Journal of Positive School Psychology
2022, Vol. 6, No. 3, 8461–8469

Integration of Security Hardware Module Zymkey 4i With Raspberry Pi

Rizzo Mungka Rechie[1]; Nur Nabila Mohamed2; Yusnani Mohd Yussoff[3]; Lucyantie Mazalan[4]; Suhairi Mohd Jawi[5]; Mohd Saufy Rohmad[6]

[1,3,4,6]College of Engineering, Universiti Teknologi MARA Shah Alam, Shah Alam, 40450 Selangor, Malaysia
[2]Faculty of Engineering and Built Environment, Mahsa University, Bandar Saujana Putra, 42610, Jenjarom, Selangor, Malaysia
[5]Cybersecurity Malaysia, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor, Malaysia
Email: [1]mungkarizzo@gmail.com, [2]nurnabila.m@mahsa.edu.my, [3]yusna233@uitm.edu.my, [4]lucyantie@uitm.edu.my, [5]suhairi@cybersecurity.my, [6]saufy@uitm.edu.my

Abstract

The use of Raspberry Pi as a personal computer for daily or office use has been perceived as a new norm in the past few years. The motivation for this is the reliability and the reasonable price for it. As the usage of Raspberry Pi has increased especially along the IoT related industry, the demand of hardware security towards the Raspberry Pi has also increased. As Raspberry Pi are relatively new for some users, the implementation of Zymkey 4i as the security module is an added advantage when the Raspberry Pi is being implemented in a large number at one institution. The implementation of the hardware security module had shown great result on securing the Raspberry Pi without it being monitored regularly.

Key-words: Raspberry Pi, IoT, Computer Security, Zymkey, HSM.

I. INTRODUCTION

Raspberry Pi is a microcontroller that can run as a Home Digital Voice Assistant (HDVA), Smart TV or a personal computer. The usage of Raspberry Pi has been growing commercially from time to time in the past few years. The use of Raspberry Pi varies in application such as video servers (Salih & Mysoon Omer, 2018), home security systems (Pi & B, n.d.), CCTV monitoring devices (Rohadi et al., 2018) and image processing devices (Gopal & Vijayashree.T, 2017) have been widely used in everyday life or work life. The use of Raspberry Pi is not just for the IoT study where it is also being used for normal users that want to have low-cost computers (Balon & Simic, 2019) in their homes or in their organisations. The use of Raspberry Pi has shown its great advantage for any user that wants it for its compactness, mobility, flexibility, and cost-efficient features.

The wide use of Raspberry Pi in this IoT era does have its added advantages such as in automation, communication, and data transfer. However, as the internet connects everyone through multiple devices, security issues will follow. The security issue pertaining to the Raspberry Pi is mostly on the device itself which is related to malware attack through SD cards or USB drives and the devices' default configurations from the start. This results in most of the normal users that lack in information on how to keep the Raspberry Pi up to date manually since some Raspberry Pi security updates have to be monitored manually. Malwares can be easily installed on devices if eh devices are not regularly checked by the users or administrators. This results in

the introduction of a hardware-based security module, the Zymkey 4i where this device can be attached on the Raspberry Pi General-Purpose Input/Output (GPIO) pins for it to be integrated fully. It boosts the security features on the Raspberry Pi devices, mainly towards the data encryption and tampering detection features, such as prevention of SD cards to be modified and the possibilities of the Raspberry Pi devices to be breached. This study has shown the impact of the Zymkey 4i towards the Raspberry Pi in terms of physical security and tampering prevention. This study highlights the implementation of the Zymkey 4i towards the Raspberry Pi towards the improvement of security features in the device. This security countermeasure feature will show its' energy usage as well as its successful implementation in terminal.

