Secure Multi-Agent Integrated Password Management (MIPM) Application

N. Awang[1], N. H. A. Zukri[2], N. A. M. Rashid[2], Z. A. Zulkifli[2], N. A. N. Adnan[2]
[1] *Faculty of Computer & Mathematical Sciences, UiTM Shah Alam,*
[2] *Faculty of Computer & Mathematical Sciences, UiTM (Melaka) Kampus Jasin*
shaini@tmsk.uitm.edu.my

***Abstract*— Rapid development of the Internet and increasing number of available Web applications require users to memorize the passwords for authentication. Password management system is a solution to store login information such as password and help users to log in automatically. In preventing from password leaking, this project was integrated with the multi-agents using Java Agent Development Environment (JADE). The purpose of the embedded agents is act as the third party software to ease the encryption process. The application agent sends data to Crypto Agent using Agent Communication Language (ACL) message, a method of communication between agents. Upon receiving, crypto agent encrypts the data using AES 128-bit encryption algorithm. AES will facilitate the process of the encryption and decryption, and withstand the application from the brute force attack. As whole, MIPM is a development on android application that provide secure platform to store their login account information such as their username, emails and passwords. While, the implementation of the multi-agent communication in this project can be expanded to be commercialize for any web applications development.**

***Index Terms*— AES encryption; multi-agent; Java Agent Development Environment; password management.**

I. Introduction

Password management system is a system that preserves user's information and privacy by keeping the list of password in a database. The password would be retrieved from the database whenever the user needs them. Studies have proven that many users tend to forget password [11] due to different passwords are required for each account. Corresponding to the evolution of software technology, password management system took one step further by implementing the multi-agent architecture as part of the system. The use of software agent has become one of essential component in most of the application [2]. A multi-agent system is a loosely coupled software agents that interact to solve problems. By considering the risks of vulnerabilities and threats towards the user data, this application implement AES algorithm to increase the confidentiality, integrity and availability of user data.

This application provides users a platform place to store their password securely. Username and password that save in the database automatically encrypted before store it in the database. When users submit the request to retrieve the password and username, the system will decrypt the original data. Hence, it is difficult for attackers to retrieve or steal the password. Besides, the auto login was developed to help users log in to the web application without memorizing the username and password. This paper is organized in five sections. The second section provides related work which includes issues of cryptography and multi agent system. In the third section, architecture of the system, particularly encryption and decryption process, and system testing is discussed. Section four describes the findings from user acceptance test. The last section focuses on conclusion and direction for future works.

II. Related work

Proliferation usage of the password has led the developers to introduce the password management system. Password management system is protecting the confidentiality of the user's privacy by storing the list of the encrypted passwords in the database. As long as users connected with the internet, the passwords could be retrieved. For authentication and authorization purposes, users need to remember the master password of the application. Password management system acts like a digital safe, which securely stores user's usernames, password and other sensitive information. This securely system is actually encrypt the passwords by using the cryptography techniques.

A. Cryptography

According to Agrawal and Mishra (2012), cryptography converts the original message into non readable format and sends the message over an insecure channel. There are two types of cryptographic systems that have been developed for the security purpose, which are symmetric (secret key) and asymmetric (public key) cryptosystems [4].

In symmetric cryptography, for both encryption and decryption methods only private key is used [4]. The owner of private key cryptography system will hold only one key that is the secret key; it may also be swapped between the sender or the owner and the receiver of the message [6]. Saranya, et al. (2014) specifically categorized the likewise private key situation above into five components that is plaintext, cipher text, encryption algorithm also with decryption algorithm, and last but not least is the private key. The mechanism for symmetric key cryptography may seem clear and simpler as analogy of using the same key to lock and unlock the door. However, this cryptography in this digital environment requires mechanism to distribute the same private key to both sides and the mechanism should be sophisticated enough to support the operation [7]. Pahal and

Kumar (2013) state an example of the symmetric cryptography is Advanced Encryption Standard (AES).

AES uses 128-bit data block and may use three different key sizes which are 128, 196, and 256 bits. The 128-bit data block is divided into 16 bytes and are mapped into a 4 x 4 array called State. Implementation of the AES based on one complete round, the data is iteratively looped until the encryption or decryption is completed. Encryption ratio for AES is high and there is no tenability, in terms of speed, AES have a fast speed in encrypting and decrypting process [8]. U.S government sees the high performance of AES, thus adopted as an encryption standard and approved by the National Security Agency (NSA) for encryption of confidential information.

