National College of Ireland

Advanced techniques for storing passwords using image steganography and multi-level encryption with password splitting method

MSc Internship
Cyber Security

Nilisha B. Wandile
Student ID: x18134416

School of Computing
National College of Ireland

Supervisor: Ben Fletcher

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Nilisha Balkrishna Wandile	
Student ID:	X18134416	
Programme:	MSc in Cyber Security	**Year:** 2019-2020
Module:	MSc Internship	
Supervisor:	Ben Fletcher	
Submission Due Date:	12/12/2019	
Project Title:	Advanced techniques for storing passwords using image steganography and multi-level encryption with password splitting method	
Word Count:	5754	**Page Count:**19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.
I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature:	……
Date:	11/12/2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Abstract

In today's era of digitalisation and internet, user authentication and the use of passwords has been a centerpiece for decades. There is substantial research in the field of password security. Unfortunately, the methods of storing the passwords and its mechanism have failed to fetch enough attention from the researchers. In the past few years, a lot of researchers recommended the use of a combination of security methods like steganography and cryptography to elevate the security of the system. However, it still does not assure the security of the password. In this paper, we propose a system that implements multi-level encryption with decentralized and distributed storage of enterprises' sensitive information and offers multi-layered security to the system. In this design, we introduce a password security equation, *Password Protection = Encrypt + Split + Hide*. According to our equation, we encrypt the password using AES-256, split it into pieces and hide it behind the images using steganography using AES-256 adding another layer of encryption and store it in a decentralized manner. When the user logs in to the system all the pieces of the password are retrieved from all the steganographic images, merged and decrypted to form a legitimate password in one piece.

Keywords*: Image steganography, encryption, and password splitting*

SECTION I

1. Introduction

Every single day the technology is advancing however, the password remains unchanged as the most popular way of user authentication. To access any private and 'secure' system a user needs to go through a user authentication process. The user authentication process can involve the verification of one or more of some knowledge (like a password), something in possession (like an RFID access card) or inherence (like the fingerprint) of the entity [1]. Over the past few years, the primary mean of user authentication has been the passwords in the form of text. Users register on the systems by putting username and password to login to the system and gain access over it. The user is expected to remember this secret information to pass the user authentication process [2]. Considering the number of user accounts people have these days on social media, work-related emails, drives, and websites, etc. it could be very difficult for the user to remember all those passwords. Then users tend to use the same password for different systems putting its security on stake and then come RFID access cards and biometric user authentication in the picture, which are unique. But at the end of the day, the question arises on the security of the password storage. No matter what guidelines have been followed to create an ideally secure password, the question remains the same, is it being stored on a secure system in a secure manner?

Different systems store passwords in their database in their own style. It also depends on the type of database system being used. Having confidential information stored at the centralized database system has proven to be highly insecure from time to time. However, storing this confidential information in a distributed manner makes it difficult for the attacker to track it down to every single location and perform the attack. The maintenance and security of the password file in the database is still a point of concern in various fields. Many websites are not following the good storage practice which would ideally include encryption, good hashing algorithms, salting, and a large number of rounds. In the past, many renowned organizations like Yahoo, LinkedIn, United Nations and Sony Online Entertainment were attacked for ignoring these security measures [3]. Sensitive business information is at risk when it falls into the hands of unauthorized people with malicious intentions. Considering the easy availability of hacking tables and tools on the internet, to interrupt the primary level of security with attacks such as Dictionary attack, Rainbow table attack, Hybrid attack, Brute force attack and Smart brute force attack it is very difficult for these security measure to defend the system against such attacks when they are considered to be safe. It would be a mistake to rely on these measures completely assuming that the system is unbreakable. Also, looking at the advanced hacking techniques trending in the market these days, white hat hackers are coming up with innovative solutions to protect the system against them. To increase the security of the systems and keep the data transfer safe throughout the internet, various techniques have been designed such as Digital Watermarking, Cryptography, and Steganography [4]. However, the attackers are successfully being able to penetrate the system in some or the other way. Hence, it has become very crucial for the security researchers and professionals to come up with a stronger solution that is rigid enough to be penetrated and flexible enough to be accommodated in varied environments. The solution must be something that blends the existing security solutions and

enhances the security even more. It is possible with multi-layered security implementation using the decentralized password storage system.
This research revolves around a combined methodology that merges these security techniques with respect to the threshold cryptography system. Performing the combination of steganography and cryptography guarantees a high level of data security [5].

The rest of the document is organized as follows:

- Section II discusses about the existing research papers, their work, and findings around the use of steganography and cryptography for user authentication, database security, and threshold cryptography- password splitting/secret sharing. In the review of the existing literatures, we first review through some research work that focuses on image authentication, steganography. Then the research work that has proposed to blend the security measures like steganography and cryptography. Later in the rest of the part, we reviewed through papers that compared and evaluated the existing encryption algorithms and finally the secret-splitting or credential-splitting method with reference to the concept of Threshold Cryptography.
- Section III specifies the method and implementation of the proposed password splitting method using, encoding, multilevel encryption using AES 256 and steganography technique.
- Section IV concludes the paper suggesting the steps for enhancement of the proposed solution in future.

SECTION II

2. Literature Review

We can classify passwords as the most important and crucial resource being stored in the database. Hence, in the past extensive research has been done and a lot of researchers from around the world have published their research about its security, attacks on the database system, password attacks, innovative ways of performing those attacks and defense systems against them. Some recommended the use of images or separate use of steganography and cryptography for user authentication, while some suggested the combination of steganography and cryptography for the same. Considering the fact that each one of them had some strengths and weaknesses that sometimes they significantly shielded the system or failed due to some gaps in their method of implementation.

2.1. Image authentication

In [6] the authors proposed an image-based authentication system with the use of steganography. Highlighting the issues with current authentication systems such as, having to remember the alphanumeric passwords and people having a number of user accounts on the internet they accidentally or lazily set the same password for multiple user accounts which is a potential security threat. They suggest an innovative system where the password is hidden behind a secret image using steganography and stored in the database. At the time of sign up, a traditional process is followed but the user is also asked to browse and provide a secret image that has the password embedded in it. The user details with the password embedded secret image stored in the server database. Next time the user logs in, the login interface sends the

username in plaintext and the secret image is sent in the bit stream. They argue that in case the password and the secret image is known to the attacker, it would still be impossible for them to penetrate through the system as the database not only compares the images and the passwords but also the dimensions of the image [6]. However, in [7] a unique way has been introduced for user authentication. In their exceptional way of authentication, they used a click-based graphical password scheme which is a cued-recall graphical password technique. Basing their research on Passpoint [8], which was proposed in 2005, where the password was composed of numerous points anywhere on the image. They also suggested a "robust discretization" system with a variety of conflicting grids enabling login attempts to be recognized that strongly resembled with the correct form and translating the password inserted into a key for cryptographic authentication. Hence, CCP proposed an alternative way to it using the hotspot technique. Rather than clicking on multiple points on a single image, the user has to click on one point on each image as there would be multiple images to be clicked by the user. The authors conclude by arguing that the CCP is securer than the graphical authentication methods that have been introduced previously. Resulting in which, it makes it difficult for the attackers to acquire the correct sets of images for the user and then analyze the correct hotspots on each of those images [7].

