ISSN 2278-3091
Volume 9, No.1.2, 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse3591.22020.pdf
https://doi.org/10.30534/ijatcse/2020/3591.22020

# Hybrid Encryption Algorithm towards Secured Instant Messaging Application

**Gemma D. Belga[1], Nelson Maligro[2], Ma. Ian P. De Los Trinos[1]**
[1]Technological University of the Philippines, Philippines, gemma_belga@tup.edu.ph
[2]Philippine Navy, Philippines, nelsonmaligro@gmail.com
[1]Technological University of the Philippines, Philippines,maian_delostrinos@tup.edu.ph

## ABSTRACT

Instant Messenger has been an effective tool for communication. However, the popularity and pervasiveness of such applications come with the constant threat of data interception. This study aims to provide an alternative solution for a secure and reliable communication medium and file exchange by implementing a multi-layer encryption of hybrid Advanced Encryption Standard (AES) 256-bit algorithm and Hidden in Plain Sight (HIPS) image hiding technique. The messaging application was evaluated using ISO 25010. The application was rated 4.00 with a descriptive rating of "Very Good" based on the result of the evaluation conducted. While it is recommended for further enhancement, the SIM proves to be a viable alternative application for instant messaging whose focal strength is in security.

**Key words:** instant messaging, cryptography, mobile security, AES, Hidden in Plain Sight, encryption, steganography.

## 1. INTRODUCTION

Data exchange or data communication is a key process of providing accurate and timely information to senior leaders, top management, and decision-makers, however, recent studies show that Twitter, Skype, LinkedIn, Dropbox, and Zoom has been subjected to information compromise contributed by malicious attacks[1]. The rapid development of communication mediums also increased vulnerability to malicious users. This scenario brings the increase of information transmitted electronically to rely on cryptography and authentication[2]. These forms of communication have also been used in military operations [3], which recognizes the information compromise it poses. In the Philippines, The Philippine Navy (PN), a branch of service in the Armed Forces of the Philippines (AFP), sought to acquire similar tool but is constrained with limited resources and lack of communication infrastructure to securely extend its network to mobile units thereby utilizing the un-secure Internet for its data communication. The PN, with its prevailing need for secure data communication like instant messaging, started to adopt several free and open-source IM but unsatisfied with the minimal encryption it provides. These mediums can be intercepted, decoded, and read in clear text as it travels across the network from sender to receiver[4]. The pervasiveness of these applications makes it vulnerable from various attacks especially sniffing and hijacking[5].The effect of confidential messages being sniffed, leaked, or compromised can be devastating to any individuals, businesses or institutions such as the Armed Forces. Information security can be ensured with the use of encryption[6]. Towards the development of a secured mobile application for the PN, this research designs multi-layer encryption by combining Advanced Encryption Standard (AES) algorithm and Hidden in Plain Sight (HIPS) image hiding technique. This study implements the hybrid of algorithms in an Instant Messaging desktop application called Secured Instant Messaging (SIM) system. The system has the essential features of a typical IM such as one-to-one chat, room chat, and file transfer. It is designed and developed in modules or components and provides the platform that allows other library/API/modules integrated into**.**

## 2. RELATED STUDIES

Padate emphasized the importance of data security and how the Advanced Encryption Standard (AES) improves the security of any data communication application. AES which was originally called Rijndael is an open-standard or open-book specification of an encryption algorithm published by the U.S. National Institute of Standards and Technology (NIST) in 2001. AES was selected among fifteen (15) encryption algorithms that were subjected to rigid test and evaluation. AES was designed to be resistant against all known brute-force attacks. It is a symmetric block cipher that comes in three key lengths: 128, 192, and 256 bit. The basic implementation of AES is simple mathematical and logical table lookup operations. This symmetric key algorithm is fully open to the public for evaluation and scrutiny, and to ensure transparent analysis and validation of the design[7].This was further proved by Mahajan who concluded that the AES algorithm is faster than RSA which consumes the longest encryption time. Further, the decryption of AES is better than other algorithms[8][9].

