Robust Device Authentication Using Non-Standard Classification Features

Supriya Yadav, Pooja R.Khanna
School of Engineering and Digital Arts
University of Kent
Canterbury, UK
sy227@kent.ac.uk
pk327@kent.ac.uk

Gareth Howells
School of Computing
University of Kent
Canterbury, UK
W.G.J.Howells@kent.ac.uk

Abstract— This paper investigates the use of novel hardware features derived from the physical and behavioral characteristics of electronic devices to identify such devices uniquely. Importantly, the features examined exhibit non-standard and multimodal distributions which present a significant challenge to model and characterize. Specifically, the potency of four data classification methods is compared whilst employing such characteristics, proposed model Multivariate Gaussian Distribution (MVGD -address multimodality), Logistic Regression (LogR), Linear Discriminant Analysis (LDA), Support Vector Machine (SVM). Performance is measured based on its accuracy, precision, recall and f measure. The experimental results reveal that by addressing multimodal features with proposed model Multivariate Gaussian Distribution classifier, the overall performance is better than the other classifiers.

Keywords—Security, ICMetric, Authentication, Classifiers, Key generation, Multidimensional space.

I. INTRODUCTION

To protect passwords, encryption keys and various secrets, applications rely heavily on underlying native security platform offered by OS, device and microprocessor providers. A majority of these main stream providers have been successfully attacked multiple times for example Pegasus attack against WhatsApp encryption keys (iOS & A droid) [12], Jeff Bezos iPhone hack, Meltdown & Spectre [13] (Intel, ARM, AMD, Linux/Windows etc.) and most recently SGAxe and Crosstalk [16] targeting Intel H/W. To avoid total dependence on native security features, we can adopt an effective 'layered' security approach. In this research paper, we explore ICMetrics as an additional layer.

Identity fraud can wreak havoc on societies and economies and this crime is often committed to facilitate other crimes such as credit card or, money laundering, mail, bank, and wire fraud etc [11]. These frauds affect not only individual citizens and nation's economy but it is a national security threat as well.

A significant amount of this fraud can be tackled effectively if there is a robust way to link users' physical identities to their online identities and the credentials strongly bound to their devices. Integrated Circuit Metrics (ICMetrics) can play a crucial role here. ICMetrics is a software client which reads various dynamic and static (hardware/software) feature values of a device and it generates a unique identifier for the device [8]. This unique identifier is used to generate the key pair of which the private key is not stored permanently on the device nor at database. Every time a crypto-operation is required, the ICMetric client reads these feature values and reconstructs the private key [5]. If the ICMetric client is skimmed, then on a rogue device, the feature values will differ from what ICMetric client expects, which will result in failed crypto operation. This technique eliminates 'offline brute force' attack [1,4]. An ICMetrics system generally consists of two phases, the calibration phase and the operation phase. The key steps involved in the calibration phase are as follows:

1. Measurement of feature data for each sample device.
2. Generation of feature distributions for each feature, illustrating the frequency of each discrete value for each device.
3. Normalisation of the feature distributions, generating normalisation maps for each feature.

Then comes operation phase which generates key for a given device. The operation phase contains the following steps:

1. Measurement of feature values for the device.
2. Application of the normalisation maps to feature values in order generate values suitable for key generation.
3. Application of the key generation algorithm.

The focus of this paper is to propose a novel model for device classification and compare the performance of classifiers in terms of accuracy, precision, recall and F measure on features data extracted from computing devices.

The rest of the paper is organized as follows: -
Section II Introduces device characteristic feature extraction, Section III Describes, ICMetric generation methodology, Section IV Summaries experiments and results and Section V Concludes the paper.

978-1-7281-5546-3/19/$31.00 ©2019 IEEE

II. Feature Extraction

A. Calibration Phase

Calibration Phase is useful to extract suitable features in pre-production with the aim of giving sufficient correlation when combined. By combining device features appropriately, an ICMetrics system can form a unique identifier. For this work, the unique identifier is a unique random number, with high entropy (lack of predictability) to provide the cornerstone value, for example if device features fail in unexpected ways during the operation phases, this will give a reference point from which the acquired features can map and combine to provide an applicable key for authorization.

