Biometrically Authenticated Boot Loading System from USB Drive by Exploiting the Fingerprint and Finger Vein

Alycia Sebastian(✉), Vijaya Padmanabha, and Joseph Mani

Department of Mathematics and Computer Science, Modern College of Business and Science, Muscat, Sultanate of Oman
{Alycia.sebastian,Vijaya.Padmanabha,drjosephmani}@mcbs.edu.om

Abstract. Commonly, a USB flash drive is utilized for storing, transferring and backing up data like personal files, software, media files, etc. But users might not have much knowledge about its other hidden characteristics. It could also act as a replacement of CD/hard disk/DVD media as OSs handler resources and as a plug and play portable system. However, security is a major concern for external boot system and to fix this issue, numerous solutions were proposed and implemented. Out of all the existing security provisioning schemes, biometric based security solutions are always reliable and hassle free to process. The USB drives now available with fingerprint protection and to come out of the box, this article secures the USB drives with the combination of fingerprint and finger vein. On successful authentication, the user can boot OS from USB. The performance of the work is analysed in terms of FAR, FRR, accuracy and time consumption rates and observed that it achieves greater accuracy rate when compared with other classifiers.

Keywords: Biometrics · OS · USB drive security

1 Introduction

With the augmentation of advancements in computers, multi-stage BLs are implemented to incorporate all its features like scalability, upgradability and flexibility. After power ON or reset processors suchlike Acorn RISC Machine (ARM) and x86 fetch code from a specific address. As time goes by, owing to the advancement in ICs, non-volatile memory like NAND and NOR flash memory are employed to hold BLs on account of their high density and low operating cost, which so lessens the OS load time. The boot program can also be divided into multistage and bootstrap each other which are launched in sequence. The use of flash memory is stipulated rather than Read Only Memory (ROM). Every loader comprises its own specifications. There were disparate implementations of the boot process. The BLs could be a complex, basic, or multi- BLs. On account of limited capabilities in MBR and for including more features, the BL was saved in external or internal optical disk or flash memory.

N. Bacanin and H. Shaker (Eds.): ICIITB 2022, ACSR 104, pp. 250–261, 2023.
https://doi.org/10.2991/978-94-6463-110-4_18

With the rapid growth in the semiconductor devices leading to the availability of different varieties of mediums with USB-FDs top the list is the most convenient and user-friendly devices for multiboot and as a portable system. Though multiple OS can be installed in a computer, it is better to hold the OS which is occasionally used in an external media like USB and utilize it when required. The booting from any external medium is restricted to user knowledge of the boot order change. The current boot process is static and it needs the user to modify the BIOS settings. So, the dynamic boot loader approach was proposed to eliminate this dependency which allows the user to directly boot from portable live USB.

Existing work pertaining to securing the data of the USB are surveyed as security is the main concern as it interacts with different host system based on user need. To offer portability for USB, biometric authentication is the preferred choice for authenticating access to USB boot medium due to its distinctive characteristics of each person. These features can be explored to guard system against fraudulent access and identity theft. As the proposed work targets the people who will need ease in carrying the secured portable system everywhere they go, to avoid any misuse the proposed approach presents a simple authentication procedure based on fingerprint and finger vein. The contributions of the work are as follows.

1. The Live USB can be made bootable with any available free source OS and the proposed dynamic boot loader eliminates the BIOS dependency.
2. This work ensures security by incorporating both fingerprint and finger vein of the users. This idea reduces the false positive and false negative rates.

The remainder of this paper is organized as follows. Section 2 discusses the related review of literature and the proposed approach is discussed in Sect. 3. The performance of the proposed approach is evaluated in Sect. 4 and the paper is concluded in Sect. 5.

2 Related Literature

This section discusses the related literature with respect to the USB security and its protection and USB as external boot medium.

In [1], the security of USB and the technology is discussed. In this work, the USB protocol is analysed with certain possible vulnerabilities and its lack of security to protect against both passive and active attacks. A USB protocol is formed that concerns about both encryption as well as authentication of data on the bus line.