Recent developments in email communication introduce the adoption of asynchronous or certificate-based encryption into the email. Another study proposes a secure method of e-mail communication using hybrid encryption. It combines hash function, symmetric encryption and asymmetric

encryption[10]. The study is further enhanced by Calinawan in his research "Hybrid Encryption Algorithm Implementation on Electronic Mail Service". The researcher implemented the Rivest–Shamir–Adleman (RSA), certificate-based encryption. The result shows faster execution time, reliability, and improved security[11].

Further studieswere conducted to demonstrate the integration of an open standard encryption algorithm like the AES into a communication application[11] (i.e. Instant Messenger). MohHeng Huong successfully integrates the AES 128-bit encryption into his communication application. The AES 128 encryption provides a more secure communication than other encryption algorithms[12]. The encrypted data is unbreakable until today using this algorithm[4]. The integration of the chosen encryption algorithm into the IM presents a huge challenge. By studying and adopting the previous development of network application incorporating encryption algorithm significantly reduces the amount of time hard-coding these algorithms. Krishna[13] published a method of integrating AES encryption into file transfer application using secure shell protocol. It uses a "Toolkit" to handle the process of encryption and password authentication. Toolkits are a readily available software component that can be integrated into the main application.

Previous studies on cryptography at the Technological University of the Philippines (TUP) offers a viable option for encryption to complement the AES. Agustin[14], with his Hidden in Plain Sight thesis, discusses the method of hiding plain files into an image file. Contents or bytes of the file fill in the white spaces of the image file using the Least Significant Bit (LSB) insertion technique[15]. These blank spaces are almost unnoticeable to the human eye. Such a method of encryption is commonly called steganography[16]. While there are several methods in steganography, in this case, the plain password is translated into a numerical value using shift, modulo and XOR bitwise operations[17]. Steganography technique is further implemented in the thesis of Torres[18], titled Development of FileGuard: Steganography Software. Torres simplified the steganography technique by creating modules or algorithms for extraction, encryption, and compression.

Establishing a secure data transmission requires the implementation of cryptographic in the communication channel. Both the sender and receiver must agree upon a set of security protocols for them to communicate and understand each other[5].

In this study, the AES algorithm serves as securing the messages between clients and the HIPS algorithm implements steganography techniques in the attached files in the secures messaging communication

## 3. METHODS

### 3.1. Conceptual Framework

This study employs the conceptual framework depicted in Fig.1. The Secured Instant Messaging (SIM) system is the messaging application software that provides encrypted chat communication and file transfer for secure and reliable exchange of information[19].

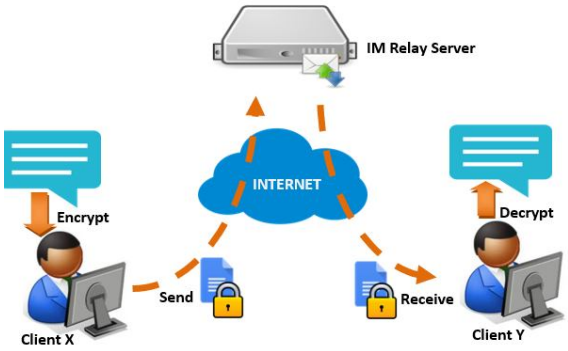


**Figure 1:** Conceptual Framework

The Client X initiates a chat conversation with Client Y using the SIM client software. During the chat conversation, the system automatically encrypts the text message before sending it to the IM Relay Server. The server now searches for the intended recipient from its list of client sockets. When the recipient's socket is available, the server transmits the encrypted text message to Client Y (recipient) through its socket. The server manages all clients' sockets. It handles authentication of all incoming clients before acceptance of the socket. IM Relay Server maintains a minimal database using MS SQL Server for accounts and activity logs. As soon as the text message arrived at Client Y (recipient), the software then decrypts the text using the system-generated key as the default. Unreadable messages indicate unmatched passkey. Both Client X (sender) and Client Y (recipient) must encode manually the correct or agreed passkey to decode their conversation. The manual encoding of a passkey is done to deny the system administrator or any authority from decoding the message during the conduct of any special investigation. System generated keys are stored in the server specifically for this purpose. It is strongly suggested that this manual key should be transmitted through another mode of transmissions like SMS, Email, or voice call.