Calibration is carried out once per application domain. The suitability of device features (in terms on high entropy) depends on the nature of the device. In most cases, it will include surveying a device for a set number or period and gathering stable values. In other cases, it may be variable features that are likely to change over time. This leads into static and dynamic variables [3].

Calibration phase contains four parts i.e., (a) Data Collection, (b) Feature Selection, (c) Feature Modelling, and (d) Feature Analysis that describes the processes sequentially undertaken prior to build a model.

The device characterizations employed by the system are known generically as features. Features are a major part of the ICMetric system, and the features utilized straightforwardly influence the strength of the security provided. The data collected from the 'devices' described here are the Apple Laptops, namely MacBook Pro and MacBook Air. With weak features i.e., features which do not change at all with the functionality and restricts the ICMetric system in how much security it can offer means a feature value is ultimately used to identify a device, so the more discriminative it is, the better when evaluating the security of an ICMetric system [7]. The values of features are dependent upon the usage of the machine. Ideal candidate features can provide the basis for a secure system that can guarantee an increase to the trust associated with existing security protocol. The analysis and mapping techniques allow the system to incorporate features whose value can change while still being able to transform these dynamic values into a unique and static value that can be used to distinguish a device. To facilitate this, features that exhibit low intra-sample variance and high inter-sample variance are selected as a priority for the mapping process. Because the values of features employed in the ICMetric system can change, the feature behavior and the influences on that feature value need to be understood before an ICMetric value can be reproduced consistently.

1) Data Collection

The data is collected using code written in Python and in a monitored natural environment which gives an insight into the behavior of the features during the analysis. The features extracted is uniquely affected by each user's machine usage (as each user uses their machines differently). This is to allow conclusions to be drawn between the presence of background processes for a system resource and the influence they can have on the various candidate features being analyzed. The features were initially narrowed down through a variety of techniques, including the analysis of their variance and their correlations with other features to find any stable correlations that were distinctive to any set of devices. This will lead to greater understanding of feature correlations per device in order to exploit their internal relationship. It is not just framework measures that might actually influence low-level hardware feature values. Client controlled cycles could likewise adjust the distribution of a feature. To help with this problem, the situation of the device is observed and recorded when data are read for analysis. The selected features were subsequently divided into sets in order to increase operational robustness via the employment of Shamir's secret sharing to allow controlled potential partial failure of the system whilst still retaining some security verification.

These feature sets offer more natural obfuscation and are more reliable than individual features and generate stronger base for applying ICMetric system [2].

B. Feature Selection

There are three categories of features that are collected i.e., CPU related features, speed of hard disk related features and memory-based features. Out of collected 30 features, from which we create 3 feature sets. Feature sets of a device can be sensibly considered into individual sets. Each set contains features, which share alike qualities or are affected by the same changes of a device.

These 3 feature sets consist of the eight, six and three features respectively in each set. Each feature set contains: CPU related features, speed of hard disk related features and memory-based features respectively, to recognize low-level behavior of the features.

Table 1 Lists New Dynamic Features selected to build an ICMetric system:

Sr. No.	Feature Name	Feature set Number
F1	Maximum speed for copy function	1
F2	Maximum speed for scale function	1
F3	Maximum speed for add function	1
F4	Maximum speed for triad function	1
F5	Average duration for copy function	1
F6	Average duration for scale function	1
F7	Average duration for add function	1
F8	Average duration for triad function	1
F9	Sequential write(block)%CPU	2
F10	Sequential write(block)MB/sec	2
F11	Sequential write(rewrite)%CPU	2
F12	Sequential write(rewrite) MB/sec	2
F13	Sequential read (perchar)%CPU	2
F14	Sequential read (perchar)MB/sec	2
F15	Duration for add function	3
F16	Quickest duration for add function	3
F17	Longest duration for add function	3

The following are the properties (related to raw features) that we have explored to identify the devices uniquely:

1. Correlated features provide higher stability than individual features as they offer a predictability of range amongst them, this means that there is less intra-sample variance. Thus, increasing reproducibility of the generated key These

978-1-7281-5546-3/19/$31.00 ©2019 IEEE

correlated raw feature sets as they contribute to build a robust system.

2. The lower intra-sample (samples of the same device) variance is needed means the more feature value can vary, the harder the value is to map and the less stable the value is when contributing to key generation.
3. Higher inter – sample (samples between two or more devices) variance contributes to the larger entropy of the system

The high inter-sample variance and low intra-sample variance are examined to observe the potential overlap of the data between two or more devices [7].