2.2. Steganography and cryptography

In the past, a lot of researchers have directed their research towards the combination of two or more security measures. Therefore, steganography and cryptography have been the most relied one by them. Considering the fact, that steganography itself applies cryptography while hiding the text behind. Hence, [9] explain the way of using steganography and encryption to make the data secure. They suggest that Elliptic Curve Cryptography can be used for image encryption Huffman Coding method for image steganography, and Discrete Wavelet Transform for image compression. In their work, they argue that the Elliptic Curve Cryptosystem is the most secure one out of all the existing cryptosystems. Most importantly by comparing the Elliptic Curve Cryptosystem with Diffie-Hellman or RSA, they highlight that it provides the same level of security with much shorter keys. Provided that, it offers higher speeds, lesser power consumption, bandwidth savings, and storage efficiencies. This could particularly be useful where processing capacity, bandwidths, power availability or storage are required [9]. As well as in [10] the researchers are using the combination of security measures like steganography and cryptography to make the system secure. But with a different method as they proposed a blend of steganography with the use of discrete cosine transform (DCT) and cryptography using the one-time pad or vernam cipher implemented on a digital image and measuring the quality of the image using the peak signal to noise ratio (PSNR) and the quality of the extraction of the decrypted message using Normalized Cross Correlation (NCC). They proposed an embedding algorithm to perform this operation. The results were measured in Peak Signal to Noise Ratio (PSNR) and Mean square error (MSE). The smaller value of the MSE depicted the better quality of the output image. In the final result, the average value of the MSE was 0.50232 which proved to be a reliable way of implementing a combination of cryptography and steganography. Also, they suggested that encryption methods such as AES, RSA, and DES could be considered to be combined with steganography in future [10].

Few researchers tried to go one step ahead and use the renowned and stronger encryption algorithms as they are considered, such as AES and RSA. While few tried to modify the existing one or combine them with other algorithms like SHA512 and MD5. According to [11]

integrating AES and RSA together and applying steganography elevates the security of the system. In their research, they propose a combined and alternative approach to secure the system using cryptography and steganography. They used two encryption algorithms, AES (with symmetric keys) and RSA (with asymmetric keys) with image steganography. To justify their choice, they mention that combining these three techniques together helps to construct a strong communication system based on steganography that can sustain several types of cyber-attacks, reverse engineering and detection systems. The entire operation has three main components: sending, transmitting and receiving. The sender uses three inputs for the communication, secret data to be transmitted, cover image to hide the secret data and the public RSA key of the receiving party. By using the private RSA key, the steganographic image, receiver decrypts the secret data. Their system used AES-256 for primary encryption. With each new communication string, the encryption key (256 bits) is formed and consists of two parts: calculated and random, both being128-bit in size. a pseudo-random pattern generator (PRPG) seeded with either variable or constant data is used to generate the 1st half of the AES Key which is a random part of the key. Non-volatile color information (NVCI) is used to calculate the computed part of the encryption key which is the 2nd half of the AES Key. Using a powerful algorithm such as SHA512, MD5, the high order bits of each color channel (RGB in a color image) are hashed. Also, it needs to be noted that only half of the hash value is used in the 2nd half of the AES key. The data is embedded in the image using the data rearrangement method and LSB pixel mapping. The steganographic image travels through any communication channel like email, network, file sharing, etc. and in the end, the receiver's system processes the VCI part of the steganographic image to decrypt it. The receiver's system recovers the 1st and the 2nds half of the AES key. In the end, the receiver decrypts the intended secret data after retrieving both the halves of the AES key [11]. Research done in [12] also introduced the same combined approach to ensure the security of the data by merging steganography and cryptography using the AES-256 algorithm but they also modified the existing AES algorithm. They mentioned that security operations take place in parts to achieve two levels of security. In the two parts of their research, in the first part, they modified the AES algorithm to accommodate the steganography method that they implemented, and it is called AES_MPK algorithm and in the second part, the same AES_MPK algorithm is merged with the steganography algorithm to hide the encrypted secret behind the image. To perform the first part of the operations, the authors required the output in the form of MPK digits as MSLDIP-MPK and PVD_MPK methods use the MPK digits for hiding the data. Hence, the revised AES algorithm was called AES_MPK algorithm. And in the second part, they encrypted the message M using AES_MPK algorithm using the key K and produce the ciphertext. Later the same ciphertext is hidden behind the cover image C using the MSLDIP-MPK and PVD_MPK methods to produce the steganographic image S. They conclude that with the implementation of their proposed method it would be easier to transmit the data even over the open channels as the ciphertext would not fetch the unwanted attention as it is hidden behind the image and it is capable of hiding a large amount of information than existing methods. Also, their system provides two layers of security making the system even more secure [12].

2.3. Evaluation of cryptography algorithms

While giving attention to the multi-layered and decentralized security measure, it is very crucial to choose the method that is more reliable and flexible. Some researchers have written papers where they have compared and evaluated the existing cryptography algorithms. Like in [13] an evaluation operation was performed to compare the top encryption algorithms, the authors

compared these algorithms with respect to the encryption, decryption and packet size. Referring to the results shown below in figure 1, they concluded that AES is much better than RSA and DES.

S.NO	Algorithm	Packet Size (KB)	Encryption Time (Sec)	Decryption Time (Sec)
1	AES	153	1.6	1
	DES		3.0	1.1
	RSA		7.3	4.9
2	AES	196	1.7	1.4
	DES		2.0	1.24
	RSA		8.5	5.9
3	AES	312	1.8	1.6
	DES		3.0	1.3
	RSA		7.8	5.1
4	AES	868	2.0	1.8
	DES		4.0	1.2
	RSA		8.2	5.1

Figure 1: Comparisons of DES, AES, and RSA of Encryption and Decryption Time [13]

Whereas in [14] the authors evaluated these cryptography algorithms based on the encryption time, decryption time and avalanche effect. According to the results they determined that AES is the best choice where confidentiality and integrity are of the highest priority as AES scored the highest score among the rest.

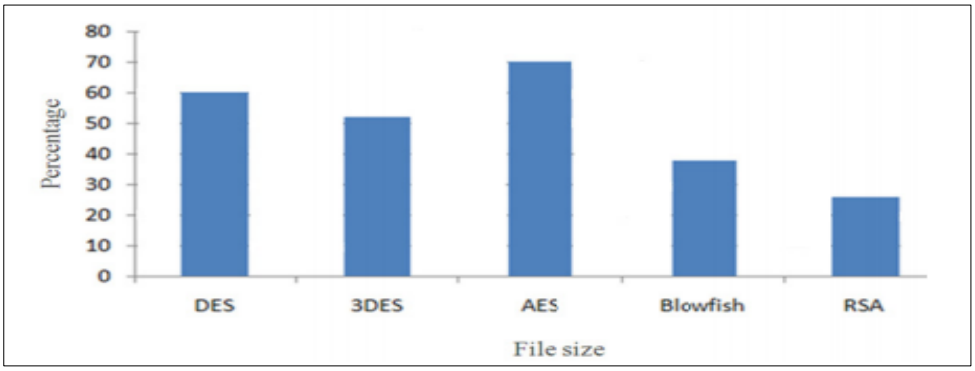

Figure 2: Avalanche effect for DES, 3DES, AES, Blowfish, and RSA [14]

2.4. Secret sharing and password splitting

The secret splitting technique was first introduced by Adi Shamir in 1979 in his research work titled “How to Share a Secret” which was referred to as Threshold Cryptography [15]. Based on that, many research papers were published in the past and referring to the same technique, in [16] RSA (Rivest, Shamir, Adleman) the security division of EMC introduced a new method of password protection- Distributed Credential Protection (DCP) in their white paper. In their system, it allowed the user to split their password into two pieces and store it on two different servers. The credentials were scrambled using cryptography, later randomized and stored across two servers after splitting them into two halves.