II. . MOTIVATION OF STUDY

This study serves to provide users with the integration analysis between Zymkey4i and Raspberry Pi devices. Compromised devices can lead to negative effected towards users who are unaware to securing their devices. With this in mind, compromised Raspberry Pi devices such as computers or IoT end devices is very dangerous because the malwares injected into the devices can monitor the devices' internet activities. As a solution, a hardware security module has been proposed as a countermeasure feature to protect the Raspberry Pi devices. Following are the motivation of this study:

Raspberry Pi devices' system security is hard to be monitored by normal users which means users that do not have the knowledge of utilising the Linux terminal. With cases such as the Raspberry Pi devices being compromised, the hardware module Zymkey 4i will overcome this problem by performing automatic encryption and secure boot for it to secure the Raspberry Pi devices from being compromised by data stealing or tampering activities.

The study of the integration between Zymkey 4i and Raspberry Pi have not been formally conducted yet, whereby some users need the correct information regarding on how the integration is performed.

III. RELATED WORKS

a) Raspberry Pi for Personal Use

The Raspberry Pi is a small single board computer that targets robotics and computer science educational promotion. The Raspberry Pi can be used for various projects such as Soil Moisture Monitoring System (Dewi et al., 2017) and a Personal Assistant Robot (Hameem Shanavas et al., 2018). These projects have proven that the usefulness of the Raspberry Pi in projects are as being crucial and highly advantageous. The main reasoning for this is the open source for the framework of the Raspberry Pi OS and the low cost of the device itself that motivates users to worry less about any damages done to it when running any project. With this in mind, users can buy the device again with cheap price rate. Taking consideration of all the applications made by using Raspberry Pi, the common main application that can be made by using it is for it to become a personal desktop computer. One example can be made by using the Raspberry Pi is to make a high quality, low-cost computer for educational purpose (Yamanoor & Yamanoor, 2016). This approach towards education is a new milestone for it as the use of Raspberry Pi has been showing greater performances than any other personal computers or laptops within its price range.

b) Vulnerability of The Raspberry Pi

The vulnerability of the Raspberry Pi devices arises when there is an absence of user maintenance towards it. With this in mind, the maintenance consists of software updates, operating system updates and device status monitoring. This maintenance usually is not be performed regularly due to the users' lack of skill sets to update the system into its' current version. This reason to the added complexity or manual updates performed towards the Linux system.

The vulnerabilities found in the Raspberry Pi is at the GPIO Logic Levels and Serial Access (Sainz-Raso et al., 2019). This study shows that

the GPIO pins of the Raspberry Pi use 3.3V and it does not have any protection in the case of overvoltage. With this in mind, the possibilities of the devices security breach tampering scenario might happen. The use of hardware security module to counter the tampering is much needed to secure the system. Additional to that research, the minor security vulnerability that can make a big impact is the default username and password of Raspbian and Ubuntu OS that has been tested on the device is found to be the key factor of the device breach. This is due to the additional complexity or lack of Linux skills to enhance the security. With this in mind, with Ubuntu, due to the remote use of Raspberry Pi, the use of SSH is implemented in the device software and it gives effects towards the devices' vulnerability. With this reasoning, it opens one of the devices' ports and that gives the attackers an additional door to hack the system or the devices. The type of attack or scan used to check the system vulnerability is the Nmap where this software enables the user to check the targeted system list of vulnerability before they can proceed to another type of attack.

c) Hardware Security Module

The hardware security module is a physical module that enhances the security of electronic devices such as a computer. The need of a hardware security module (Doerner et al., 2018) to be implemented into any device is crucial when the used device contains a highly confidential data. Eventhough from time to time the system software grows better, the need and the superiority of the hardware security module is still high. With this in mind, one of the advantages of hardware security module is the secured and the protected cryptographic keys (Kim et al., 2014) inside the hardware security module. This reasons towards the absence of hardware security module inside a processor, whereby the cryptographic keys which is responsible to store the computer password is contained inside the hard drive, which is easier to be accessed than the hardware security module.