In [2], the dangers encountered by USB pen drives are explored. This work states that the average users do not understand the danger of connecting unfamiliar peripherals to a computer and underestimates the risks thrown by the USB drives. The USB sniffing attack is explained in [3], in which an USB device eavesdrops through all the communications between the host and the other devices. This attack is prevented by implementing a lightweight encryption solution called UScramBle without any need for setting change requirement or any user interference.

In [4], a new mediation architecture namely GoodUSB is proposed for the Linux USB Stack. This technique works against BadUSB attacks by imposing permissions on the

basis of user expectations with respect to functionality of the device. GoodUSB contains a security image component and a honeypot mechanism for observing suspicious USB activities.

A novel technique is proposed in [5] for safeguarding the host through a software/hardware solution namely USBWall. This technique utilizes a cheaper and an open-source computer called BeagleBone Black (BBB), which behaves as a middleware and specifies the devices with respect to the host. A program is done to help out the user for detecting the risk on a device. A simulated USB device with malicious firmware to the USBWall is presented.

The authors in [2] conducted a survey to study on the awareness of users on the dangers of connecting USB to unfamiliar peripherals. The experiment was performed using 100 USB drives containing HTML files with embedded image hosted to the campus server in the University of Illinois. The outcome of the experiment showed that user underestimates the risks of using their flash drive in unknown host system.

In [6], an USB based software attacks and protection solutions are discussed. This work studies the currently identified USB based software attacks on host computers and USB storage devices and the taxonomy of the security attacks. Finally, this work concludes that a multi-layered security solution framework with software implementations at the User Mode layer in the OS supports in reducing the root cause of the problem.

In [7] initiated Unified EFI (UEFI) which was a specification that proffers a software interface in-between the OS and firmware. In the future, the conventional BIOS would be substituted by UEFI. TCG has evolved as a hard research domain in the computer security domain. That declared the need of Trusted Bootstrapping (TBS). Here, a notion of TBS utilizing the USB key was proffered which encompasses the framework of the Portable Trusted Platform Module, supported with UEFI. It targeted to diminish motherboard modification and turned the system extremely less vulnerable to human disruptions.

USB is devised as a highly secured portable boot medium with fingerprint authentication to ensure data security in [8]. The features database is constructed by extracting features using Local Directional Pattern (LDP) and Histograms of Oriented Gradients (HOG) features and classification and recognition is performed using random forest classifier. The proposed approach shows significant performance over other algorithms. The experimental result produces the low false rate when compared to the analogous approaches.

A dynamic boot loader for loading an OS from an external medium is designed in [9, 10]. The proposed boot loader eliminates the BIOS dependency by allowing the user to boot from USB without any BIOS boot order change. The proposed USB system is made portable with puppy Linux and the dynamic boot loader automatically detects the connected portable system which allows the user to directly boot from the USB system. This approach of removing BIOS dependency proves nearly 50% reduction in booting time from an external system. The time spent by the user in changing the boot order settings is eliminated compared to the existing static boot loader available.

From the existing literature, it is observed most of the works focus to build security solutions for USB devices through scans and tracking the behaviour of the USB devices. However, the process of user authentication in USB devices is scarcely discussed in the

existing literature. Besides this, devising a USB as a portable system is more convenient and hassle-free to the users, for why this article focuses on this aspect as well.

3 Proposed Boot Loader with Secure USB

The objective of this article is to provide security to the portable USB drive, which is accommodated with the OS. Though the term security has multiple facets, this article stresses on the authentication-based security. Authentication is one of the most fundamental security requirements of any confidential application, such that the application can be gained access upon proving the user's identity [11]. This means that only the legitimate users can gain access to the system, while the illegitimate users are denied access. Hence, the application serves its purpose only for the intended users.