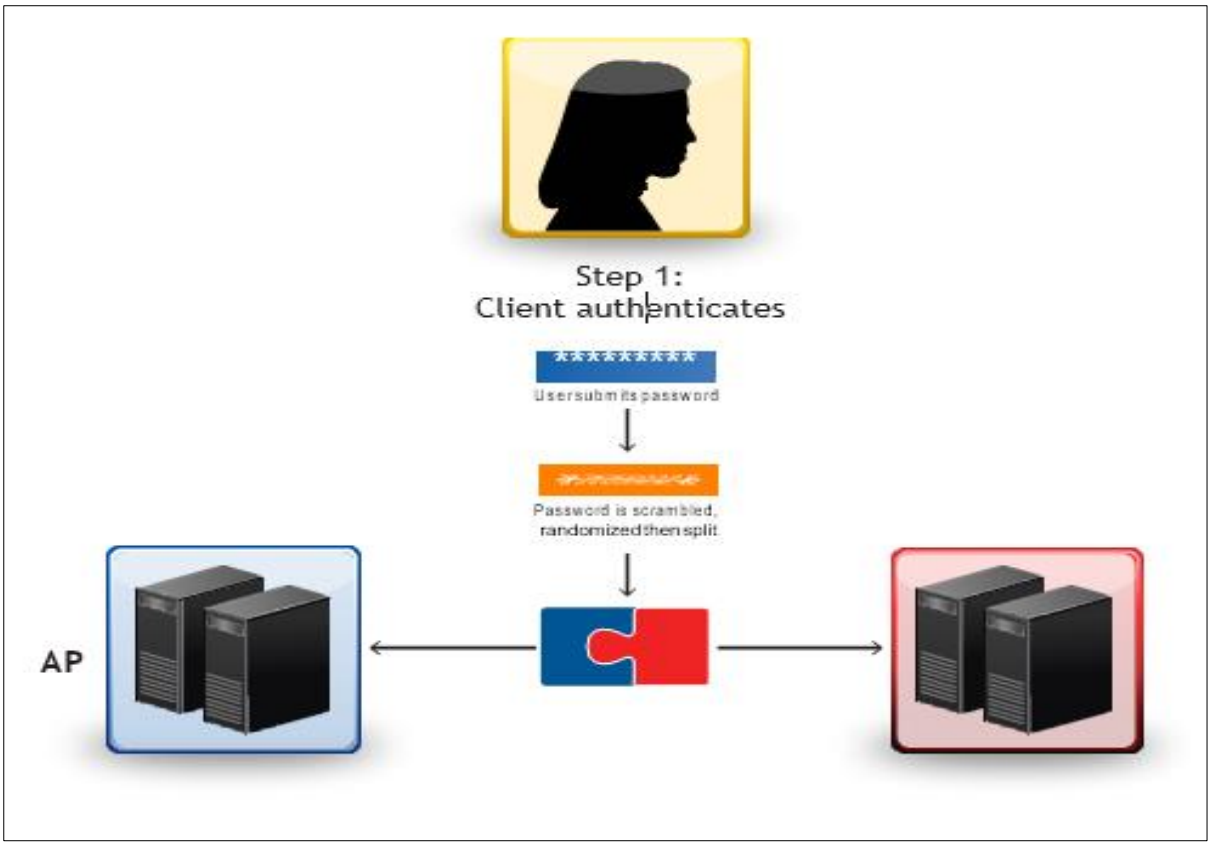

Figure 3: Gaining access to secrets with RSA Distributed Credential Protection- Step 1 [16]

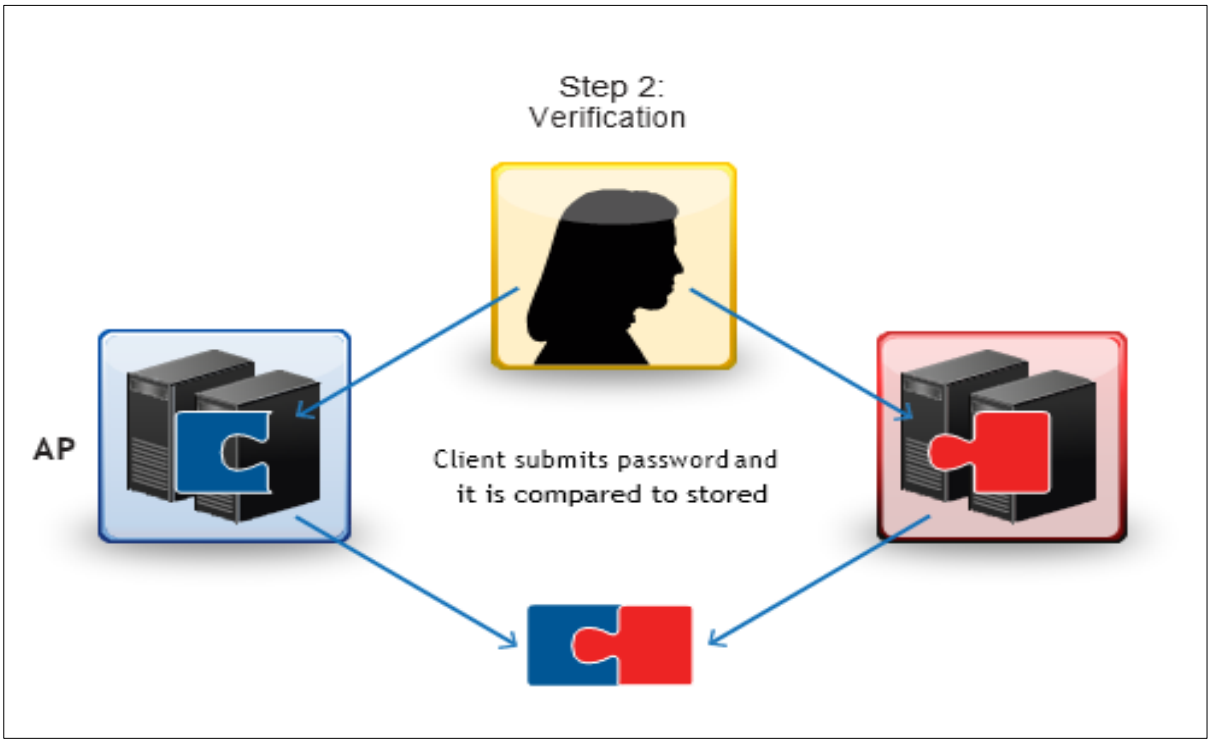

Figure 4: Gaining access to secrets with RSA Distributed Credential Protection- Step 2 [16]

Today most of the systems store confidential information in a centralized manner, and the attacker only has to compromise one server to extract the entire piece of sensitive data. If an

organization has confidential data in a primary point of a compromised system, it carries all the potential risks arising from an internal infringement. RSA highlighted the centralized password server or an authentication server as the primary point of compromise for the attackers. Regardless of the credentials being encrypted, hashed or salted attackers have always found ways to breach through such security measures. Their system offers the re-randomization of the secret as it periodically refreshes the randomness used to split the credentials. This method, recognized as proactive cryptography, makes it possible to retrieve even those that go undetected from temporary breaches. Hence, even if the attacker is successful to hack the server 1 and later in time hacks the server 2, they still have hacked only the server 2 alone [16].