d) Zymkey 4i

The Zymkey 4i is a hardware security module for Raspberry Pi. Some of the key factors that the Zymkey has are the multifactor device identify and authentication, data encryption, signing engine, key generation, secure key storage and physical tamper detection features. These features enhance the security features into the Raspberry Pi for both hardware and software spectrums.The security of the devices are well stores inside the Zymkey 4i, which leave the Raspberry Pi to be more secured in terms of the devices' cryptographic key's storage and device tampering. The features that the Zymkey 4i provide towards the Raspberry Pi when they are integrated are the SD card encryption feature to protect the data and credentials.Also, the device provides an autonomous security for unattended IoT devices. With this in mind, it also provides security when we are surfing online with Amazon Web Services which the device provides a secure device registration when the users are surfing. These features such as the SD card encryption and autonomous security do boost the devices' data encryption, and the key that is able to decrypt the data is stored inside the Zymkey 4i.

Other than that, the algorithms used by the Zymkey 4i are the ECDSA: FIPS186-3 Elliptic Curve Digital Signature Algorithm and ECDH: FIPS SP800-56A Elliptic Curve Diffie-Hellman Algorithm. The usage of the Elliptic Curve Digital Signature (Abidi et al., 2014) is to authenticate any digital content sent by a known sender. With this in mind, by using the ECDSA, the sender cannot deny that they have sent the message and the message was not altered during the transit. With this in mind, the Elliptic Curve Diffie-Hellman Algorithm is a secured key exchange algorithm used to exchange keys securely online. Elliptic Curve Diffie-Hellman Algorithm (Kodali & Naikoti, 2017) is used to share the confidential key and ECDSA (Sankar et al., 2011) is used to authenticate the content. With this in mind, the ECDH does not provide authentication, by that the ECDSA is used for that purpose. Once the secret key is shared, a secure data exchange is established.

IV. TESTING REQUIREMENTS

Figure 1 - Flow chart of the system integration

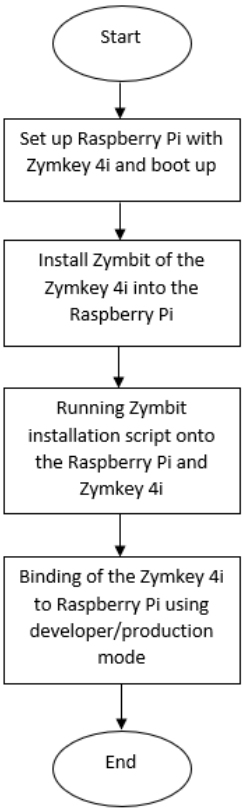

Figure 1 shows the flow chart of the system integration between the Raspberry Pi and the Zymkey 4i. Regarding this, the Raspberry Pi must be set up first with monitor, keyboard, and mouse ready along with the Zymkey 4i attached to its GPIO pins. After the Raspberry Pi is switched on, the Zymbit installation script will be made at the terminal to enable the integration of both of the devices. The Raspberry Pi will restart and show its' successful installation stahe or integration indicator by flickering its blue LED at the Zymkey 4i. This integration can be confirmed by running python to get the ECDSA public key and by testing a random block generation, encryption, and decryption. After the device has been successfully integrated, the process of device binding is made. The device binding process is being made between both devices which are the Raspberry Pi and Zymkey 4i. This binding process will enable the features of device tampering detection. One of the main highlights in this feature is when anyone tries to change the Zymkey 4i or SD Card of the Raspberry pi, the device will not boot up. This feature boosts the system physical tampering prevention which is a good additional feature. With all of these being executed, the integration between the Zymkey 4i and the Raspberry Pi is complete.