This article is carried out by splitting the complete work into two stages, which are biometric security provision and OS setup in USB drive with secure boot load. Initially, the USB is made portable with open-source OS and the biometric security provisioning is performed by combining fingerprint and finger vein. This idea is to avoid unauthorised access to the confidential data stored in USB. The overall flow of the work is depicted in the Fig. 1.

Though there are several fingerprint authentication based USB drives available in the market, there are several drawbacks associated with the utilization of single biometric. Recognizing the drawbacks, this work proposes to combine two different biometrics yet simple to collect and they are fingerprint and finger vein. A special scanner is utilized to collect both these biometrics. The OS is loaded only when the authentication is done perfectly. All these stages are described in the following sections.

3.1 Biometric Security Provision

Due to the convincing reliability of biometric based security, this article chooses biometric security over several traditional security provisioning algorithms. Usually, the USB drives are protected through passwords or pins. However, the passwords or pins can easily be stolen by the malicious users and hence better security solutions such as hashing is utilized for many USB drives. The main drawback of hashing-based technology is the computational complexity and prior knowledge about hashing.

On the other hand, biometrics are easy to collect and process yet reliable. The biometrics can be neither duplicated nor copied by malicious people, which brings in more security. Additionally, the basic concept is simple when the recorded biometric is matched with the current one, then the access is granted to the legitimate user. In the previous work [8], fingerprint alone is utilized to ensure authentication. Though the performance of the work is satisfactory, the False Positive (FP) and False Negative (FN) rates are planned to be minimized further.

Relying on a single biometric is not a good idea for ensuring better security and hence, utilization of multiple biometrics is always safer to avoid wrong access grants. However, incorporation of multiple biometrics demands extra effort to match the samples and the time consumption may be increased. Hence, the choice of multi-biometrics must be made carefully such that the time and computational complexity can be reduced as