SECTION III

3. Research Methodology

The execution of the proposed research was divided into 3 phases- encryption and encoding, steganography and decryption. In the first phase, the password was encrypted using AES-256 and encoded in Base64 i.e. multiple of 4, as we split the password into 4 pieces. In phase II we performed steganography by hiding the encrypted pieces of password behind the images by implementing another lever of AES-256 encryption. In the final phase III, we retrieved the password by decrypting the pieces of the password and combine them in the correct order.

According to the analysis done in the literature review, it is clear that keeping the sensitive data at the decentralized location by implementing a stronger encryption algorithm like AES-256 and merging it with other security methods like steganography upraises the security of the system. Hence, the research focuses on the security of the storage system by bringing in the attention towards the manner in which the sensitive information is stored, encryption, image steganography, and password splitting method.

3.1. Database Security

As we have seen how important the word 'data' itself is and its security in today's digital world, it is needless to say how important the storage system and its security would be. The security of the database is based on the way the data is being stored in the database. Though the trend of digitalization has taken over the world a couple of decades ago, researchers have been talking about it since 1978 or even before that. As the authors in [17] brought light towards the storage of the data, the ways data can be extracted from the database and its security. Once again, years later in [18] the researchers raised their concern on the security threats on different database systems like RDBMS, ODBMS and object-oriented database system as different organizations implement different database storage methods. There are several security methods in the market for database security, but our focus is on the decentralized and distributed storage of the data.

3.2. Encryption

Cryptography and encryption is not a new concept anymore. It is considered to be the best most crucial security method to defend against the attacks on sensitive data which is our password in this case. But it's not enough just to encrypt the passwords. The process of authentication should be accurate and with a better algorithm applied. The algorithm method should be able to accommodate itself according to the requirements and stronger enough to be implemented. In our proposed system, we are using AES-256 which has proven to be the best choice where confidentiality and integrity are important [14]. AES has a key function, i.e. number of rounds

and apart from safety its cost-effectiveness. The key's length determines the number of rounds in AES. AES uses varied bits keys like 128-bit, 192-bit, and 256-bit and uses 10, 12 and 14 rounds respectively [19] [20]. AES offers more efficiency and feasibility among other encryption algorithms like DES, RSA, and 3DES. Also, the only effective attack against AES is considered to be a brute force attack [20].

3.3. Image Steganography

The main goal behind steganography is to hide the sensitive information behind a cover media file and transmit it to the intended receiver. Images are considered to be the best medium for steganography. In our proposed system we are using the images as the cover medium. The more image looks perfect the lesser it fetches the attention of the attackers. We are using high-quality images to enable them to contain more data and not let the image quality get affected.

3.4. Password Splitting

Password splitting is the most important part of our proposed system. Implementing encryption, encoding and steganography would be of no use if we do not split and store these passwords in a decentralized and distributed manner. As the main concept that we are trying to coin through this proposed system is password splitting. It is very important to split the password and store it in pieces in different images to make it difficult for the attacker to perform an attack and steal the passwords even if they are successful to penetrate through the system.

4. Design Specification

In the proposed system, we did not make any modification in the existing encryption algorithms. Instead, we used the same algorithms by coalescing them with multiple algorithms at several stages of the program to add multiple layers of security in the system and split it into parts to store them into pieces at different locations which are our steganographic images. Through this research, we introduced a new security equation, Password Protection = Encrypt + Split + Hide. Figure 5 below depicts the design architecture of the proposed system.

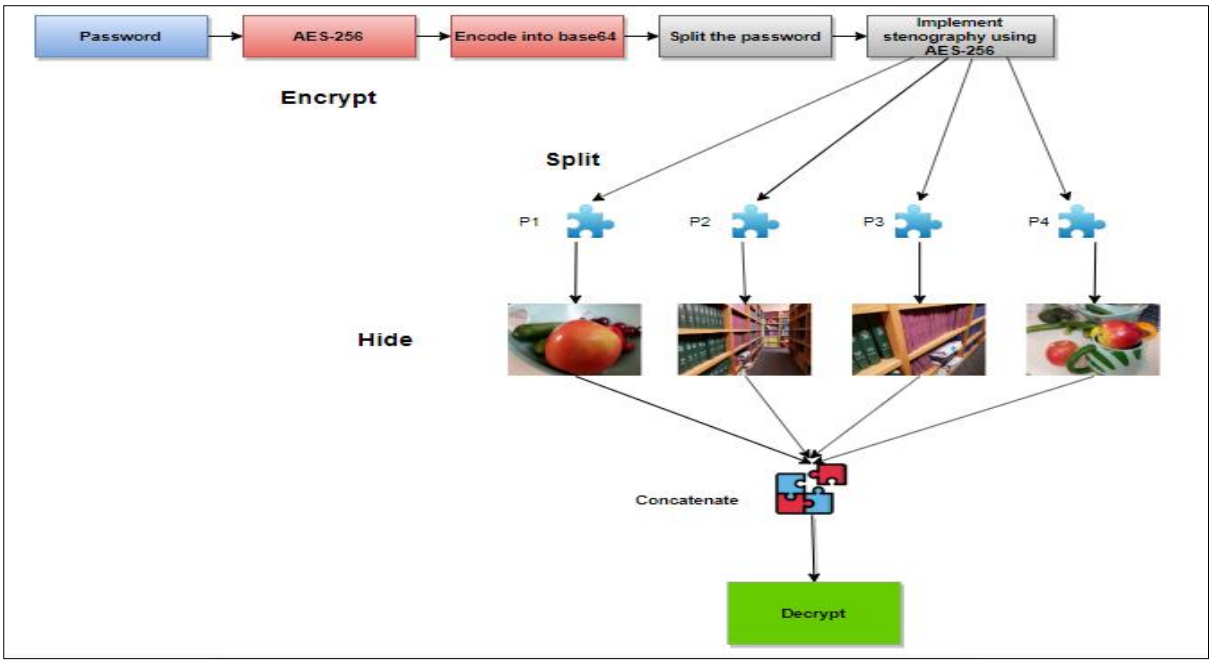

Figure 5: Architecture of the system

By implementing this, we are making sure that the system is still safe even if the attacker is able to penetrate through the system. Suppose, an attacker is able to hack the system and even gains access over any one of the steganographic images it would be very difficult for them to decrypt the data as it has been encrypted using AES-256 algorithm which uses the iterative approach with block size of 128 bit and key size of 256 bit where it takes 14 rounds for the key. The larger key increases the security as well as the complexity of the cryptography algorithms [5]. Also, they would still not be able to get any password in a single piece as they would have to go through each and every single image and decrypt it which is nearly impossible as we have applied stronger encryption algorithm at multiple levels. The section below illustrates the algorithm that we have followed for the proposed system.