In asymmetric cryptography, two keys are used which are public and private [4]. Rivest-Shamir-Adleman (RSA) is an asymmetric cryptographic algorithm that is most commonly used compared to other asymmetric cryptographic algorithms. Agrawal and Mishra (2012) stated that they are two types of keys that are used for encryption and decryption which are public and private key. Zhou and Tang (2011) explained that the implementation of RSA cryptosystem is a rather complex process. It involves large integer modular arithmetic, generation of prime numbers and other mathematical calculations. The most important factor is RSA cryptosystem is the efficiency in generating large prime numbers. The key length in RSA cryptography algorithm is generally high, thus difficult for an attacker to break the code. However, Agrawal and Mishra (2012) concluded that asymmetric key algorithms such as RSA runs slower compared to symmetric key algorithms. Furthermore, the security aspect of symmetric key encryption is much better than asymmetric key encryption. Our approach is aligned with Agrawal and Mishra (2010), we design the multi-agent system which encrypt the data using symmetric key algorithms.

B. *Multi Agent System*

Agents are computer systems with two important capabilities. First, autonomous action which deciding what need to do in order to satisfy design objectives. Second, communicating with other agents by engaging in the analogues of social activity such as cooperation, coordination, and negotiation [10].

According to Sulaiman (2010), each agent has its own specialty in order to complete the tasks. There are several characteristics of multi agent system. First, each agent controls its own task there is no global system control. Second, data is distributed whereby each agent will need to communicate among each other in order to obtain resources. Third, each processes of agent are independent of each other. Multi agent system is suitable to handle complex problems by using modularity.

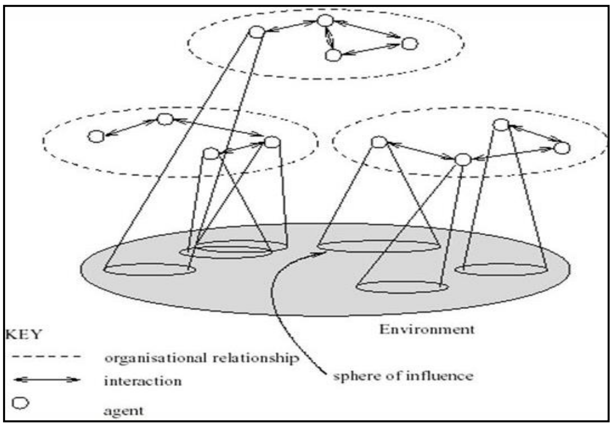


Figure 1 Typical structure of multi agent system
(Source: Wooldridge, 2002)

Based on Figure 1, the system contains a number of agents that interact with each other via communication. Different agents have different spheres of influence, which at least able to influence different part of environment. Spheres of influence may correspond in some cases and may give rise to dependency relationships between the agents. Agents typically linked by other relationships [10]. By using agents, there are several benefits can be gained. For example, reduced communication costs, limited local resources, easier coordination, asynchronous computing, and flexible distributed computing architecture.

III. METHODOLOGY

A. *System Architecture*

This project proposed two agents that involved directly in the security of user data which are application agent and crypto agent. Application agent acts as third party software in between the password management application and the crypto agent throughout the process of encryption and decryption. The communication between these two agents may limit the usage of processing power to process the encryption and decryption. Therefore, the aim of this project is to design and develop multi-agent integrated password management application using encryption technique by implementing application agent and crypto agent. Two main components of the agents which are the Graphical User Interface (GUI) and the Application Agent are resides in the JADE platform. The GUI acts as an interface for the database, user and the application agent. The GUI reads and writes from the database and displays the data on the application screen. Additionally, the GUI retrieves the application agent whenever an encryption process is needed.

When called by the GUI, application agent receives related data. Application agent main function is to act as an interface between the android platform and the Java Virtual Machine (JVM). The agent sends data to Crypto Agent in JVM using Agent Communication Language (ACL) message which is a method of communication between agents. Once received the data, Crypto agent will either encrypt or decrypt data and send it back to the application agent.