C. Feature Modelling

Unlike static features which can be used to generate a stable unique identifier directly, dynamic features require statistical modelling to be used for unique identifier generation; owing to the fact that they are continuously changing, statistical features such as the mean and variance of a set of raw data are required in order to generate a stable unique identifier, as these values are unlikely to change much with time. Since different approaches may be required for different feature sets some may be normally distributed, some may conform to a multimodal distribution etc. This section presents techniques that used to model dynamic features.

1) Unique Identifier Generation

The primary goal of an ICMetrics system is to generate a unique identifier for each device, which is derived from various device characteristics. This unique identifier can then be used to generate encryption keys, authenticate the device, and detect changes in device operation. This unique identifier should have high intra-sample stability (on the same device) but low inter-sample stability (between different devices). In other words, a given device should always generate the same unique identifier, which should be unique to that device [2].

In order to increase the entropy, feature values from multiple features are combined in order to produce a unique identifier with sufficient inter-device entropy to be used for key generation [8], and that is stable enough that it can be reliably reproduced. Feature values can be generated from both static and dynamic features, but the process of doing so varies for each type. For dynamic features, it is likely that each time the feature is sampled, the feature will hold a different value. Instead, it is important to take numerous estimations of the feature, quantize the deliberate values into discrete values, and produce a frequency distribution for that feature. One possible approach to extract a feature value from a feature distribution is to map every value to a single value that is representative of the distribution, for example the median of the set. This number would then be the feature value for that set. Since we have 17 unique features as of now, the entropy is 217 and as we research new stable features, it is likely to go up.

2) Normalization

In the calibration stage, features which are described in Section 2.2 have been utilized. At that point, the data is sent to quantize and normalize process.

If the data measured from device is non-normally distributed, it may be necessary to normalize the data so that it can be used for key generation. One approach that can be used to achieve this is to map the values from the raw distribution to a set of values in a normal distribution [9]. Finally, a multidimensional normalization map is produced dependent on normalized data. In the operation phase, a measured data is mapped to multidimensional normalization map to form a unique identifier. At last, the unique identifier is forward to produce encryption key [14,15].

D. Feature Analysis

To analyze the data, we generate a probability distribution graph for each feature to understand how the data is distributed in multidimensional space. The importance of visualizing the data in multidimensional space, helps differentiate between the overlapping data from different devices. This will infer the data to be unimodal, bimodal, or multi-modal in nature. Addressing this multimodality will increase the probability of the devices being recognized correctly. The proposed algorithm is presented as a flow chart in Figure 1 where component operations undertaken step by step to Identify these Computing Devices uniquely and build a Robust Classifier.

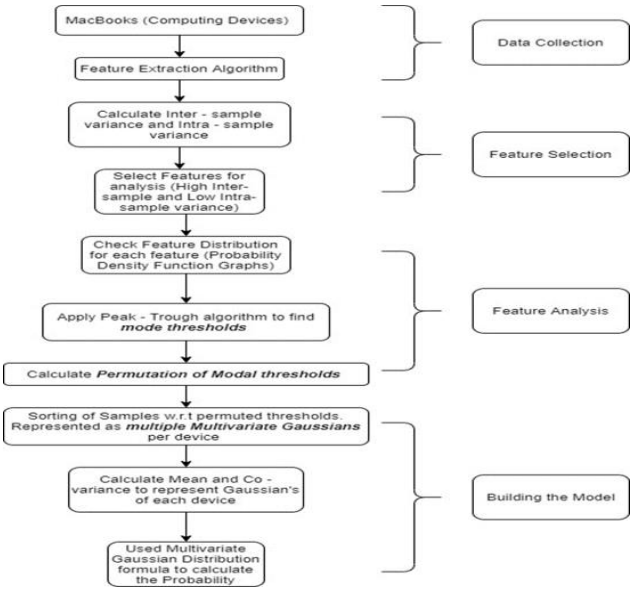

Figure 1 Shows the calibration phase of the proposed system

1) Multimodal Distributions

After feature analysis, we concluded that multimodal set of features do not generate a unique identifier. To address this challenge, we divide the distributions into a series of components where each component is approximately normal and where each mode on the original distribution become the mode of its own normal distribution. Simple approach to this problem is to apply a peak-trough detection algorithm to the histogram of each feature where the troughs split the multimodal distribution into separate normal distributions (converting this to unimodal) with the peaks forming the modes, to decrease the overlapping of data amongst devices.