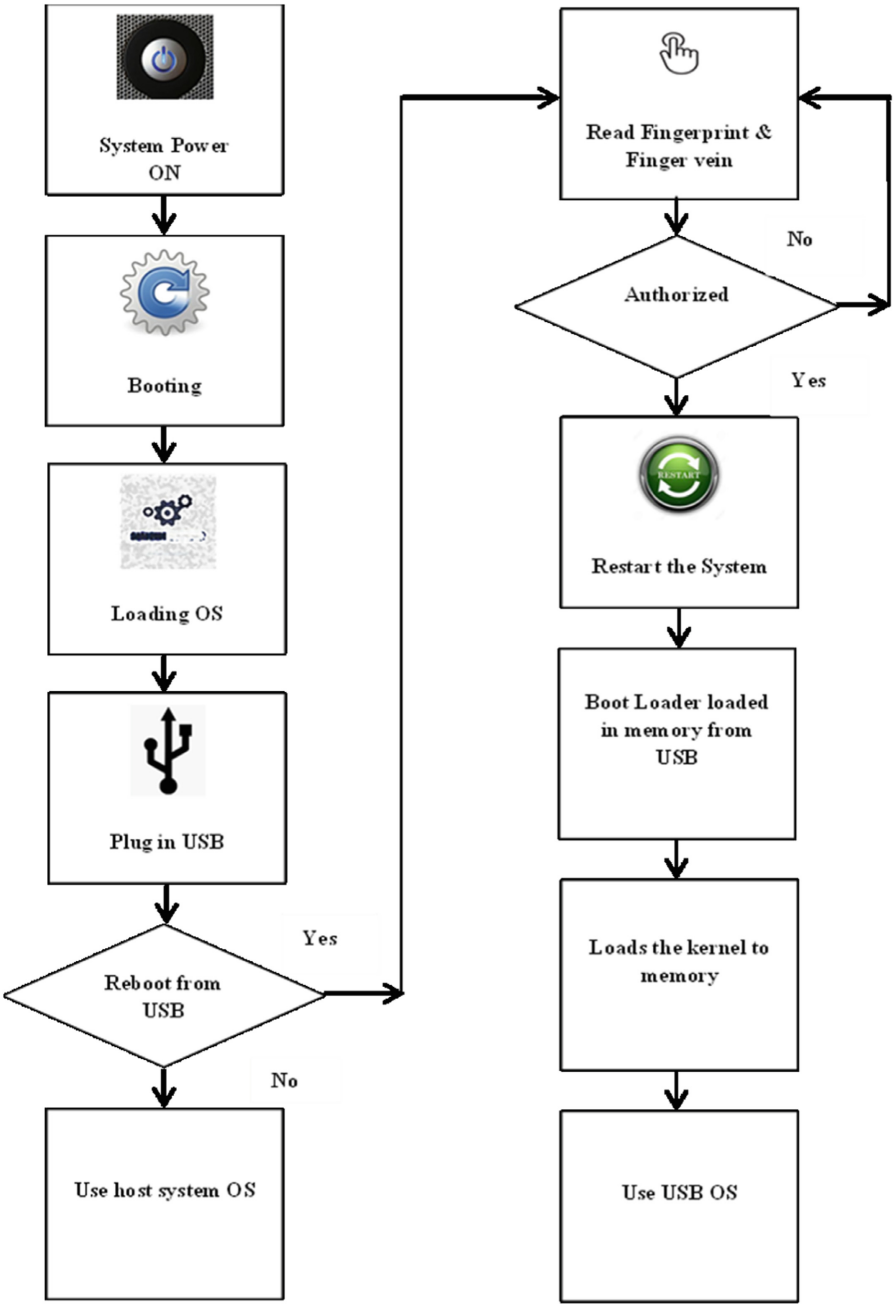

Fig. 1. Overall flow of the work

much as possible. Recently, an USB device with iris recognition is introduced [12], which is proven to be greatly secure. On the negative side, it is very expensive and requires user's proper cooperation for data collection every time.

Conversely, fingerprint can be captured on the go, without any special user cooperation. Yet, fingerprint cannot support cent percent accuracy rates, due to several reasons such as poor or noisy fingerprints. In order to overcome this issue, another simple but potential biometric is required and the choice of this work is finger vein. The finger vein is also easy to capture, effective and available for all the individuals. The finger vein-based authentication is more powerful than fingerprint biometric. Usually, the fingerprints may fade out due to aging, weather conditions and so on, which is not the case in finger vein. As this work incorporates both the fingerprint and finger vein together, the security is much more enhanced.

The fingerprint is processed as in the case of the previously proposed work [8]. The fingerprint is processed to extract the Local Directional Pattern (LDP) and the Histograms of Oriented Gradients (HOG) features. The same set of fingerprint features are utilized for this work and the finger vein recognition is focussed in this work.

Though finger veins are more potential than fingerprints to prove authenticity, they are difficult to process due to the blood flow and the intricate pattern of veins. This makes it necessary to extract the area of interest and then the features are needed to be extracted. This idea regularizes all the finger vein images and the Gabor Local Vector Pattern (GLVP) features are extracted. The features extracted from both the fingerprint and finger vein are clubbed together. This section discusses about the area of interest extraction and feature extraction parts of finger vein, followed by which the classification is done by the previously presented random forest classifier [4].

3.1.1 Area of Interest Extraction in Finger Vein

Finger vein reader scans the complete finger for capturing the vein pattern of the finger. However, the work needs not to process the complete finger vein and the most significant area of the finger vein is extracted. The process of extracting area of interest helps in reducing the computational and time complexity. Initially, the edges of the finger image are computed by Sobel edge detection approach [13]. As the edges are detected, the significant area of the finger is detected by finding the upper and lower regions of the finger by applying a mask with window size 4×20 (Fig. 2).

Now, the finger portion between the upper and the lower boundaries is detected and the in-plane rotation compensation is applied to make the finger vein straight and to alleviate certain left and right portions to escape from the finger thickness variations. This idea helps in extracting features in a better way [14]. The sample area of interest extracted image is depicted in Fig. 3. Hence, the area of interest is extracted from the finger vein image and then the GLVP features are extracted from it, which is presented in the following section.