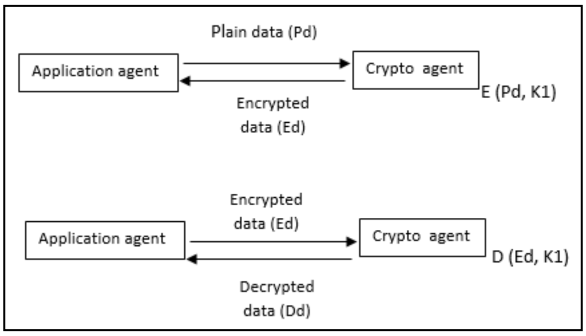


Figure 2 Encryption and Decryption Process

Figure 2 illustrates the encryption and decryption process involves in this project. As mention earlier, application agent responsible to fetch data to the crypto agent that reside in the server to perform the encryption and decryption process. This cryptography process used AES algorithm with the key size of 128Kbytes. Crypto agent generates new key as it received new plain data from the application agent. Crypto agent runs the encryption function by encrypt the data received with the generated key. The same generated key is used for the decryption process. Generated keys kept by the crypto agent in a secure storage.

Refer to Figure 3 illustrate the system architecture of MIPM application using encryption technique. In the system, there were 3 environments involved which are the Android Platform, the JVM and the Online Database itself. From the system architecture of MIPM, if the system chooses encryption process, it calls the application agent then passes the plain text to the crypto agent. Once the encryption process is complete, the ciphertext is then passed to the application agent and write to the database. The decryption processes in MIPM start when the application agent read the data from the database. The ciphertext sent to the crypto agent for decryption process. Once the decryption process is completed it passes back the plaintext to application agent to display the data on the application

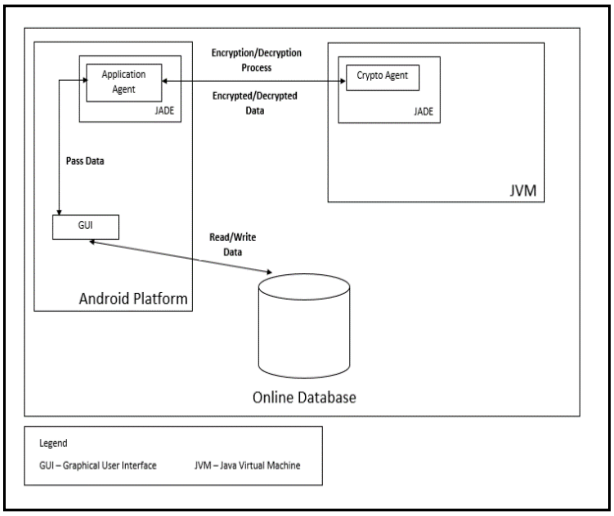
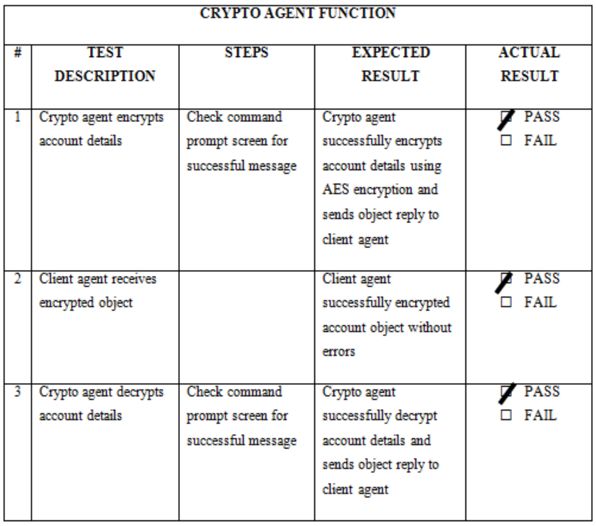


Figure 3: MIPM System Architecture

.

B. System Testing

Functionality testing is one of the important tests to check whether the application is work properly. Functionality test is conducted for user to evaluate the functionality of the MIPM application. Figure 4 shows the module of functionality testing which test on the crypto agent function. This module tests the encryption and decryption functions that crypto agent performs in this project. The result from this test shows that there is no error while using this module.



CRYPTO AGENT FUNCTION				
#	TEST DESCRIPTION	STEPS	EXPECTED RESULT	ACTUAL RESULT
1	Crypto agent encrypts account details	Check command prompt screen for successful message	Crypto agent successfully encrypts account details using AES encryption and sends object reply to client agent	☑ PASS □ FAIL
2	Client agent receives encrypted object		Client agent successfully encrypted account object without errors	☑ PASS □ FAIL
3	Crypto agent decrypts account details	Check command prompt screen for successful message	Crypto agent successfully decrypt account details and sends object reply to client agent	☑ PASS □ FAIL

Figure 4 : Functionality Test on Crypto Agent Module

IV. Finding

The result from this research proves that the project has successfully stored and encrypted sensitive user data. The project achieves its main purpose of providing a platform for users to store the social media login information. The project increases security of storing users' information by using a strong encryption algorithm and utilizing JADE technology.