A similar procedure can be done during sending and receiving files. Client X starts up the File Transfer window and browses for the file. By default, SIM uses Advanced Encryption Standard (AES) 256-bit to encrypt and decrypt text messages and files. The sender can choose to double encrypt the file by hiding it in an image file using steganography or Hidden in Plain Sight (HIPS) algorithm. A checkbox or option button is provided for this feature. By checking both AES and HIPS, the system first encrypts the file using AES and hides the resulting file into an image. The SIM then transmits the image file to the IM relay server. The server searches for the intended recipient from its list of client sockets. When the recipient's socket is available, the server relays the file or message to Client Y (recipient). As soon as the encrypted file or image file arrived, the user can now browse for the file and choose to open it with the same application. The encrypted file is extracted from the image file using the HIPS function

for file extraction. After the successful extraction, the SIM now tries to decrypt the extracted file using the system-generated key. If unsuccessful, it asks for the passkey which must be encoded by the receiving user to convert the encrypted file back to its original form. Manual encoding of passkey happens when the sender chooses to encode a different key other than the default system-generated key[20].

To visualize its deployment from a technical perspective, the network diagram is drawn for this purpose (Fig. 2). the SIM primarily utilizes the Local Area Network (LAN) for inter-office instant messaging and collaboration. Client X (desktop computer) transmits messages to Client Y (laptop computer) through the adjacent relay server within the same LAN.

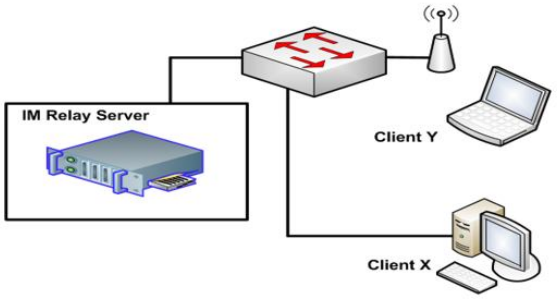


**Figure 2:** Local Network Setup

The software application is also applicable to clients outside the office and anywhere in the world. For a Wide Area Network (WAN) setup reflected in Fig. 3, the Internet is utilized. The IM Relay Server is assigned with a global or public IP address to allow connection from clients outside of its network. This allows interconnectivity among regional offices, company branches, or widely dispersed units through this system.

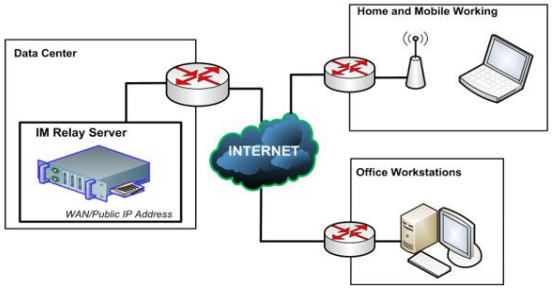


**Figure 3**: WAN Setup

Users who bring their work at home can now securely communicate and send files to the main office through the SIM. Firewalls and Intrusion Prevention System (IPS) provide optional security reinforcement at the network layer between the SIM relay server and SIM client. Servers are ideally housed in a Data Center for large-scale deployment.

### 3.2. File Encryption and Decryption

Encryption and decryption of chat messages and files are done automatically by the system. The SIM adapts AES 256-bit by default as its first-layer encryption[19]. Users may choose not to use the system-generated key and encode a different manual key known to both the sender and recipient. This is done to deny the system administrator or any authority from decoding the files during any special investigation. Fig. 4 depicts the process of encrypting and decrypting files. Users may choose to double encrypt the file using steganography hiding technique[21]. To do this, end-users click the HIPS radio button to hide the data file in an image before the transmission. When the recipient received the encrypted file, the user can now browse for the file and open it with the same SIM application. The system extracts the embedded file using HIPS algorithm and then decode it using the AES decrypt function.