3.1.2 GLVP Feature Extraction

GLVP is an efficient feature descriptor as it is based on both gabor filter and LVP. At first, a 5×5 sized gabor filter is applied on the finger vein image followed by which LVP

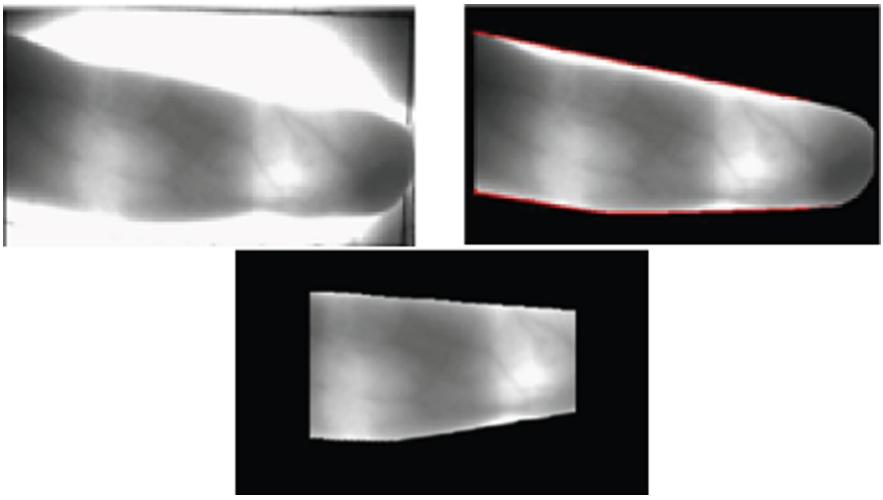

Fig. 2. (a) Original input image (b) Edge detected image (c) Area of interest extracted image

is applied. Gabor filter is very popular in detecting the edges and extracting the texture features [15]. In order to increase the efficiency, the gabor filter is combined with LVP [16]. The gabor filters are created by the following equation.

$$gf = f(a, b, \omega, \theta, \sigma_a, \sigma_b) \tag{1}$$

$$f(a, b, \omega, \theta, \sigma_a, \sigma_b) = \frac{1}{2\pi\sigma_a\sigma_b}\exp\left[\frac{-1}{2}\left(\frac{a}{\sigma_a}\right)^2 + \left(\frac{b}{\sigma_b}\right)^2 + k\omega(acos\theta + bsin\theta)\right] \tag{2}$$

Where ω is the frequency, θ is the orientation, (a, b) is the pixel coordinates, σ is the spatial width. The LVP is then computed by considering different angles of pixels such as 0,45 and 90 degrees for every corresponding pixel. This can be represented by

$$lvp_{a,d}\left(Cor_p\right) = \{lvp_{a,d}|ang = 0°, 45°, 90°\} \tag{3}$$

$$lvp_{a,d}\left(Cor_p\right) = \{lvp_{a,d}|dis = 1, 2, 3\} \tag{4}$$

In the Eqs. (3) and (4), *ang* and *dis* represent the angle and distance between the corresponding and the neighbourhood pixels respectively. Now, the GLVP is computed by combining both the LVP and gabor features as in Eq. (5).

$$GLVP_{Cor_p} = gf \cup lvp \tag{5}$$

Where LVP is computed by

$$lvp = lvp_{a,d}\left(Cor_p\right) \tag{6}$$

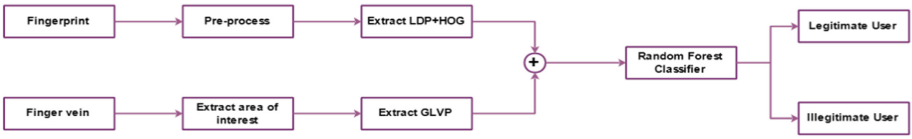

Fig. 3. Biometric security provisioning scheme

The combination of LVP and gabor features proves better performance, as all the intricate information of the finger vein image is considered. The flow of biometric security provisioning is depicted in Fig. 4.