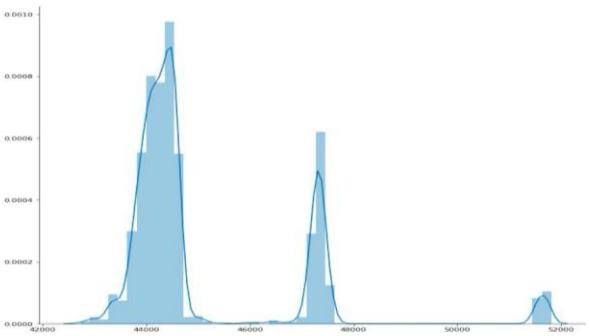

Figure 2 Shows Probability Density Function graph as an example of multimodal distribution of a feature where x axis represent feature values and y axis represent frequency.

The Peak – trough algorithms take in the histogram data and divides the modes based on the troughs of a probability distribution graph [17]. This is used to create modes to associate samples to their respective permutations. This in turn will show the relationship between the features for each modal combination. Hence, examining these features in relation with each other creates a unique device print where these combinations are generated.

When the following set of sequence of instructions is applied to training data in our experiment, some features from all devices have normal or multimodal distribution. To identify the data belonging to a particular device, the samples are taken from each device, then mean and covariance of the modes within these distributions are calculated. Each mode has its own 'identity' i.e., mean and covariance per distribution. Each device having multiple modes will have more than one distribution to represent it. After this we check which mode each of the samples fall into and compute the likelihood of the sample and rehash similar cycle for different modes. Similarly, a sample from different device is taken and if it falls into one of the modes of first device, the probability is calculated as a deciding factor. Lastly, the higher the probability of the test data against any device, more likely it is that the data belongs to that device. We repeat the same process for' n' devices. By separating these modes, we increase the accuracy percentage of the classifier.

E. Operation Phase(Key Generation)

The operation phase starts each time an encryption key is required. For this, all features in the three-feature set (as described in section 2.2) are dynamic in nature which requires statistical/mathematical modelling for unique identifier. In other words, a given device should always have unique identifier, which is the primary goal of the ICMetrics system [11,12].

There is a challenge with ICMetrics. The unique identifier generated by the device used to authenticate, is formed of several device characteristics, having just one characteristic change significantly, may change the unique identifier although the variation may still be consistent with the operation of the device. Subsequently, the device will fail to authenticate because basic approaches to combine feature values, like simple concatenation, don't allow for device characteristics to change.

One possible solution to this problem is to implement a secret sharing algorithm to combine feature values, which allows the unique identifier to be recovered even if a limited number of the device characteristics have failed.

1) Secret Sharing Scheme

In the case of Shamir's Secret Sharing algorithm [10], this is done by defining a polynomial where the y-axis intercept defines the unique identifier, and upon which all of the devices feature values (at the time of calibration) lie. Since a polynomial can be defined if a given number of points are known (i.e., a straight line with 2 points, a parabola with 3 points, a cubic polynomial with 4, etc.), the y-axis intercept and therefore the unique identifier can be recovered even if a limited number of the characteristics fail. For example, a parabola with 5 total points would allow up to 2 points to be invalid and the unique identifier can still be calculated correctly using the other 3 valid points available.

Next step is how we use Secret sharing concept in ICMetric key generation process. What we commonly do here is generate unique identifier to pass in as the X values. We then calculate the associated Y values to create the points needed to reconstruct the unique identifier, or device identity. When we need to reconstruct the secret, we can get the Y values from where we stored them & read ICMetric values to get the X values. Once we have these X & Y pairs, we can reconstruct the secret using interpolation. In this way, we can only re-construct the secret correctly when enough ICMetric values are valid.

So, we can split a secret into several shares, with a threshold needed to be met before the secret can be reconstructed. Generating a secret using some form of cryptographically secure RNG then using ICMetrics to represent the points on the polynomial allows the secret to be reconstructed with valid ICMetrics, and also allow the key to be revoked if it gets compromised. It also allows us to set a level of tolerance in the system with difficult-to-map features so the reliability of the ICMetric system is acceptable.