Figure 2 – System Block Diagram

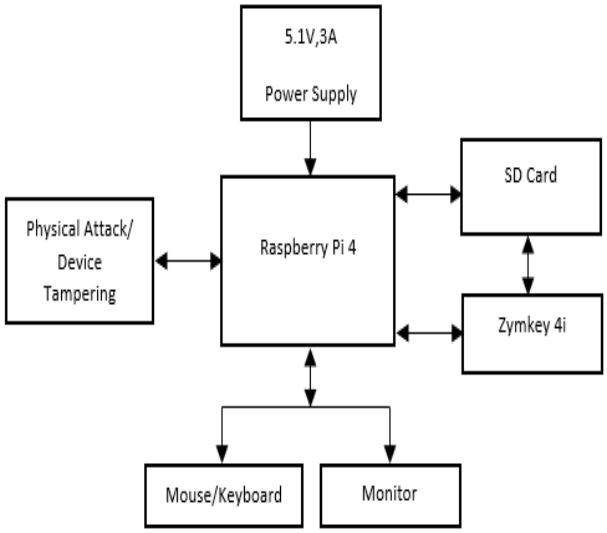

Figure 2 shows the system block diagram. The Raspberry Pi 4 is supplied by 5.1V and 3A along with it are the mouse, keyboard, and a monitor. These are the basic equipment and supplies needed by the Raspberry Pi to run as a computer in desktop mode. Provided that, the security system is not enough when the device is being tampered or compromised. The Zymkey 4i is to be added to the Raspberry Pi. The reason why the SD Card, Raspberry Pi, and Zymkey 4i is interconnected to each other is because the binding process that is being made towards each other. During the binding process, the Zymkey 4i run a script to make all these 3 dependent devices to become one permanent ID of a host system. The Zymkey unique features which is to generate a unique identity (ID) for the host system is based upon a fingerprint that measures specific system components. Once all the devices have gone through the binding process, the alteration of any of these devices will cause the Raspberry Pi as a whole will not function accordingly.

Figure 3 - System Boot Flow

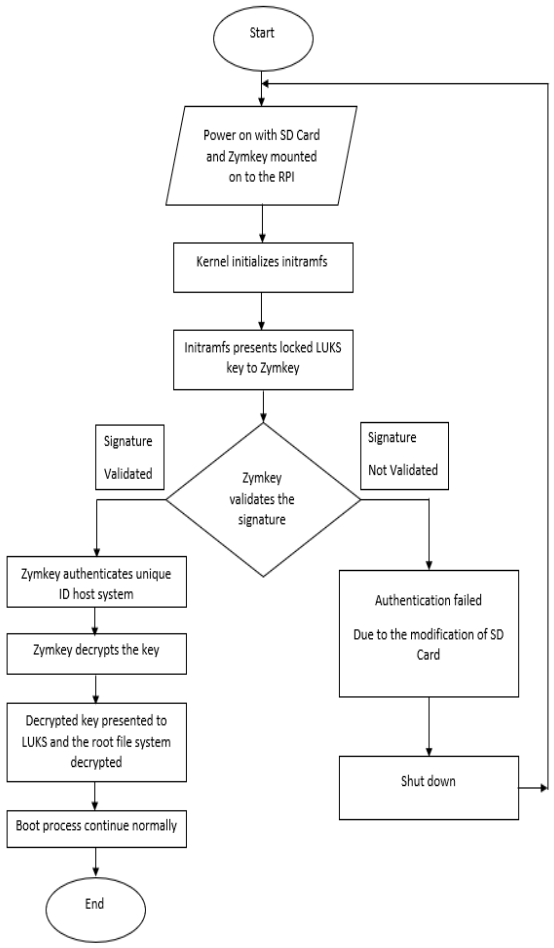

Figure 3 shows the system boot flow when the Raspberry Pi had been integrated with the Zymkey 4i. There are two situation that might happen to the Raspberry Pi when it boots up. First situation is when the device is not being altered or tampered where the device will boot up as usual. The second situation is when the device had been altered or tampered where the Zymkey 4i had been changed with a false Zymkey 4i, by that the Raspberry Pi will cancel its boot up and shut down the system automatically. The reason why the Raspberry Pi shut down when there is device tampering takes place is because the Zymkey 4i did not recognise the signature of the new added device. Any modification towards the Raspberry Pi, SD card and Zymkey will cause the unique ID signature to change.