A. Auto-Login and Auto-Fill Email Account

Figure 5 shows the interface of the MIPM application. This application provides user a place to store their password securely and list down their password and respective account of the password. Besides, this application provide user an auto login approach only for Yahoo Mail account and auto fill approach only for Gmail account. Users can use this application to list down their password and respective site or application of the password. Once users save the username and password, the application automatically encrypt the username and password before store it into database.

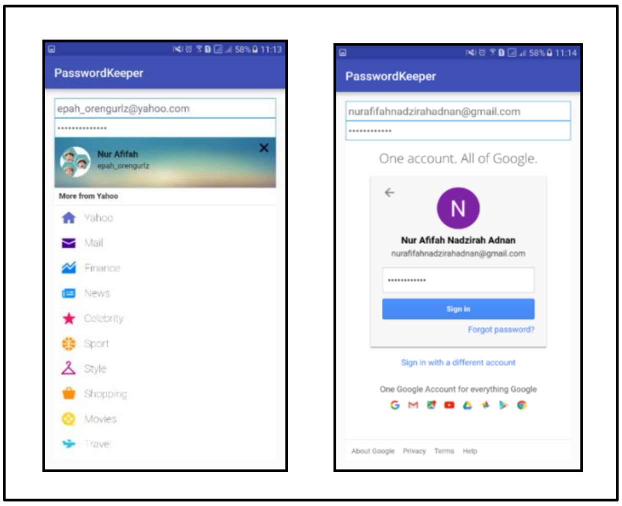


Figure 5 : Auto-Login and Auto-Fill Email Account

B. *Result of Crypto Agent and Application Agent*

In order for the user to retrieve the password and username, MIPM automatically decrypt it back to the plaintext. Hence, it is hard for third party to retrieve or steal the password. FIGURE 6 shows the crypto agent and application agent that reside in the server once they are launched. Auto login approach eases the users because it is no need for the user to memorize the username and password anymore.

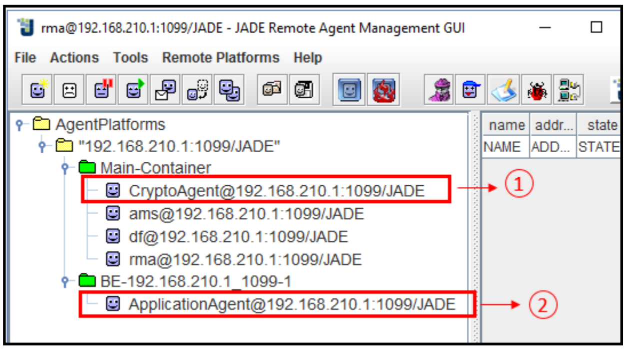

rma@192.168.210.1:1099/JADE - JADE Remote Agent Management GUI
File
Actions
Tools
Remote Platforms
Help
AgentPlatforms
"192.168.210.1:1099/JADE"
Main-Container
CryptoAgent@192.168.210.1:1099/JADE
ams@192.168.210.1:1099/JADE
df@192.168.210.1:1099/JADE
rma@192.168.210.1:1099/JADE
BE-192.168.210.1_1099-1
ApplicationAgent@192.168.210.1:1099/JADE
name
addr...
state
NAME
ADD...
STATE
1
2

Figure 6: Crypto Agent and Application Agent

C. *User Acceptance Test*

User acceptance test is conducted to make sure MIPM can handle the task in real-world scenarios, according to specifications. A total of 32 android users were provided with the application to further be installed and tested. After using the MIPM, respondents will evaluate the system by tick the rating scale for each section in the User Acceptance Test (UAT) Test Script. The rating scale is divided into five which are strongly disagree, disagree, neutral, strongly agree and agree. While, two sections evaluated are user interface to identify the presence of defect and usability to understand the users experience in using the system.