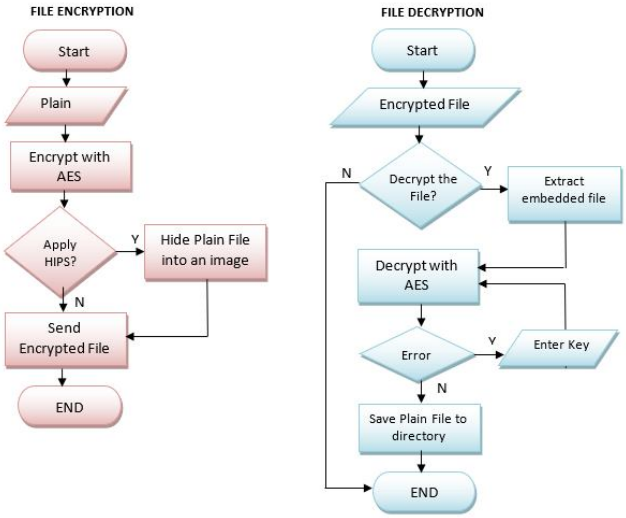


**Figure 4:** Hybrid Encryption Algorithm

### 3.3. Software Development

The development of SIM adopted agile methodologyso that various modules or components including other third-party modules (e.g. DLL, API, and SDK) can be integrated into the system; vis-a-vis one component can be released or launched even without the other unfinished components making it ideal for evolutionary or continuously improved software. The SIM was divided into three (3) components - the Base, File transfer, and Cryptography component. These components served as sprints or iterative work activities needed to complete the entire system.

### 3.4. Testing Procedure

The SIM was subjected to functionality, security, latency, and portability testing to ensure an error-free application across all versions of the MS Windows operating system. The testing was conducted among computer programmers and system administrators to immediately resolve or debug any fault of the system. Test cases were documented and form part as supplementary material or evidence of this research. Table 1 enumerates the sample steps conducted during the testing of SIM through various scenarios.

**Table 1:** Sample Testing Scenario

| Scenario | Steps to be Undertaken |
|---|---|
| Client Login | 1. Attempted to login with erroneous entries<br>2. Login with correct entries<br>3. Verified the password was encrypted using the Wireshark tool of KALI Linux |
| Send and Receive Chat Message | 1. Sent a long and random chat message to determine the possibility of a crash.<br>2. Sent readable chat message and verified the accuracy of the message as it reached the recipient.<br>3. Measured the time it took for the message to reach the recipient<br>4. Verified the text was encrypted during transmission using Wireshark tool of KALI Linux |
| Send and Receive File | 1. Sent a file and measured the time it took for the file to reach the recipient.<br>2. Sent large files and checked the possibility of crashes.<br>3. Verified the HIPS hiding technique by locating the image file containing the embedded data file.<br>4. Verified AES encryption is applied by opening the encrypted file if it is unreadable. |

### 3.5. Software Acceptance

Software acceptance was conducted to ensure conformity to the required specification and that the system exhibits characteristics that meet the criteria of quality and secure application software. This study adopted the ISO 25010 standard for evaluating the software quality and validates the security of the SIM. The following are the essential criteria of this standard which were used as the evaluation metrics for this study: Functional Suitability, Performance Efficiency, Usability, Reliability, Security, Maintainability, Portability, and Compatibility. Five (5) IT experts, Five (5) cybersecurity professionals and Five (5) end-users were invited as respondents/evaluators for this research.The evaluation metrics and the Likert Scale were presented and discussed to the evaluators, (5 as Excellent through 1 as Poor).

## 4. RESULTS

### 4.1. Application Development

Fig. 5 shows the final components of the system.

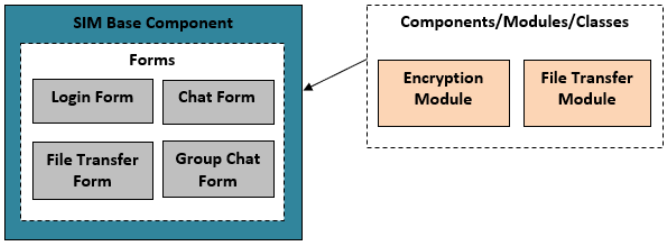


**Figure 5**: SIM Components

The SIM is composed of three (3) modules or components: Base, Cryptography, and File Transfer.

