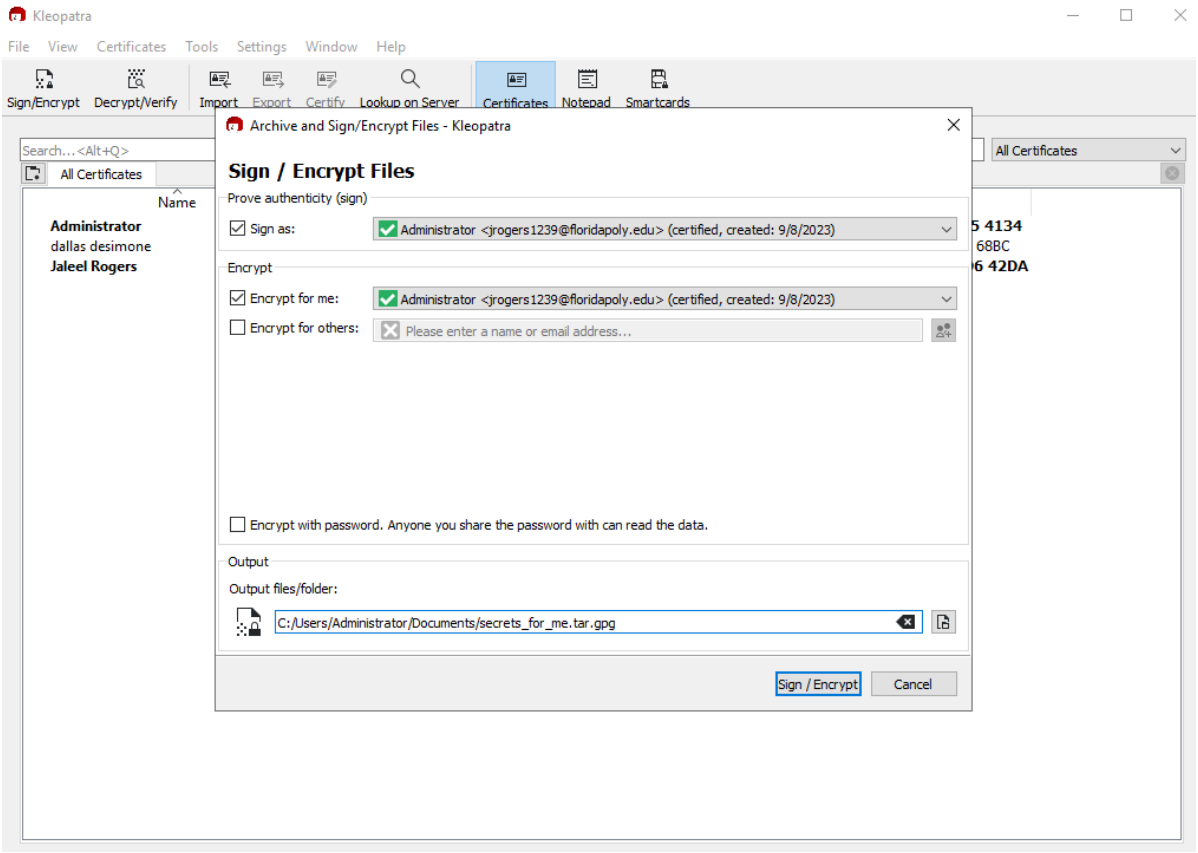

Figure 18 - Encrypting a with only my Certificate

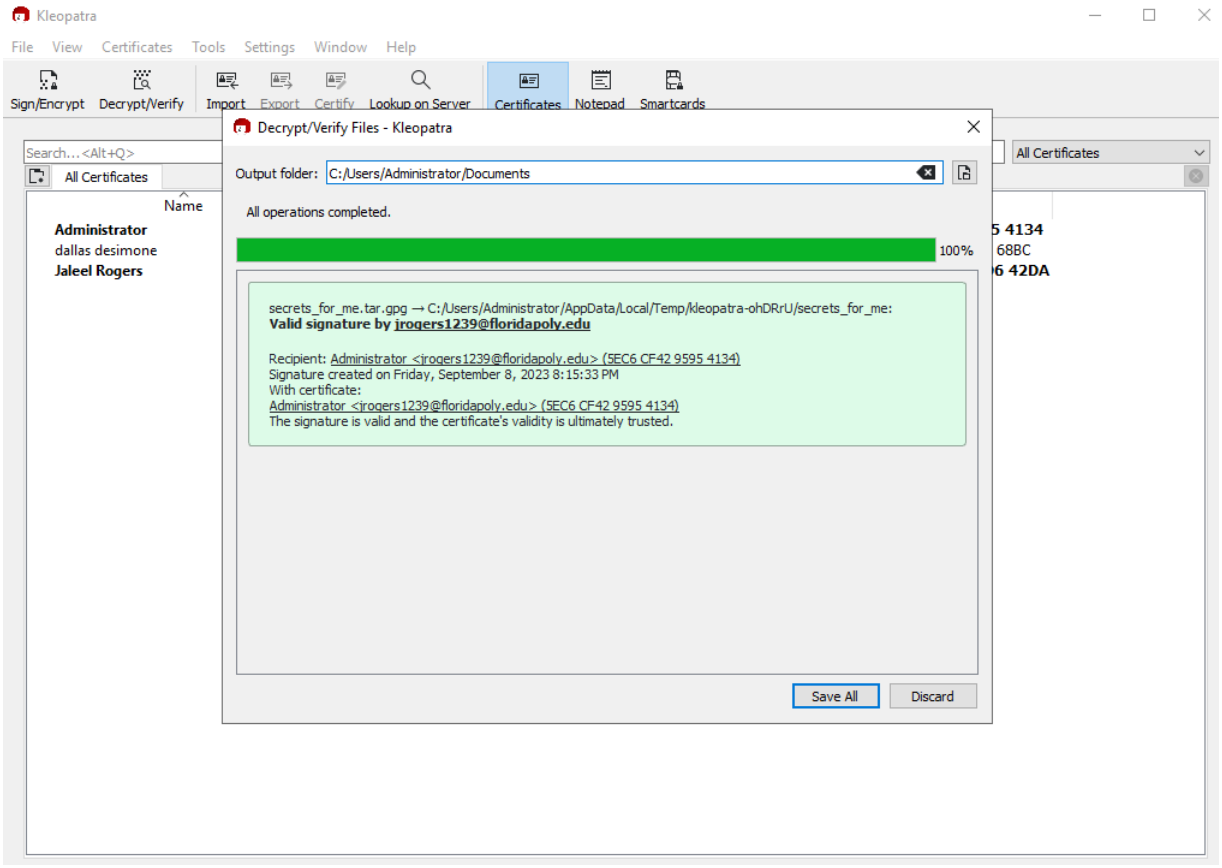

Figure 19 - Successful Decryption Using only my Certificate

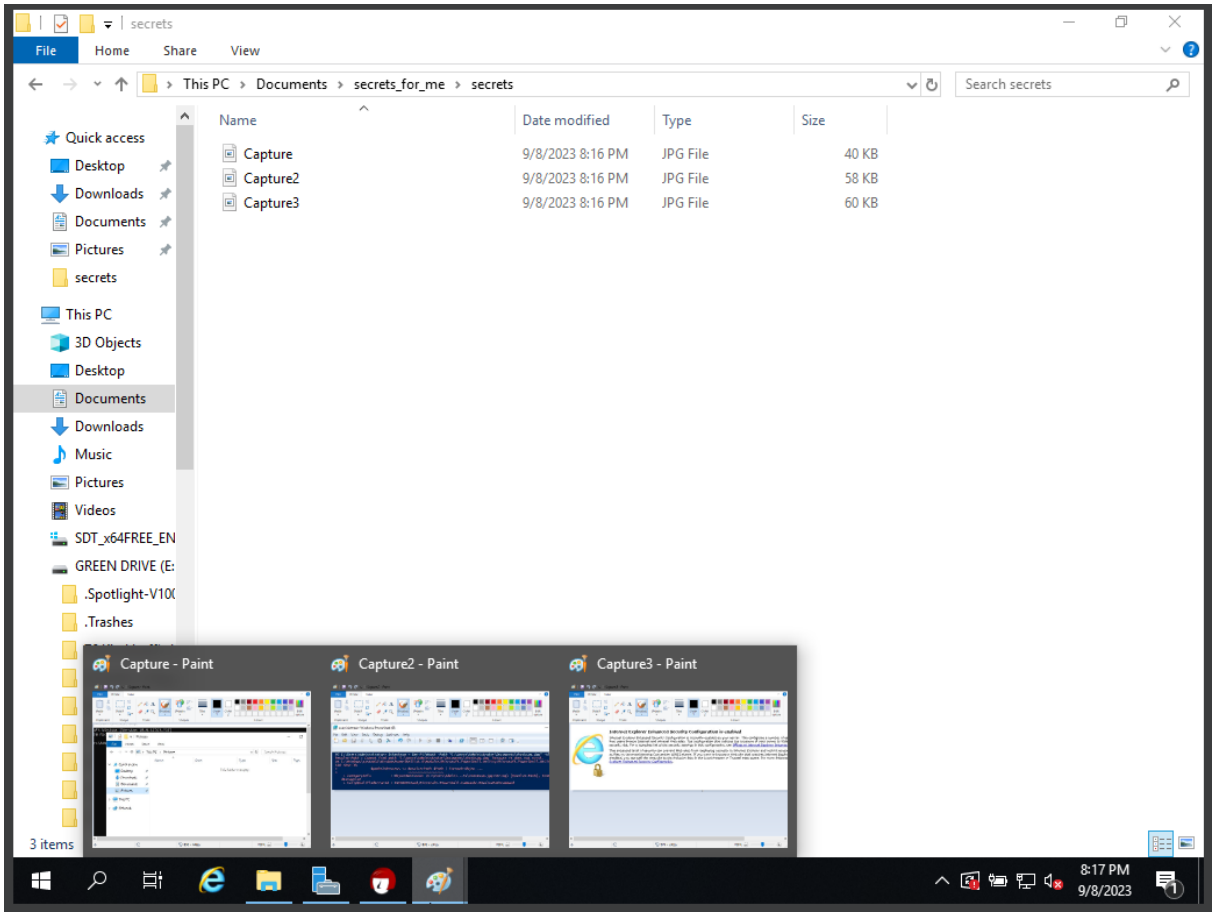

Figure 20 - Checking the Contents of Decrypted File

In Figure 18, I encrypted a file for myself that only I can open. Figure 19 is just a verification that the process was successful while Figure 20, is just the contents of the file I encrypted for myself solely using the Administrator's keys.

3 Issues or problems

I had two problems. The first was the creation of the checksum. After creating the checksum, I could not find its file. I tried the same command a few times with the same output but not MD5 file. I then tried it in my VM to see if it was just a problem of the location of my jpg. But I could not create the MD5 file, so I just guessed the contents of the file by looking at my output. The second problem I had was the strict browser rules of Windows Server 2019. Every time I wanted to go to a different site, I had to allow it to a rule sheet. This was especially annoying when it came to the process of logging into a website like Gmail for sending the encrypted file. My solution was to use a thumb stick instead, which was a fine alternative.

4 Conclusions

In this lab, the primary goal was enhancing file security using cryptographic techniques. The tasks to pursue this goal were securing emails and files using Gpg4win, creating a checksum for a file along with making modifications to the file verifying if it is still the original file, and file signage of different and same users. While there were some issues, Lab 02 provided hands-on experience in enhancing data security which will prove useful in ensuring and protecting data using digital assets.

5 References

Ufidon, & Wang, X. (n.d.). *comsec/labs/lab01 at main · ufidon/comsec*. GitHub. https://github.com/ufidon/comsec/tree/main/labs/lab02