Table 1
User Interface Result

No.	Module	Rating Scale		
		Neutral	Agree	Strongly Agree
	Section A: User Interface			
1	The application interface design was consistent throughout the system	0%	40%	60%
2	The application design was simple and easy to understand	5%	50%	45%
3	The application interface was interactive	5%	60%	35%
4	The navigation between pages was easy and clean	0%	30%	70%
5	The application user interface was well constructed	5%	50%	45%

The resulting user interface and questions are defined and illustrated in Table 1. User Interface involves checking the screens with the controls like menus, buttons, icons, and the design structure. From the responses, none of the respondents strongly disagree and disagree with the interface of the application. 40% agree and 60% strongly agree that all images, colors fonts and design of the system were consistent throughout the system. The user interface was designed as simple as possible yet easy to understand by all respondents. 60% from the respondents agree that the interface was interactive, thus can lure many users to use the application. The navigation between pages was easy and clean to minimize the work of the users. In overall, the total of 95% from the respondents agree and strongly agree that the interface was well constructed.

Table 2
Usability Result

No.	Module	Rating Scale		
		Neutral	Agree	Strongly Agree
	Section B: Usability			
1	The process of adding new account ran smoothly	0%	50%	50%
2	The process of viewing added account ran smoothly	0%	40%	60%
3	The process of browsing website in the application ran smoothly	5%	55%	40%
4	All buttons in the application was function well	0%	40%	60%
5	All modules in the application was function correctly	0%	30%	70%

The modules developed in the application were functioned correctly as 30% from the respondents agree and 70% strongly agree. Table 2 shows respondents can add new account and view added account. Additionally, respondents can surf any website that being added previously within the application itself.

V. CONCLUSION

Multi-Agent Integrated Password Management Application is providing users a secure way to store their social media account information by using encryption method. This project developed on both android platform and Java Agent Development Environment (JADE).

The main application pre-installed in the android platform act as the main interface between the user and the application itself. The android application contains an application agent as a way to communicate with the crypto agent which is located on the Java Virtual Machine (JVM). The crypto agent performs the encryption process using AES 128-bit encryption algorithm.

The result of the tests conducted proves that the project has successfully stored and encrypted confidential data. Good responses received from the respondents indicate all functions were performed correctly and the user interface is attracted enough. The project achieves its main purpose of developing multi-agent system for AES algorithm implementation and providing a platform for users to store the social media login information. The project increases security of storing users' information by using a strong encryption algorithm integrated with multi-agent technology.

ACKNOWLEDGMENT

The authors would like to thank the Universiti Teknologi Mara (UiTM) for sponsoring this research under the iRAGS Grant (600-RMI/DANA 5/3/IRAGS (1/2015)) of Research Management Centre, Universiti Teknologi Mara (UiTM).

REFERENCES

[1] Agrawal, M., & Mishra, P. (2012). A comparative survey on symmetric key encryption techniques. *International Journal on Computer Science and Engineering (IJCSE)*, *4*(05), 877-882.

[2] Amato, A., Di Martino, B., & Venticinque, S. (2012, July). Semantically augmented exploitation of pervasive environments by intelligent agents. *In Parallel and Distributed Processing with Applications (ISPA)*, 2012 IEEE 10th International Symposium on (pp. 807-814). IEEE.

[3] Hariri, M., Aldhubaib, R., Qurashi, R. J., Khan, F. Q., Fazil, M. (2012). *Fourth Annual Undergraduate Research Conference on Applied Computing (URC 2012)*.

[4] Pahal, R., & Kumar, V. (2013). Efficient Implementation of AES. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(7), 290-295.

[5] Saranya, K., Mohanapriya, R., & Udhayan, J. (2014). A Review on Symmetric Key Encryption Techniques in Cryptography. *International Journal of Science, Engineering and Technology Research (IJSETR)*, *3*(3), 539-544.

[6] Singh, A., & Gilhotra, R. (2011). Data security using private key encryption system based on arithmetic coding. *International Journal of Network Security & Its Applications (IJNSA)*, *3*(3), 58-67.

[7] CGI. (2004). Public Key Encryption and Digital Signature: How do they work? [White Paper]. CGI Group Inc.

[8] Tripathi, R., & Agrawal, S. (2014). Comparative study of symmetric and asymmetric cryptography techniques. *International Journal of Advance Foundation and Research in Computer (IJAFRC)*, *1*(6), 68-76.

[9] Zhou, X., & Tang, X. (2011). Research and implementation of RSA algorithm for encryption and decryption. *International Forum on Strategic Technology (IFOST)*, 6(2), pp. 1118-1121.

[10] Wooldridge, M. (2002). An Introduction to MultiAgent Systems. J*ohn Wiley & Sons*. LTD

[11] Parkin, S., Driss, S., Krol, K., & Sasse, M. A. (2015). Assessing the User Experience of Password Reset Policies in a University. In *International Conference on Passwords* (pp. 21-38). Springer International Publishing.