The base component is the most important component of the SIM. It provides the platform in which other components are integrated. It comprised the forms, references, properties, libraries, and the main program needed to run the system. The following are the forms created for the base component: Login Form that provides the interface needed to authenticate the user to the SIM server. The password entered by the user is automatically encrypted upon clicking the login button; the Main Window Form that displays after a successful login. Groups and users are populated in the tree view. The green user icon indicates an online user while the gray icon indicates an offline user; a Chat Form (see Fig. 6) that allows the user to interact with other peers in the SIM through chat or text-based conversation. The chat communication is encrypted by default using AES 256-bit encryption with passkey generated by the system. The user has the option to manually assign a key for a more private conversation. All system-generated keys are changed weekly and stored on the server. This allows the organization to decode the previous conversation during an internal investigation as the need arises. Having keys manually entered denies the possibility of having someone to decode the conversation other than the two (2) communicating peers; and File Transfer Form that allows transmission of documents, images, audio, video, applications, and other files from one user to another. When applying encryption to the file, the encryption module/class is called. The user has the option to use the HIPS for multi-layered encryption of the file.

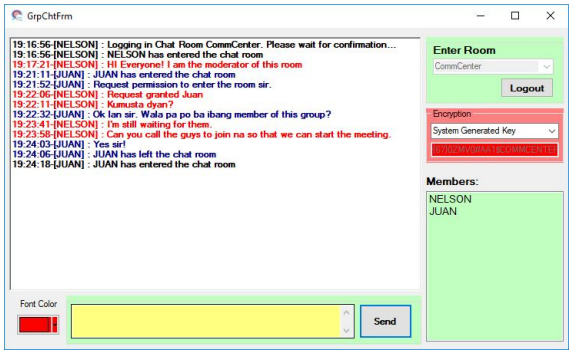


**Figure 6:** SIM Chat Window

The Cryptography Component/ Module is the component that provides encryption and decryption through AES and HIPS of the SIM. This module can be deployed separately through DLL and can be integrated into any applications. The following are some of the enhancement of the HIPS: Auto-resize of the image file – the original HIPS requires the user to import an image file that must be larger than the embedded file or else an error will return. This method was enhanced with a function that auto-resize an image file to a length twice the size of the embedded file; Direct image file insertion –the original HIPS loads the image file into the Image Control of the Form before hiding the embedded file and then saving the image from the control into another filename. This technique is a slow process; thus, it is replaced with a function that allows direct insertion of the data byte into an object that encapsulates the image file; and Simplified Formula – insertion of the data byte is done by dividing the image file into several bytes and recursively replacing its 4-bits LSB with every 4-bit data byte of the embedded file.

The File Transfer Component/ Module is the component that allows file transmission from one user to another. File

transmission is done by establishing a TCP socket connection from an online user to the server and then the data byte is transmitted through this socket. A detailed explanation of socket communication is presented in the previous chapters. The Base component calls file transfer module/class when the File Transfer form is opened. Upon clicking the send button, the SIM calls the cryptography module and encrypts the file; the encrypted file is then divided into several bytes for transmission. These bytes are sent into streams so that when it reaches the server it is re-assembled back to its original form. The SIM is further improved to satisfy some of the evaluators' requirements. These added competitive features are: sending of voice messages, option for self-delete or self-destruct messages, Users are organized into groups or departments, Multi-colored font for room chat, Server logs for auditing, System-generated key is changed weekly, Encrypted username and password on the database, and Users can decrypt the file at the time of their choosing.

### 4.2. Security Testing

There were four (4) instances conducted to test the resulting data or message as it traverses across the network, as reflected in Table 2. The penetration testing software sniffed the transmission from clientX (source) to clientY (destination) using ARP Spoofing technique. Sniffing can also be done using a port-mirrored network switch. During login, the resulting message appears to be scrambled characters when sniffed. All penetration testing software detected the resulting message as MD5 hash values. Another instance is when clientX sends a chat message to clientY, the message appears to be scrambled characters in the chat window of clientY (the key is changed to prevent decrypting the message). All penetration testing software detected the resulting message as AES-256-bit encryption. During file transmission, clientX sends a plain document to clientY and browsed a JPEG file to hide the document. The jpeg image containing the encrypted file was received by clientY. All penetration testing software detected the data as an image file. When clientY manually decrypts the image file, it extracted sequentially the AES file and the plain document (Docx) file.