The collected features are reduced by means of IGR [8]. Now, the features of finger print and finger vein are combined together for forming the final feature set of this work.

$$FS = F(fingerprint) \oplus F(fingervein) \tag{7}$$

The final FS is built by concatenating the features of both the biometrics being utilized by this work. The final FS is utilized for training the random forest classifier [8] for granting access to the USB drive. Hence, this section is completely meant for providing biometric security to attain secure boot load.

3.2 Secure Boot Load

The USB is bootable from a BIOS or UEFI machine with different distributions of Linux OS. The basic requirement to carry out this work is an USB drive with 8 GB storage with 4 GB of persistence. The default File System (FS) for USB is File Allocation Table 32 (FAT32) as it is compatible with all types of OS. The dynamic boot loader is developed to offer maximum flexibility that when boot loaded overlooks the BIOS boot priority settings. Live-USB is boot priority first. DBL is executed utilizing VB script and Windows Management Instrumentation (WMI) code which is executed as a batch file. The proposed boot process is as follows:

1. On recognizing a USB plug in, it prompts the user to scan fingerprint and finger vein for authorization. On acceptance, the proposed boot loader using WMI, a core Windows management technology and WMI command line obtain information about the device identifier of the bootable medium.
2. It calculates the sector address of USB boot loader and load it at memory address 0044-0047 of stage1 boot loader in MBR of hard disk.
3. It the reads the first 512 bytes of USB MBR and store it as bin file.
4. Using the device identifier, it edits the BCD to add the new entry of USB OS and link its path to the bin file.
5. On reboot, the windows boot menu displays the USB OS, then the user is prompted to select the UBS OS to boot directly; there is no requirement to change the BIOS settings.

The performance of the proposed biometric approach is evaluated in the following section.

4 Results and Discussion

The performance of the work is validated with respect to False Acceptance Rate (FAR), False Rejection Rate (FRR), booting time, loading time and so on. The FAR and FRR are computed by employing the fingerprint, finger vein individually, combination of fingerprint and finger vein. The booting and loading time analysis comparison is already discussed in the previous work [9]. The following subsections discuss about the FRR and FAR rates.

4.1 FAR and FRR Rates

The FAR and FRR rates are the most sensitive performance measures of any security provisioning applications. Increased FAR represents access grant to the USB drive for illegitimate users, while greater FRR indicates the access denial to the legitimate user. Both these scenarios are extremely serious and FAR is still more dangerous. This is because, when the illegitimate users gain access to the secured system, the integrity of the system may get collapsed. Hence, this work intends to reduce the FAR and FRR as much as possible by clubbing the powerful biometrics such as fingerprints and finger veins. The performance of the combination of fingerprint and finger vein is proven in the following Fig. 4.

From the experimental results, it is evident that the combination of fingerprint and finger vein shows minimal FAR and FRR with greatest accuracy rates. It is always recommendable to use multiple biometrics, which could complement each other. There are some scenarios in which the fingerprint cannot be captured perfectly due to weather conditions or the person may have some external injury on the already enrolled finger. In such scenario, the finger vein could complement the situation. Similarly, Table 1 presents the results by employing LVP, Gabor and GLVP to note the performance. The experimental results given in Table 1 are meant for finger vein alone.