Zymkey provides a general locking service whereby a block of plaintext data is encrypted and signed. When LUKS is used, the User Key is sent to the Zymkey 4i to be encrypted and signed when the file system is created. When the system boots and needs to decrypt the root file system, the locked LUKS key signature is to be verified and the contents will be decrypted, and then presented to dm-crypt. Dm-crypt is a transparent disk encryption subsystem in Linux kernel and part of the device mapper infrastructure. If the key was verified and unlocked successfully, the boot process will continue normally. Every time the Raspberry Pi boots up, the Zymkey 4i will recheck the unique ID fingerprint to be verified before locked LUKS key signature to be unlocked and the root file system to be decrypted. If any of the bonded component has been altered, the system is deemed to be compromised, and all the system will shut down due to the failed authentication.

V. INTEGRATION RESULT

Table 1 - Table of System Energy Usage

Mode	V	I
Raspberry OS	238.8	0.033
Alexa running	240.3	0.38
Zymkey 4i insert	240.6	0.035
Alexa with Zymkey 4i insert	239.6	0.038
Testing Uitils Run	239.9	0.034
Crypto.G	239.9	0.034
Crypto.Enc	239.8	0.034
Crypto.Dec	239.9	0.034
Zymkey 4i pulled out	237.7	0.051

Table 1 shows the energy usage of the system. Based on the table shown on top, do note that the testing utils is a script to run utilities test in Zymkey 4i and Raspberry Pi to make sure that the device is properly installed and integrated successfully. The Crypto.G, ENC, and DEC are the names of the scripts used to run generate cryptographic block, encode the block, and decode the block. These scripts are the necessary scripts to make sure that the Raspberry Pi and Zymkey 4i are properly installed and integrated with each other by taking into consideration the table above, the voltage is consistent to run at 240V with 3 volts

of error. Other than that, the current is running between 0.033 to 0.051 which only shows the difference of 0.018A. The energy consumption when the device runs crypto block generation, encryption and decryption is low and it does not have much differences compared to when the device is only running the Raspberry Operating System in its 'idle state. This has shown that the implementation of Zymkey 4i in Raspberry Pi is feasible in term of energy consumption, as the energy consumption does not show any high energy usage after the Zymkey 4i has been implemented.

Table 2 – Table of System RAM Consumption

Mode	**RAM %**	**RAM (3.81G)**	**Swap (100.0M)**
Raspberry OS	6.03%	230	0M
Alexa running	6.64%	253	0M
Zymkey 4i insert	5.46%	208M	0M
Alexa with Zymkey 4i insert	5.54%	211M	0M
Testing Uitils Run	4.59%	175M	0M
Crypto.G	4.65%	177M	0M
Crypto.Enc	4.62%	176M	0M
Crypto.Dec	4.65%	177M	0M
Zymkey 4i pulled out	-	-	-

Table 2 shows the Random-access memory (RAM) usage of the system when it is running the OS before and after the Zymkey 4i has been installed into the system. When the Raspberry Pi is running its' OS with or without the Zymkey 4i, the RAM usage is still showing low percentage values between 5 - 7 percent. The Raspberry Pi has also been tested when it is running Amazon Alexa to test the system RAM usage, and the system is still consuming low amount of RAM. As observed, the Raspberry Pi does not use swap memory in case the memory requires multi-channel to run the system and programs. Throughout the Zymkey 4i crypto test and the Alexa mode, the Raspberry Pi only requires the use of RAM instead of using the swap memory for RAM backup. This shows that although the presence and implementation of the hardware security module which is the Zymkey 4i, the Raspberry Pi still runs smoothly and the RAM consumption has been shown to be low, maxing up to only 7 percent of its RAM capacity. All in all, the system analysis on the energy and memory usage have shown that the Zymkey 4i integration with Raspberry Pi is feasible.