**Table 2:** Security Tests Sample

| Instances | Original Message (Source) | Result Message (Destination) | Wireshark | Cain&Abel | Etterca p (Kali) |
|---|---|---|---|---|---|
| User Login | User: user1 Pass:USER @123 | 24C9E15E52A FC47C225B75 7E7BEE1F9D<br><br>E2B31C4CF92 DD40E079B9 B8BA414F9B D | Encrypted with MD5 Hash | Encrypted with MD5 Hash | Encrypt ed with MD5 Hash |
| Send Chat Message | Hello I am user1 | cMYEhI3nug UR4+f97OL1+ CjAFe8M7/1v PKrKvBiDvR0 = | Encrypted with AES-256 bit | Encrypted with AES-256 bit | Encrypt ed with AES-25 6 bit |
| Transmit Docume nt (AES+H IPS) | Smarthouse. docx | Smarthouse.jp g | Image w/ hidden file | Image w/ hidden file | Image w/ hidden file |
| Decrypt Image File | Smarthouse.j pg | Smarthouse.ae s marthouse.docx | Encrypted with AES-256 bit | Encrypted with AES-256 bit | Encrypt ed with AES-25 6 bit |

### 4.3. Latency Testing

Latency Test was conducted to determine the time it takes for the data to arrive at the destination when applied with dual-layer encryption. On the first instance, the encryption key was removed from the Chat feature and measured the timeframe of conversation. The transmitted message appears on the recipient window instantly with a microscopic delay of 0.2 second. This unnoticeable delay was revealed through the use of Desktop Screen Recording software. Similar step is done during the second transmission of text message but with AES encryption applied. This yields with insignificant delay of 0.5 second. However, the delay becomes observable during the file or document transmission. Following similar procedure with chat messaging, yields a truthful result of delayed transmission when AES encryption is applied to the file. The time is increased with 6 seconds when AES is implemented and plus 3 seconds when HIPS is added on a 2mb file. Similarly, on a 10mb file the delay is increased with 8 seconds when AES is implemented and plus 5 seconds when HIPS is added. This shows that the larger the file the longer it takes to encrypt; and this delay consequently adds to the time of transmission as reflected in Table 3.

**Table 3:** Latency Tests Sample

| Instances | Plain (No Encryption) | Encrypted withAES | Encrypted withHIPS |
|---|---|---|---|
| Send Chat Messagewith 50 characters | 0.2 sec | 0.5 sec | Not Applicable |
| Transmit Document with 2mb file size | 8 sec | 14 sec | 17 sec |
| Transmit Document with 10mb file size | 16 sec | 24 sec | 29 sec |

### 4.4. Software Acceptance

Presented in Table 4, the SIM got an average mean of 3.87 in terms of Functional Suitability with a "Very Good" descriptive rating. The system provides the needed functionality for data communication and office collaboration. It was rated high in terms of Security and garnered an average rating of 4.20. The system was tested and evaluated using the Kali Linux – a popular tool for conducting a vulnerability assessment and penetration testing. The validated multi-layer encryption of the SIM transcends other IMs in terms of integrating cryptography. This encryption provided the evaluator an assurance of its security. In terms of

performance efficiency, the SIM got an average mean of 4.0 with a "Very Good" descriptive rating. The system runs in optimum performance under varying conditions. It is also stable and resilient when tested in an undesirable condition. The evaluators compared the SIM with other popular Instant Messenger and found it user-friendly and analogous in terms of the user interface (UI). The SIM is modular and agile in any environment. These characteristics gave the SIM an average mean of 3.97 in terms of maintainability with a "Very Good" descriptive rating. It scored low in terms of portability since it can only be installed on a Microsoft Windows operating system. In the compatibility criteria, the system got an average mean of 4.10. It performs efficiently while sharing a similar .NET library with other applications. The SIM may not have achieved an excellent rating due to its limited features as compared to other popular IM; however, continuous improvement of the system and integration of additional modules will enable it to compete in the IM arena.