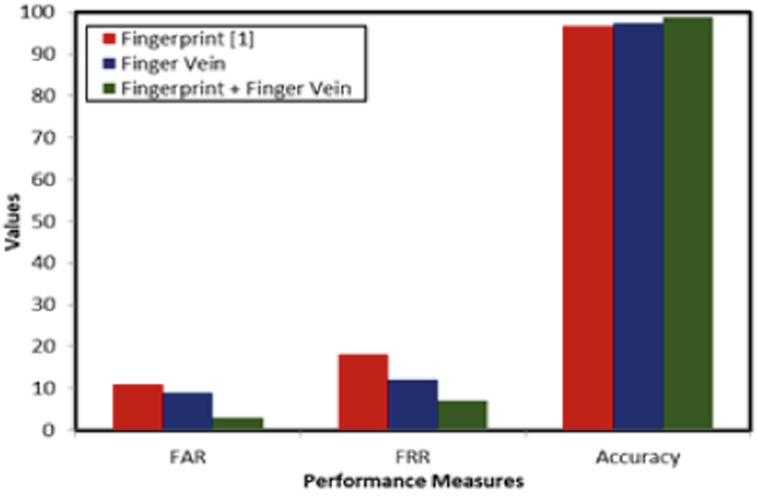

Fig. 4. FAR and FRR analysis

Table 1. Comparative analysis w.r.t finger vein recognition

Performance Metrics/Techniques	FAR	FRR	Accuracy (%)	Time consumption (s)
Gabor + Random forest	21	24	91.6	2.8
LVP + Random forest	19	22	93.2	2.0
GLVP + Random Forest	28	26	92.6	4.2
GLVP + IGR + Random Forest	**9**	**12**	**97.4**	**3.9**

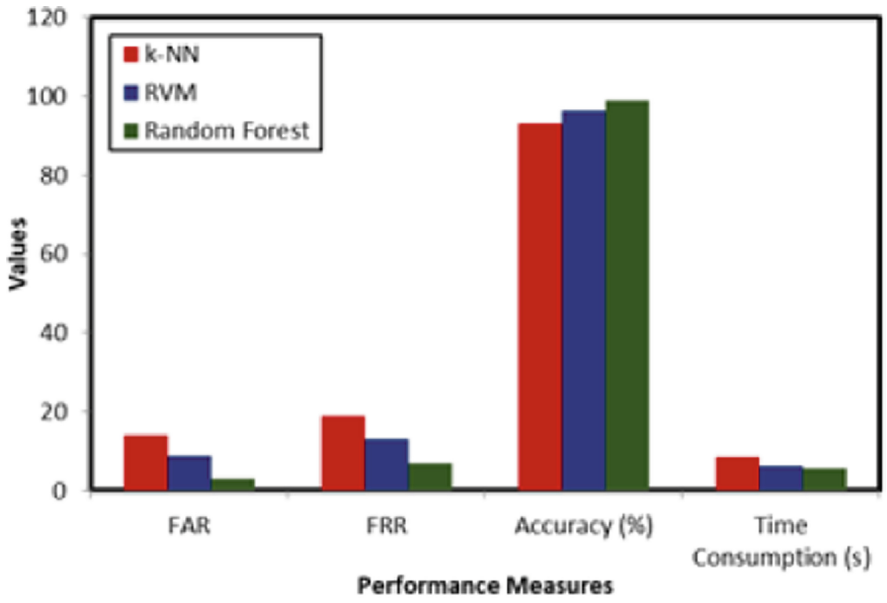

Fig. 5. Comparative analysis w.r.t classifiers

From Table 1, it is clear that the GLVP feature descriptor works well for the proposed finger vein recognition in combination with IGR and Random Forest classifier. Figure 5 depicts the performance of the classifiers.

4.2 Performance Comparison w.r.t Classifiers

The performance of the proposed approach is evaluated by changing the classifiers such as k-Nearest Neighbour (k-NN) [17], Relevance Vector Machine (RVM) [18] and random forest classifier. The experimental results attained by this work are as follows.

The performance of random forest classifier is better than the k-NN and RVM, which shows the least time consumption, FAR and FRR rates, while showing great accuracy rates.

5 Conclusion

This article presents a biometrically authenticated boot loading system from USB drive by exploiting the fingerprint and finger vein of individuals. This work is segregated into OS installation in USB drive, biometric security provisioning and secure boot loading. Initially, the OS is installed in the USB drive and the biometric security is provided to the USB with the help of fingerprint and finger vein. Only the authorized users can initiate the boot from USB. Since the fingerprint processing related operations are done inside the USB system without any role for the host system, the need to transfer fingerprint images between host and USB is removed, thus securing the pattern. Now the secure and portable system is ready to be used in any unknown machine without any concern for security and without touching the computer configuration.