4.1. Algorithm for the proposed system

Step 1: Start

Step 2: Enter the secret key

if the key is in a valid form move to step 3

else,

display message- Please enter the key in the correct form- (A-z, a-z and 0-9)

else,

exit the program after 3 failed attempts

Step 3: Confirm the secret key

Step4: If the key is correct move to Step 4

else,

display message- Failed! Please try again.

Step 5: Apply AES-256 encryption

Step 6: Apply base64 encoding

Step 7: Perform steganography using AES-256 encryption

Step 8: Enter the key to decrypt the password

if the key is correct move to step 9

else,

display message- Wrong key entered

else,

exit the program after 3 failed attempts

Step 9: Display the decrypted password

Step 10: Perform step 1 to step 9 for the next password in the list

Step 11: Exit

5. Implementation/Solution Development

In this section, we describe the implementation method of the proposed system and its flow. The ultimate aim of this research was to find a way referring to which the passwords can be protected using multi-level encryption and password splitting techniques on steganographic images. To achieve that and demonstrate the operations and results at each step, we required Python version 3.7, Spyder 3.7.4 which is a Scientific Python Development Environment and Anaconda 3.7 to display the results in the console.

The data required to implement this research is a list of user passwords to be split and a few images to perform steganography. To perform the proposed operations on those passwords we used an opensource credentials repository available on the OWASP portal. The data set is called OWASP SecLists Project [21], which is licensed under the MIT license. Also, the opensource images to perform steganography are taken from the Center for Statistics and Applications in Forensic Evidence (CSAFE) [22]. The results were analyzed based on the output (steganographic) images and being able to retrieve the encrypted passwords by decrypting them and concatenating them in the correct order with the correct pieces. The flow of the system is illustrated in figure 11. The program begins with Phase I where the user is asked to enter a secret key and confirm it again. It is possible to put a static key according to the requirement but for the illustration purpose, we have kept it dynamic. The password from the password file is picked up and encrypted in AES-256 which is later encoded into base64 for splitting purpose. The system also performs salting by default on the password. Post this operation, the encrypted password is split into 4 pieces to be stored inside the images. Here, we enter into Phase II where the steganography operation is performed using AES-256 encryption and moves to Phase III. In the last phase, the user is asked to put the secret key again to decrypt the encrypted password stored in the steganographic images. Once, the user puts the correct key, the password is displayed in plaintext and moves to the next password in the list.