The advantage of this process is that the ICMetric is not stored on the system and the only values that are stored are one half of the co-ordinates that are necessary to generate the polynomial that produces the ICMetric. The halves that are stored on the system cannot be used to find out the polynomial that was used to generate them. Interpolation cannot be employed without the associated x value for each stored Y value, which means an attacker cannot derive the device identifier (unique identifier) with the stored data and the attacker has no way of knowing where on the X axis each point sits. Additionally, a new ICMetric can be generated any time the system needs to be changed or reset by repeating the process of taking a new arbitrary basis value and passing in feature values to generate new Y values similar to replace existing Y values, or we can use an offset to update Y values dynamically.

III. EXPERIMENTAL METHODOLOGY

This proposed method has the ability to generate unique identifier. In order to evaluate the model as classifier, we define correct classification of unique identifier of the device with our model multivariate Gaussian distribution. In this proposed methodology we have used other three standard classifiers for benchmarking which are namely Logistic Regression (LR), Linear Discriminant Analysis and Support Vector Machines (SVM). These classifiers are used to compare the prediction. We applied the above mention classifiers on different device datasets to correctly classify the test data based on three feature sets. The performance of these classifiers is evaluating on the bases of accuracy, precision, recall and F measure. These classifiers are carried out in Python language. Python is an incredible mediator language and a solid stage for research. The exploratory outcome depicts which classifier is best between them.

A. Classifiers

This work represents a comparison amongst four classification techniques evaluating which of these techniques is best suited to identify and classify the devices based on the data collected. In this section, we introduce these classification techniques briefly.

978-1-7281-5546-3/19/$31.00 ©2019 IEEE

1) ProposedMultivariate Gaussian Distribution(MVGD)

Each Gaussian/normal distribution is modelled by the mean and variance derived from sample data extracted from the device. The Multivariate Gaussian defines the joint probability distribution which are mutually independent normal variables. Hence, a need to examine the collective effect of these variables. Hence a probability of a vector belonging to a particular Multivariate Gaussian is calculated, where each of the MVGD is defined by the mean vector and covariance matrix of the distribution.

By taking multimodality into consideration, we model the data of each device as multiple multivariate Gaussian distributions. As we know in Gaussian Mixture Models (GMM) [19], the data is represented as 'n' mixture models; Similarly represented by our classifier as 'n' multivariate Gaussians per distribution. Assuming each sample belongs to one of these multivariate Gaussian.

The d-dimensional vector x is multivariate Gaussian in the event that it has a likelihood thickness capacity of the accompanying structure:

$$p(x;\mu,\sum) = \frac{1}{(2\pi)^{d/2}|\sum|^{1/2}} exp\left(\frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu)\right) \quad (1)$$

The pdf is parameterized by the mean vector μ and the covariance matrix Σ.

The mean vector μ is the assumption for x:

$$\mu = E[x] \quad (2)$$

The covariance lattice Σ is the assumption for the deviation of x from the mean:

$$\Sigma = E[(x-\mu)(x-\mu)^T] \quad (3)$$

2) Linear Regression(LR)

The Logistic Regression is a linear classification technique conducted for a predictive analysis. When the dependent/target variable is dichotomous [14]. The classification presents a binomial outcome i.e., representing the occurrence of an event or not with values 1 and 0 respectively, based on data from input variables. This can also be used as a multinomial regression which can deal with categorical classification like target variable 1 to 8 for each of the 8 devices [15].

3) Linear Discriminant Analysis(LDA)

The LDA projects the data in higher dimension onto a lower dimension space (reducing dimensions). By combining the variables in a linear or quadratic manner that gives Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) [14]. These variables are combined in a way that the differences (i.e., the separation) between the classes are maximised. The LDA technique is used when the covariance of each of the unique classes is the same and the predictors are distributed normally.

4) Support Vector Machine(SVM))

The SVM model has the capabilities to handle both i.e., regression and classification problems. Here the data is plotted and viewed in n-dimensional space, where n depicts number of features. This is a non-linear classification technique which can separate the data from different classes via a decision plane. Hence the data which seems linearly inseparable, are subjected to intricate mathematical functions called kernel which effectively separates the data belonging to their respective classes [15]. The complexity of the model ensures the higher accuracy and presents fewer possibilities of over-fitting.