The proposed approach has integrated vein and fingerprint as a multimodal scheme to from a unit authorization system. In future the proposed work will be tested with other databases. The FR error is the refusal of for legitimate users trying to access the system. A FA error occurs when the system accepts access to unintended individual. These two sorts of errors are inversely proportional i.e., lower the FAR the occurrences of FFR increases and cloud be controlled by a confidence threshold. To elevate the system security, the threshold cloud be increased, which reduces FA errors and elevate FR errors.

References

1. Noyes, D., Liu, H., & Fortier, P. (2016, May). Security analysis and improvement of USB technology. In 2016 IEEE Symposium on Technologies for Homeland Security (HST) (pp. 1–3). IEEE.
2. Tischer, M., Durumeric, Z., Bursztein, E., & Bailey, M. (2017). The danger of USB drives. IEEE Security & Privacy, **15**(2), 62–69.
3. Neugschwandtner, M., Beitler, A., & Kurmus, A. (2016, April). A transparent defense against USB eavesdropping attacks. In Proceedings of the 9th European Workshop on System Security (p. 6). ACM.
4. Tian, D. J., Bates, A., & Butler, K. (2015, December). Defending against malicious USB firmware with GoodUSB. In Proceedings of the 31st Annual Computer Security Applications Conference (pp. 261–270). ACM.
5. Kang, M., & Saiedian, H. (2017). USBWall: A novel security mechanism to protect against maliciously reprogrammed USB devices. Information Security Journal: A Global Perspective, **26**(4), 166–185.
6. Pham, D. V., Syed, A., & Halgamuge, M. N. (2011). Universal serial bus based software attacks and protection solutions. digital investigation, **7**(3-4), 172–184.
7. 8. Kushwaha A. S., (2013), A trusted bootstrapping scheme using USB key based on UEFI, International Journal of Computer and Communication Engineering, **2**(5), 543.
8. Alycia Sebastian, Dr. K. Siva Sankar, "A Secure Boot Loader System for Loading an Operating System with Fingerprint Authentication", International Journal of Innovative Technology and Exploring Engineering, **8**(10), 2368–2374, 2019.
9. Alycia Sebastian, Dr. K. Siva Sankar, "Design of a Dynamic Boot Loader for Loading an Operating System", Journal of Computer Science, **15**(1), 190–196, 2019.
10. Alycia Sebastian, Dr. K. Siva Sankar," Design of a Boot Loader for Operating System", Australian Journal of Basic and Applied Sciences, **9**(2), 368–374, 2015.

11. Burrows, M., Abadi, M., & Needham, R. M. (1989). A logic of authentication. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, **426**(1871), 233–271.
12. https://www.kickstarter.com/projects/eyedisk/eyedisk-unhackable-usb-flash-drive
13. Gao, W., Zhang, X., Yang, L., & Liu, H. (2010, July). An improved Sobel edge detection. In 2010 3rd International Conference on Computer Science and Information Technology (Vol. 5, pp. 67–71). IEEE.
14. 15. Kumar, A., & Zhou, Y. (2011). Human identification using finger images. IEEE Transactions on image processing, **21**(4), 2228–2244.
15. 16. Bianconi, F., & Fernández, A. (2007). Evaluation of the effects of Gabor filter parameters on texture classification. Pattern recognition, **40**(12), 3325–3335.
16. 17. Fan, K. C., & Hung, T. Y. (2014). A novel local pattern descriptor—Local vector pattern in high-order derivative space for face recognition. IEEE transactions on image processing, **23**(7), 2877–2891.
17. 18. Cunningham, P., & Delany, S. J. (2007). k-Nearest neighbour classifiers. Multiple Classifier Systems, **34**(8), 1–17.
18. Tipping, M. E. (2000). The relevance vector machine. In Advances in neural information processing systems (pp. 652–658).