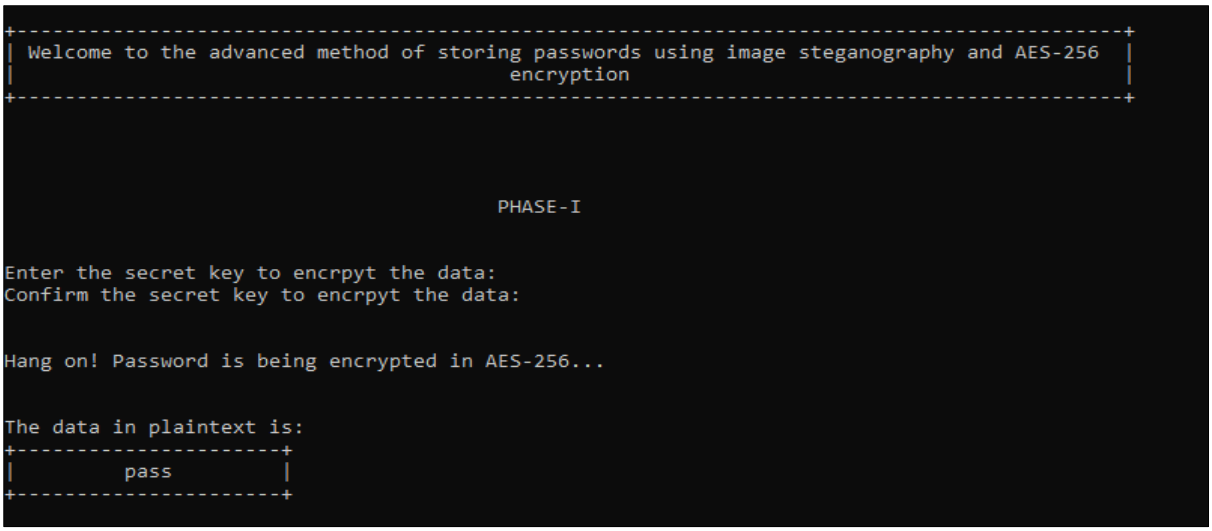

Figure 6: Phase I- First level of AES-256 encryption

Figure 7: Phase I- AES-256 encryption and Base64 encoding

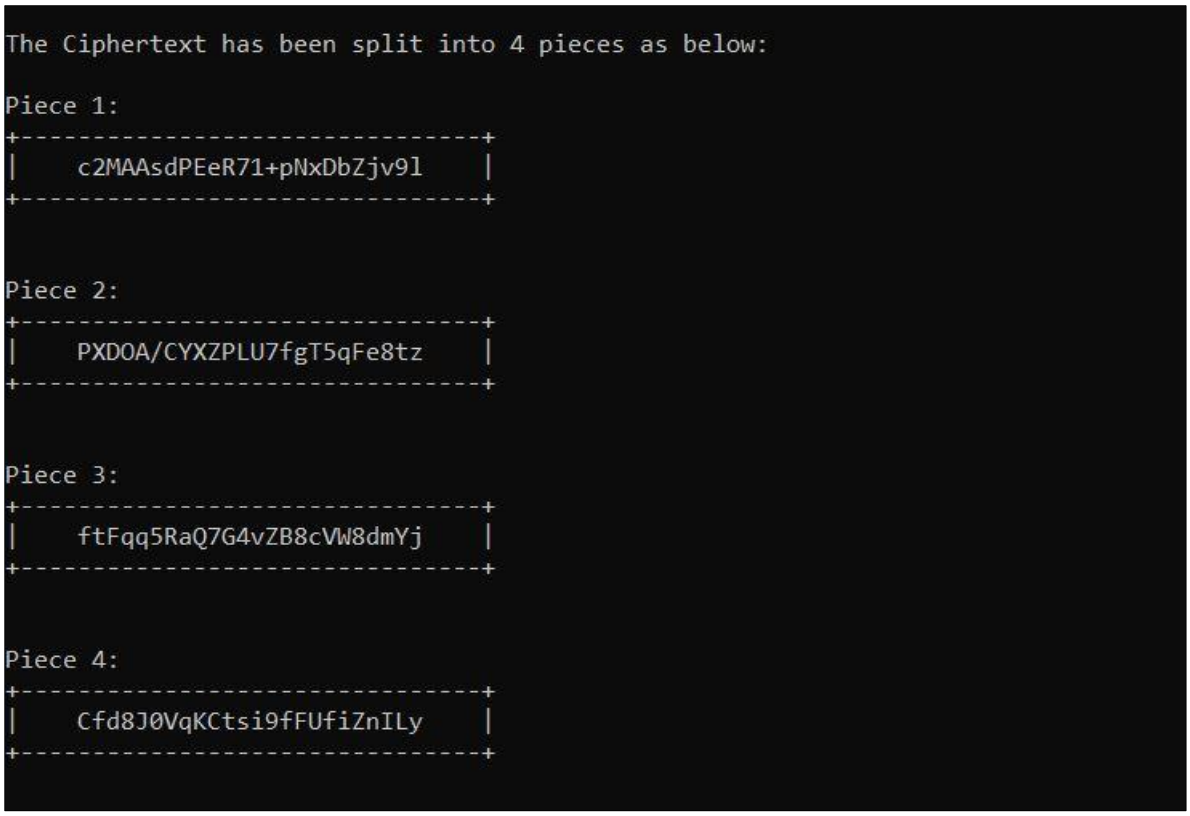

Figure 8: Phase I- Password splitting

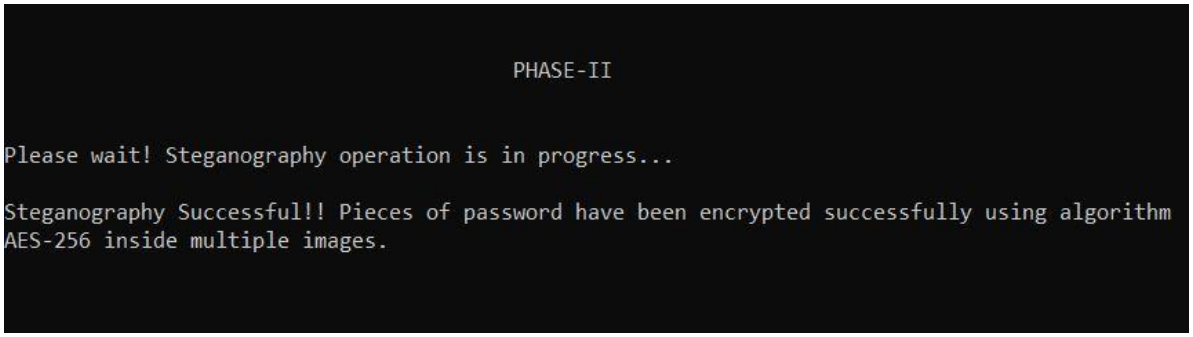

Figure 9: Phase II- Steganography operation

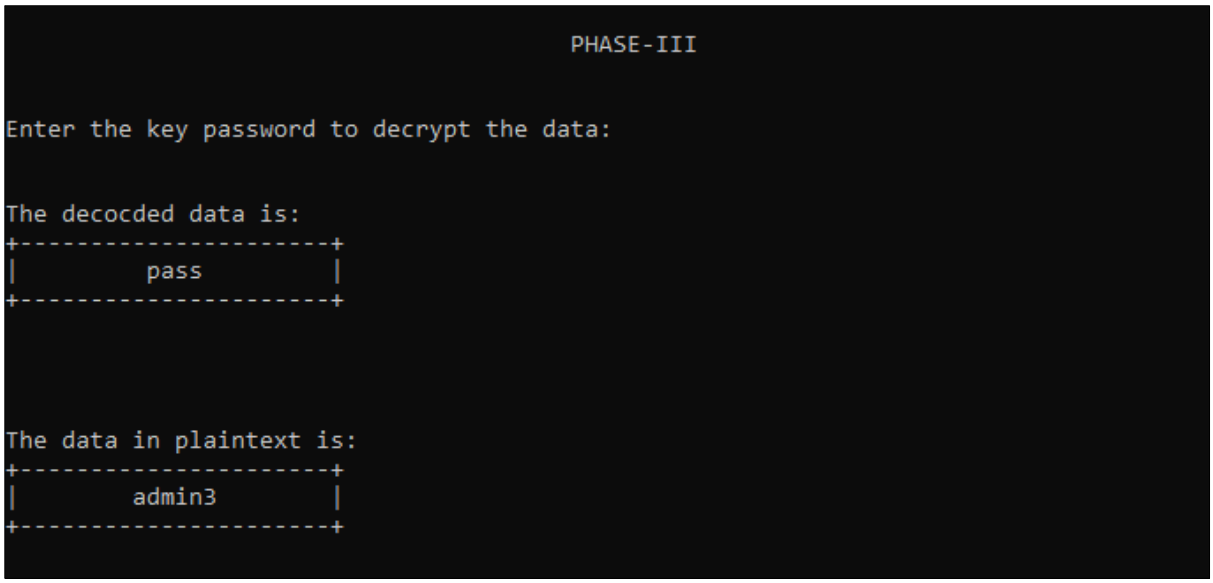

Figure 10: Phase III- Password decryption

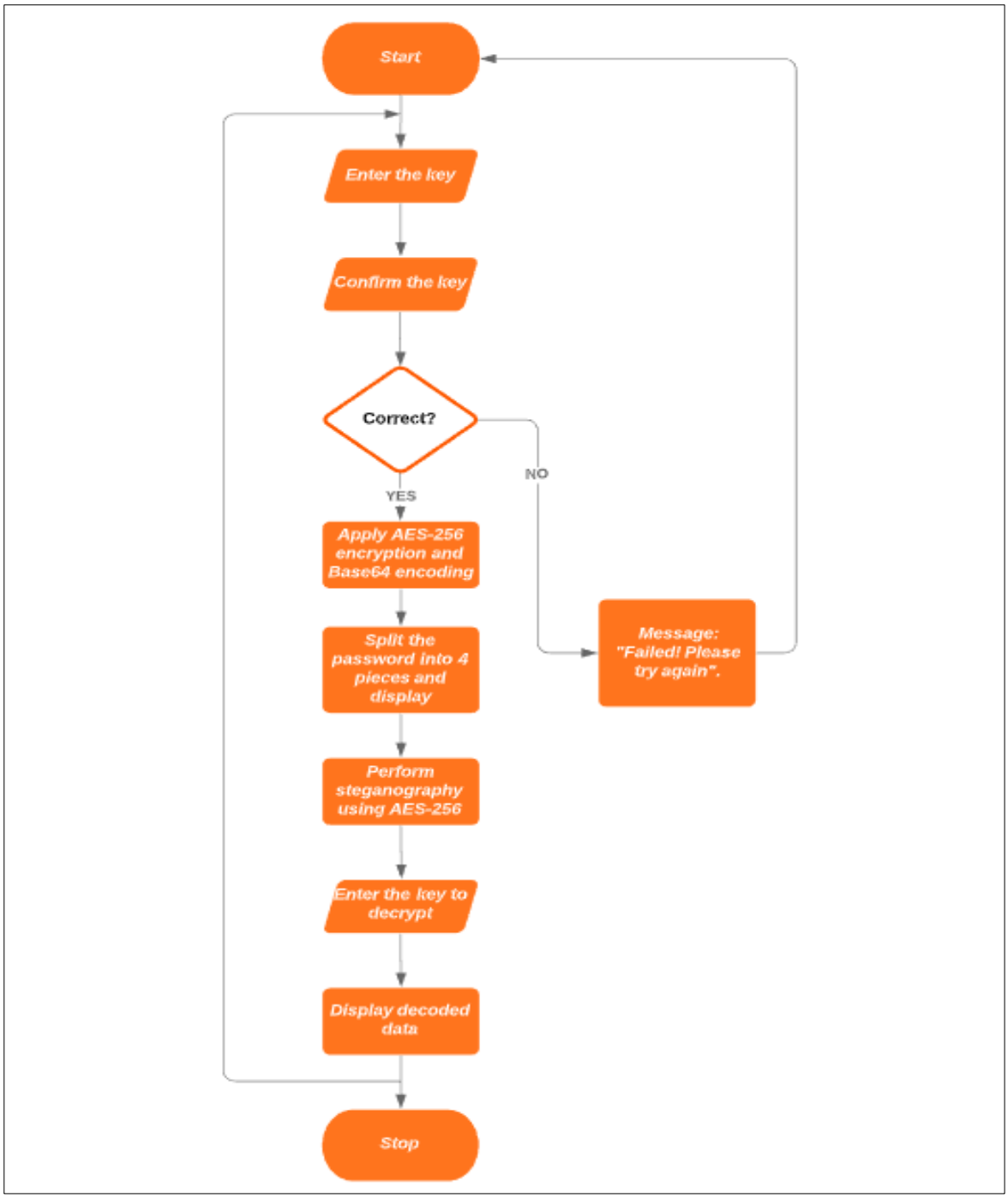

Figure 11: Flow of the proposed system

6. Evaluation- Case studies

To evaluate our system, we tested our program on various operating systems. Python being cross-platform language the code is expected to run on all platforms. Hence, it is not really required to test them on other operating systems [23]. But the configuration of the system might affect the performance, hence we tested the proposed system on Windows 10, Mac and Linux.

Case Study 1: Windows 10

The proposed solution was tested on the 64-bit Windows 10 operating system with 8 GB RAM and Intel(R) Core(TM) i5-8250U CPU with a 1.60 GHz processor. And the average time it took to perform the entire operation on 0.00012 MB data was 0.534030303 min.

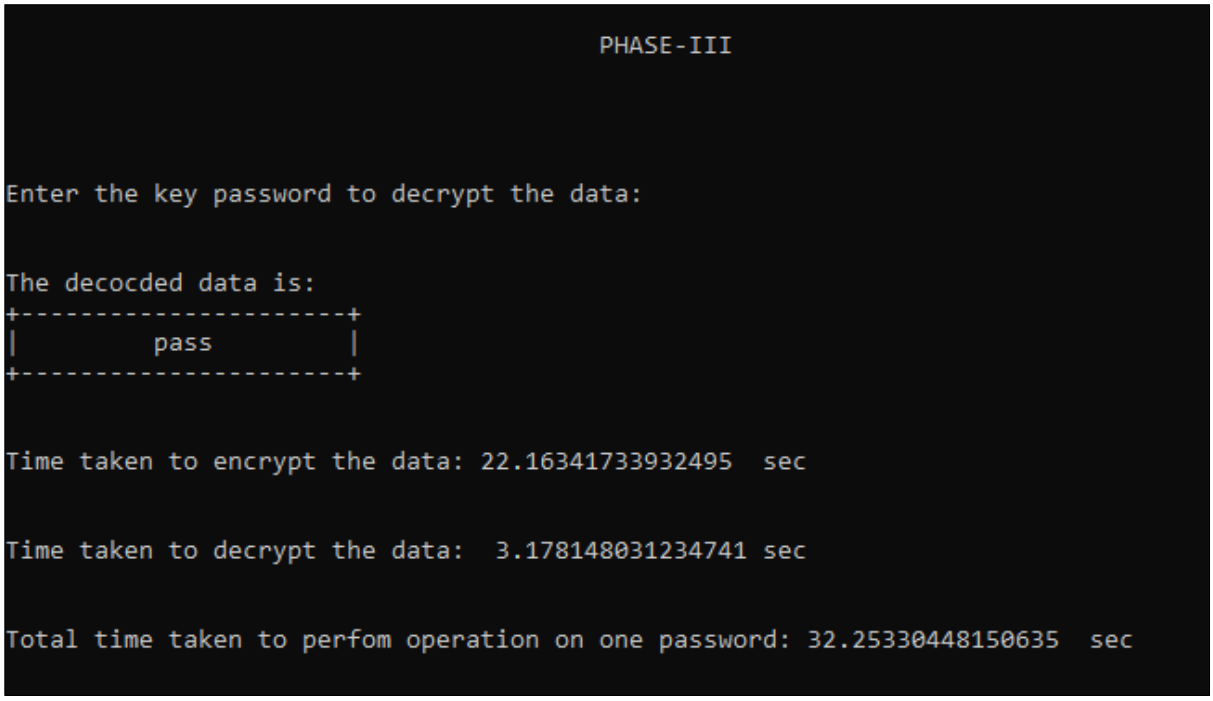

Figure 12: Phase III- Performance report on Windows 10

Case Study 2: Mac