Figure 4 - Zymkey 4i Block Generation

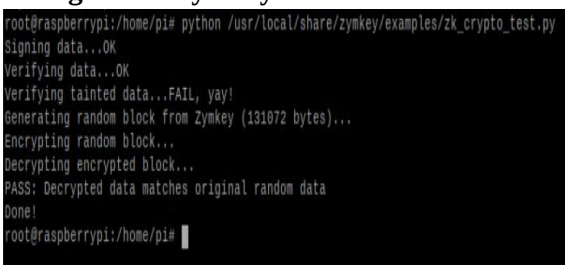

Figure 4 shows the crypto test performed towards the Zymkey 4i when it is being integrated with HDVA. A random block is generated from the Zymkey 4i, then block encryption and decryption are being executed to check if the latest decrypted data matchesthe previous original data. This has shown that the integration between the Zymkey 4i and HDVA is successful and the Zymkey 4i can be configured with the HDVA to make it more secure. This block generation encryption and decryption process from Zymkey 4i is being applied to the HDVA when it is properly configured specifically for tampering detection feature.

Figure 5 - HDVA System Boot Unsuccessful Flow

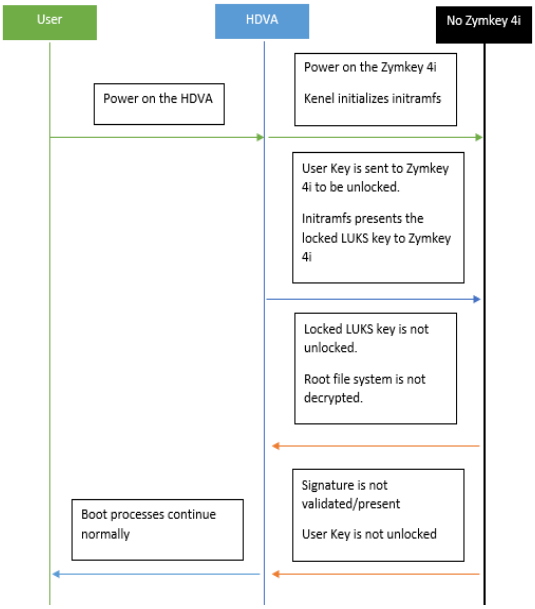

Figure 5 shows the system boot up process. As a result of the missing presence of Zymkey 4i inside the HDVA, the process of HDVA booting up is stopped after the initramfs failed to present the user key to Zymkey 4i to go the unlock phase. The boot process failed due to the device has detected that the Zymkey 4i is missing in the bonded preconfigured HDVA.

The process differs when the HDVA has the correct Zymkey 4i device being connected to it, where the HDVA will boot up properly as the private key is present and available to unlock and decrypt the encrypted root file system data storage. This process can avoid device tampering, as any bonded device such as Zymkey, Raspberry Pi or SD card missing, the device will not boot properly.

VI. . CONCLUSION

In conclusion, the integration between Zymkey 4i and Raspberry Pi has shown good results in terms of compatibility and implementation for its' future use. The study has shown that the implementation between the Raspberry Pi and Zymkey 4i is has granted the device with added security features. Without the added security features, users who are unaware on how to secure theirs device might have their device scompromised without them even knowing about it. The solution is the hardware security module that acts as a countermeasure to protect the device which is the creation of Zymkey 4i. This has further solved some of the device's security issue and security motivation which is, the Raspberry Pi system security is hard to be monitored by normal users. With cases such as the Raspberry Pi devicesbeing compromised, the hardware module Zymkey 4i will overcome this problem by performing automatic encryption and secure boot up for it to secure the devices from being compromised by data stealing or tampering activities. With the study of Zymkey 4i and Raspberry Pi being conducted, this paper will provide researchers correct information regarding it on how the integration is performed and executed.

VII. ACKNOWLEDGEMENT

The author would like to express highest gratification to the Lord, families and friends that have given their full support throughout the study. Special thanks to Faculty of Electrical Engineering, Universiti Teknologi MARA, UiTM Shah Alam for the support in this research. Additional thanks to Geran Penyelidikan Lestari no: 600-RMC/MyRA 5/3/LESTARI (098/2020) for the endless support throughout this research.