**Table 4:** Software Acceptance Summary

| ISO 25010 Criteria/Characteristics | Average Mean | Descriptive Ratings |
|---|---|---|
| Functional Suitability | 3.87 | Very Good |
| Security | 4.20 | Very Good |
| Performance Efficiency | 4.00 | Very Good |
| Usability | 3.92 | Very Good |
| Reliability | 3.90 | Very Good |
| Maintainability | 3.97 | Very Good |
| Portability | 4.04 | Very Good |
| Compatibility | 4.10 | Very Good |
| Total | 4.00 | Very Good |

## 5. CONCLUSION

The Secured IM (SIM) system is an instant messaging application that allows data exchange and office collaboration such as one-to-one chat, group chat, file transfer, and others. The completion of the three (3) modules/components – Base, Encryption, and File Transfer – allowed the whole system to deliver the desired functionalities as enumerated in the objectives and scope of this study. The SIM was able to provide the basic features found in any typical IM today such as chat and file transfer. Its prowess was seen in the implementation of a multi-layered encryption mechanism - the integration of AES 256-bit encryption and HIPS hiding algorithm. The conduct of alpha testing shows that the SIM functionally executes with minimal errors. These errors were later resolved for acceptance. It runs on any versions of the Microsoft Windows operating system except for Windows XP. The SIM was tested across three (3) different penetration testing software. The consistent result of encrypted data was evident during the conduct of security testing. The result of the evaluation revealed that the SIM has little or no advantages over other typical IM in terms of functionalities. It scored low in portability since the application can only be installed in Microsoft Windows platform and not on an android or IOS mobile devices. However, the system proved advantageous in terms of security as it garnered an average mean of 4.20. The validity of encryption assured the evaluator of its security. From the three (3) groups of respondents or evaluators, IT experts gave 3.96 average mean, Cybersecurity experts gave 4.01, and end-users gave 4.03. The end users examined the system from a usability perspective and found little or no difference with that of a typical IM, while the cyber security experts understand very well the importance of security in an instant messaging application.

## 6. FUTURE STUDIES

Towards the development of a secured instant messaging using a hybrid algorithm, the future direction of this study aims to develop modules for voice and video communication must be done. Some open-source software libraries, API, and SDK are already available to enable this feature. These libraries can be integrated into the SIM; improvement of the HIPS hiding technique should be considered. Since the programming language that was used to develop the HIPS is already obsolete, there is a need to conduct further study to learn new techniques in steganography. Tools for steganography nowadays can hide data in images, sound clips, videos, and office documents like MS Word, MS Excel, and MS Powerpoint; further research to develop modules for Short Messaging System or SMS may be conducted. This will allow messaging even if the user is offline and the only available mode of communication is the Global System for Mobile communication (GSM) commonly known as cellphone signal, and to develop modules for Voice Over Internet Protocol or VOIP may also be considered. This will allow voice communication on any VOIP-enabled phone.

## ACKNOWLEDGMENT

Credit is given to Engr. Mardonio Agustin who collaborated and shared his time and expertise for the development of the system; and to the IT experts who participated as respondents of this study. This study acknowledges Dr. Mideth B. Abisado for sharing her ideas and expertise. This study acknowledges the support given by the College of Industrial Technology through its Learning Resource Center Research Laboratoryand the University Research and Development Services.

## REFERENCES

1. E. B. Villanueva, R. P. Medina, and B. D. Gerardo. An enhanced RC5 (ERC5) algorithm based on simple random number key expansion technique.ISCAIE 2018 - 2018 IEEE Symp. Comput. Appl. Ind. Electron., vol. 5, pp. 134–138, 2018. https://doi.org/10.1109/ISCAIE.2018.8405458
2. M. V. C. Suana. Enhancement of Advanced Encryption Standard (AES) Cryptographic Strength via Generation of Cipher Key-Dependent S-Box.Int. J. Res. Appl. Sci. Eng. Technol., vol. 6, no. 4, pp. 1420–1428, Apr. 2018. https://doi.org/10.22214/ijraset.2018.4239
3. M. L. Cummings. The need for command and control instant message adaptive interfaces: Lessons learned from Tactical Tomahawk human-in-the-loop simulations.Cyberpsychology Behav., vol. 7, no. 6, pp. 653–661, Dec. 2004.