After testing the proposed system on Mac operating system with 2.3GHz dual-core Intel Core i5, Turbo Boost up to 3.6GHz, with 64MB of eDRAM, 128GB SSD configuration which was lesser than the windows system the performance of the system got lowered as the average total time taken to perform the entire operation was 0.682590909 min.

Case Study 3: Linux

We also tested the system on a Linux machine with the same configuration as the MacOS, 2.3GHz dual-core Intel Core i5, Turbo Boost up to 3.6GHz, with 64MB of eDRAM, 128GB SSD, but the performance was better than mac operating system. Here, the total average time was 0.593454545 min.

Sr. No.	Password in plaintext	Size of the password in MB	Operation time on Windows in min	Operation time on Mac in min	Operation time on Linux in min
1	pass	0.000004	0.5375	0.6861	0.6456
2	admin3	0.000006	0.4911	0.6588	0.6021
3	q1w2e3r4t5y6	0.000011	0.6943	0.6688	0.5797
4	samantha1	0.000009	0.5323	0.6476	0.5797

5	spongebob1	0.00001	0.457	0.6604	0.5732
6	1qaz2wsx3edc	0.000011	0.4445	0.7087	0.5771
7	charlotte	0.000009	0.5928	0.6922	0.6319
8	forever21	0.000009	0.5175	0.6334	0.5763
9	MaprCheM56458	0.000012	0.6568	0.6698	0.5723
10	a1s2d3f4	0.000008	0.5098	0.6424	0.6074
11	e10adc3949ba59abbe56e057f20f883e	0.000031	0.4405	0.8403	0.5827
		0.00012	0.534030303	0.682590909	0.593454545
		Total	**Average**		