BIBLIOGRAPHY

1. ABIDI, A., BOUALLEGUE, B., & KAHRI, F. (2014). Implementation of elliptic curve digital signature algorithm (ECDSA). 2014 Global Summit on Computer & Information Technology (GSCIT), x, 0–5. https://doi.org/10.1109/GSCIT.2014.6970118
2. BALON, B., & SIMIC, M. (2019). Using raspberry Pi computers in education. 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2019 - Proceedings, March 2018, 671–676.

https://doi.org/10.23919/MIPRO.2019.08756967

3. DEWI, L. P., ANDJARWIRAWAN, J., & WARDOJO, R. P. (2017). Android application for monitoring soil moisture using raspberry Pi. Proceedings - 2017 International Conference on Soft Computing, Intelligent System and Information Technology: Building Intelligence Through IOT and Big Data, ICSIIT 2017, 2018-Janua, 178–184. https://doi.org/10.1109/ICSIIT.2017.63
4. DOERNER, J., KONDI, Y., LEE, E., & SHELAT, A. (2018). Secure Two-party Threshold ECDSA from ECDSA Assumptions. Proceedings - IEEE Symposium on Security and Privacy, 2018-May, 980–997. https://doi.org/10.1109/SP.2018.00036
5. GOPAL, A., & VIJAYASHREE.T. (2017). Authentication of herbal medicinal leaf image. International Conference on Intelligent Computing and Control Systems, 1304–1307.
6. HAMEEM SHANAVAS, I., REDDY, P. B., & DODDEGOWDA, M. C. (2018). A personal assistant robot using raspberry Pi. Proceedings - 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control, ICDI3C 2018, 133–136. https://doi.org/10.1109/ICDI3C.2018.00038
7. KIM, D., JEON, Y., & KIM, J. (2014). A secure channel establishment method on a hardware security module. International Conference on ICT Convergence, 555–556. https://doi.org/10.1109/ICTC.2014.6983209
8. KODALI, R. K., & NAIKOTI, A. (2017). ECDH based security model for IoT using ESP8266. 2016 International Conference on Control Instrumentation Communication and Computational Technologies, ICCICCT 2016, 629–633. https://doi.org/10.1109/ICCICCT.2016.7988026
9. PI, U. R., & B, R. P. M.-. (n.d.). Internet of Things Based Home Security. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 1–6.
10. ROHADI, E., SUWIGNJO, S. A., PRADANA, M. C., SETIAWAN, A., SIRADJUDDIN, I., RONILAYA, F., AMALIA, ASMARA, R. A., & ARIYANTO, R. (2018). Internet of Things: CCTV Monitoring by Using Raspberry Pi. Proceedings - 2018 International Conference on Applied Science and Technology, ICAST 2018, 454–457. https://doi.org/10.1109/iCAST1.2018.8751612
11. SAINZ-RASO, J., MARTIN, S., DIAZ, G., & CASTRO, M. (2019). Security Vulnerabilities in Raspberry Pi-Analysis of the System Weaknesses. IEEE Consumer Electronics Magazine, 8(6), 47–52. https://doi.org/10.1109/MCE.2019.2941347
12. SALIH, F., & MYSOON OMER, S. A. (2018). Raspberry pi as a Video Server. 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering, ICCCEEE 2018, 1–4. https://doi.org/10.1109/ICCCEEE.2018.8515817
13. SANKAR, R., SUBASHRI, T., & VAIDEHI, V. (2011). Implementation and integration of efficient ECDH key exchanging mechanism in software based VoIP network. International Conference on Recent Trends in Information Technology, ICRTIT 2011, 2(2), 124–128. https://doi.org/10.1109/ICRTIT.2011.5972416

14. YAMANOOR, S., & YAMANOOR, N. S. (2016). High Quality, Low Cost Education with the Raspberry Pi. 46, 3–7.