4. M. Heng Huong, Implementation of (AES) Advanced Encryption Standard algorithm in communication application. Retrieved January 2019 fromhttp://umpir.ump.edu.my/id/eprint/12047/1/MOH%20HENG%20HUONG.PDF.2014.
5. I. Del Pozo and M. Iturralde. CI: A new encryption mechanism for instant messaging in mobile devices. in Procedia Computer Science, 2015, vol. 63, pp. 533–538. https://doi.org/10.1016/j.procs.2015.08.381
6. C. Liu, L. Chen, and Q. Yao.Instant Messaging Encryption System Based on Secure Decryption Technology..Telecommun. Radio Eng., vol. 78, no. 20, pp. 1847–1856, 2019.
7. W. : Www, R. Padate, and A. Patel. Encryption and Decryption of Text using AES Algorithm,.2015.
8. M. Prerna, A. Sachdeva, and P. Mahajan,. A Study of Encryption Algorithms AES, DES and RSA for Security. Int. Res. J. Publ. Glob. Journals Inc, vol. 13, 2013.
9. P. Guo and W. Xu. Implementation of RSA algorithm based on P system,. J. Comput. Theor. Nanosci., vol. 14, no. 9, pp. 4227–4235, Sep. 2017. https://doi.org/10.1166/jctn.2017.6723
10. T. Mantoro and A. Zakariya. Securing E-mail Communication Using Hybrid Cryptosystem on Android-based Mobile Devices. Indones. J. Electr. Eng., vol. 10, no. 4, Jul. 2012.
11. C. C. Calinawan. Hybrid Encryption Algorithm Implementation on Electronic Mail Service [Online]. Available:https://ejournals.ph/article.php?id=10458. [Accessed: 23-Dec-2019].
12. H. V. Gamido, A. M. Sison, and R. P. Medina. Implementation of modified aes as image encryption schem. Indones. J. Electr. Eng. Informatics, vol. 6, no. 3, p. 301~308, Sep. 2018.
13. M. Krishna, P. Jamwal, K. S. R. Chaitanya, and B. V. Kumar. Secure File Multi Transfer Protocol Design. J. Softw. Eng. Appl., vol. 4, pp. 311–315, 2011. https://doi.org/10.4236/jsea.2011.45034
14. M. . Agustin. Development of Hidden in Plain Sight (HIPS) Encryption. Technological University of the Philippines, 2012.
15. M. S. Subhedar and V. H. Mankar. Current status and key issues in image steganography: A survey.Comput. Sci. Rev., vol. 13–14, no. C, pp. 95–113, Nov. 2014.
16. M. R. D. Molato, B. D. Gerardo, and R. P. Medina,. Secured data hiding and sharing using improved lsb-based image steganography technique. in ACM International Conference Proceeding Series, 2018, pp. 238–243.
17. M. R. D. Molato and B. D. Gerardo. Cover image selection technique for secured LSB-based image steganography. in ACM International Conference Proceeding Series, pp. 1–6. .2018, https://doi.org/10.1145/3302425.3302456
18. J. P. Torres. Development of FileGuard: Steganography Software. Technological University of the Philippines, 2012.
19. G. A. Ali, Roadmap for incorporation Vocational Certification into University (IJATCSE).vol. 8, no. 3, pp. 10–13, 2019. https://doi.org/10.30534/ijatcse/2019/12832019
20. M. Grace, Z. Fernando, I. Secured Private Key Handling using Transposition Cipher Technique.International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE).vol. 9, no. 1, pp. 85–89, 2020. https://doi.org/10.30534/ijatcse/2020/1691.12020
21. M. O. Espina, A. C. Fajardo, B. D. Gerardo, and R. P. Medina. Multiple level information security using image steganography and authentication. International Journal of Advanced Trends in Computer Science and Engineering(IJATCSE) vol. 8, no. 6, pp. 3297–3303, 2019. https://doi.org/10.30534/ijatcse/2019/100862019