Table 1: Performance report on operating systems: Windows 10, Mac and Linux

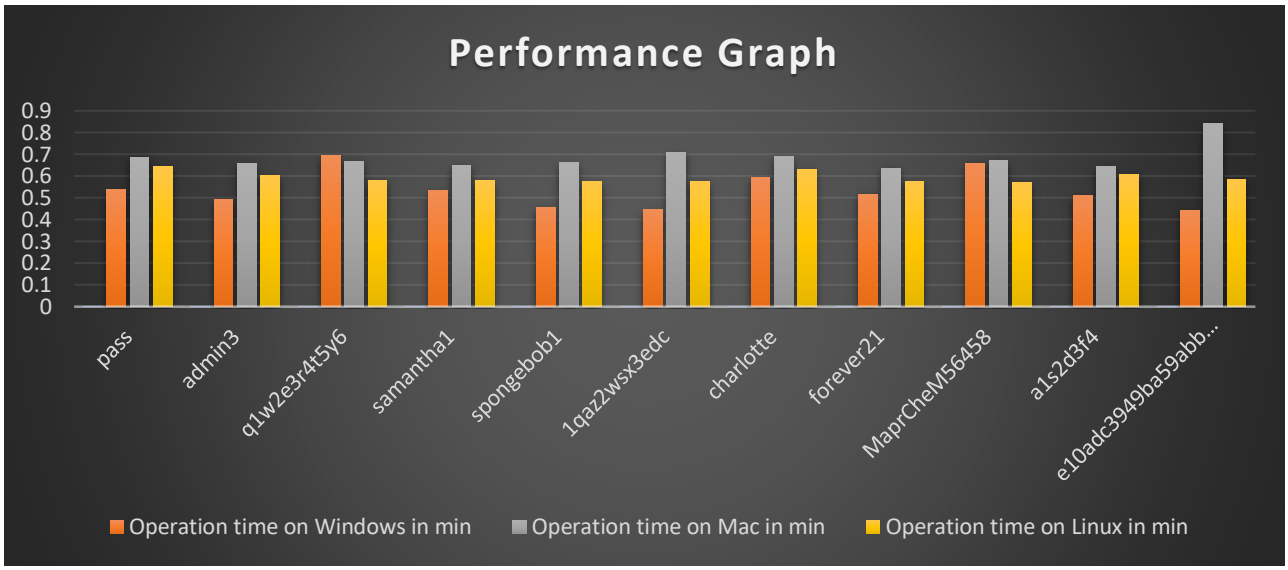

Figure 13: Overall performance graph

SECTION IV

7. Conclusion and Discussion

The main objective of this research was to determine if the security of the system can be enhanced by combining the security methods such as, steganography, encryption and splitting the confidential data which is password in our case. Also, by doing this, we wanted to test whether storing the confidential data in decentralized fashion helps to improve security. Referring to the results that we obtained by implementing the proposed solution on various platforms, we conclude that the solution is highly adaptable as Python programming is cross-platform. In addition, implementing multi-layered encryption with the use of stronger encryption methods like AES-256 boosts the difficulty level for the hackers to penetrate through the system. Varied operating systems do not directly affect the performance of our solution method which means to implement this solution and perform the operation on high-resolution images the minimum configuration required for the system is 8GB RAM with a faster processor like i5 or above. However, even with lower configuration, the performance of the system was good on the Linux operating system as compared to the Mac operating system.

Hence, machines with higher configurations are highly recommended to improve the performance of the proposed solution.

For future enhancement, we would need to optimize the code and pick random passwords from the password list and store the pieces randomly in the images to bolster the level of the security in the system and make penetrating through our system a strenuous task for attackers. Also, work on the latency as code optimization would improve the performance and condense the latency rate.

8. References

[1] V. Venukumar and V. Pathari, "Multi-factor authentication using threshold cryptography," in *2016 International Conference on Advances in Computing, Communications, and Informatics (ICACCI)*, Jaipur, India, Sept 21-24, 2016.

[2] M. M. Kassim and A. S. , "ProcurePass: A user authentication protocol to resist password stealing and password reuse attack," in *2013 International Symposium on Computational and Business Intelligence*, New Delhi, India, 2013.

[3] D. Mirante and J. Cappos, "Understanding password database compromises," 2013. [Online]. Available: https://pdfs.semanticscholar.org/0337/d1329f79b8b736295d9c056b012faf7343c4.pdf. [Accessed 3 Dec 2019].

[4] E. S. I. Harba, "Advanced password authentication protection by hybrid cryptography & audio steganography," *Iraqi Journal of Science,* vol. 59, no. 1C, pp. 600-606, 2018.

[5] J. H. Kennedy, M. T. A. Khan, M. J. Ahmed and M. Rasool, "Image steganography based on AES algorithm with huffman coding," *International Journal of Engineering Science and Computing, April 2017,* vol. 7, no. 4, pp. 6352-6355, 2017.

[6] S. K. Sonker, S. Kumar, A. Kumar and D. P. Singh, "Image based authentication using steganography technique," *International Journal of Advanced Research in Computer Science,* vol. 4, no. 8, pp. 277-282, May/June 2013.

[7] V. Moraskar, S. Jaikalyani, M. Saiyyed, J. Gurnani and K. Pendke, "Cued Click Point technique for graphical password authentication," *International Journal of Computer Science and Mobile Computing,* vol. 3, no. 1, pp. 166-172, January 2014.

[8] S. Wiedenbeck, J. Watersa, J.-C. Birget, A. Brodskiy and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system," *International Journal of Human-Computer Studies,* vol. 63, no. 1-2, pp. 102-127, July 2005.

[9] L. Sharma and A. Gupta, "Image encryption using Huffman Coding for steganography,," *International Journal of Advance research , Ideas and Innovations in Technology,* vol. 2, no. 5, pp. 1-10, 2016.

[10] D. R. I. M. Setiadi, E. H. Rachmawanto2 and C. A. Sari, "Secure image steganography algorithm based on DCT," *Journal of Applied Intelligent System,* vol. 2, no. 1, pp. 1-11, April 2017.

[11] S. F. Mare, M. Vladutiu and L. Prodan, "Secret data communication system using steganography, AES and RSA," in *2011 IEEE 17th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, Timisoara, Romania, October 20-23, 2011.

[12] M. E. Saleh, A. A. Aly and F. A. Omara, "Data security using cryptography and steganography techniques," *(IJACSA) International Journal of Advanced Computer Science and Applications,* vol. 7, no. 6, pp. 390-397, 2016.

[13] P. Mahajan and A. Sachdeva , "A study of encryption algorithms AES, DES and RSA for security," *Global Journal of Computer Science and Technology Network, Web & Security,* vol. 13, no. 15, pp. 15-21, 2013.

[14] P. Patil, P. Narayankar and S. M. Meena, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and Blowfish," in *International Conference on Information Security & Privacy (ICISP2015)*, Nagpur, India, December 2015.

[15] A. Shamir, "How to share a secret," *Communications of the ACM,* vol. 22, no. 11, pp. 612-613, 1979.

[16] RSA, "RSA Distributed Credential Protection," RSA-EMC2, London, 2012.

[17] G. I. Davida, D. J. Linton, C. R. Szelag and D. L. Wells, "Database security," *IEEE Transactions on Software Engineering,* Vols. SE-4, no. 6, pp. 531 - 533, Nov. 1978.

[18] S. Imran and I. Hyder, "Security issues in databases," in *2nd International Conference on Future Information Technology and Management Engineering*, Sanya, China, 13-14 Dec. 2009.

[19] G. Singh and Supriya, "A study of sncryption slgorithms (RSA, DES, 3DES and AES) for information security," *International Journal of Computer Applications,* vol. 67, no. 19, pp. 33-38, April 2013.

[20] G. Singh, A. Singla and K. S. Sandha, "Cryptography algorithm comparison for security enhancement in wireless intrusion detection system," *International Journal of Multidisciplinary Research,* vol. 1, no. 4, pp. 143-151, August 2011.

[21] OWASP, "OWASP SecList Project," 7 November 2018. [Online]. Available: https://www.owasp.org/index.php/OWASP_SecLists_Project. [Accessed 10 June 2019].

[22] csafe, "Stego App DB," 04 April 2019. [Online]. Available: https://data.csafe.iastate.edu/StegoDatabase/. [Accessed 15 July 2019].

[23] B. A. Meier, Python GUI Programming Cookbook - Second Edition, Birmingham, UK: Packt Publishing, May 2017.