IV. EXPERIMENTAL RESULTS

This segment presents a discussion of the obtained experimental results of the proposed model MVGD. The experiments are conducted on the features which are explained in section 2.2, this data is collected from the hardware features (Memory, CPU, Hard disk) from MacBook Air and MacBook Pro. This data after analysis, gives us a unique identifier. We used eight devices with updated software. For this work, we used data collected from the MacBook Air and Pro, Python Code and Microsoft Excel used for data analysis. For this experiment we are using these devices, where each device contains thousand samples for our analysis, Cross Validation method with fold value equal to 10 has been used for training and testing phases. Consequently, all of the records which exist in dataset will affect the training and testing of the classifiers.

Table 2, 3, 4 show the comparison of our proposed model MVGD and other standard classifier LR, LDA and SVM and evaluate the performance of the proposed model on the bases of accuracy, precision, recall and F Measure defined below. Where TP- True Positive (A true positive is a result where the model effectively predicts the positive class) – if we can prove that a unique identifier belongs to a specific device, then its TP TN- True Negative(A true Negative is a result where the model effectively predicts the negative class) – if we can prove that a unique identifier does not belong to a specific device, then its TN FP- False Positive (This wrongly identifies the data belonging to a particular class) – if unique identifier identifies a device incorrectly, then its FP and FN-False Negative (This wrongly indicates the absence of the data belonging particular class)- if unique identifier incorrectly concludes that it's not the specific device, however, in reality it is the device in question, then its FP [18].

Classification Rate or Accuracy is given by the relation

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Recall: Recall can be defined as the ability of the classifier to find all positive instances. It is defined as the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

Precision**:** Precision can be defined as the ability of the classifier not to label as positive a sample that is negative. It is defined as the ratio of true positives to the sum of true positives and false positives.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (6)$$

F-measure**:** F-measure can be defined as the Harmonic Mean of precision and recall. The F-measure corresponding to every class will tell you the accuracy of the classifier in classifying the data points in that particular class compared to all other classes.

$$\text{F-measure} = = \frac{2*Recall*Precision}{Recall+Precision} \quad (7)$$

In all classifiers for first feature set which includes eight features related to speed of hard disk to copy, add, scale and triad function MVGD perform better, its accuracy is 91.5% after that SVM perform better it holds 90% accuracy.

TABLE II. For Feature Set 1 classification performance of proposed model with standard classifiers using Training-Testing in 10-fold cross-validation setup.

Classifier	Accuracy	Precision	Recall	F Measure
MVGD	91.5%	74.2%	73.1%	72.5%
LDA	87%	70.7%	69.7%	68.7%
LA	87%	69.2%	69.7%	68.9%
SVM	90%	73.4%	73.5%	72.9%

For Second feature set which includes 6 Features related to Hard disk like CPU usage when writing to disk and memory-related features like time taken to read memory MVGD perform better its accuracy is 92% after that SVM perform better it holds 90.5% accuracy.

TABLE III. For Feature Set 2 classification performance of proposed model with standard classifiers using Training-Testing in 10-fold cross-validation setup.

Classifier	Accuracy	Precision	Recall	F Measure
MVGD	92%	74.1%	73.6%	73.4%
LDA	91.2%	74%	73%	72.5%
LA	91%	74.1%	73.4%	73%
SVM	90.5%	77.2%	77.2%	77.1%

For Third feature set which includes 3 Features related to CPU-related values like the performance of floating-point arithmetic MVGD perform better it holds 80.1% accuracy after that SVM perform better its accuracy is 67.9%.

TABLE IV. For Feature Set 3 classification performance of proposed model with standard classifiers using Training-Testing in 10-fold cross-validation setup.

Classifier	Accuracy	Precision	Recall	F Measure
MVGD	80.1%	67.9%	64.6%	62.1%
LDA	57.8%	44.2%	47%	42.3%
LA	57.1%	52.6%	46.4%	44.1%
SVM	67.9%	59%	55.1%	50.3%

From the experiment results, we observe that proposed model Multivariate Gaussian distribution perform better as compared to other three standard classifiers in the prediction of identifying devices uniquely.

After applying secret sharing (explained in section 2.5.1), the result for ICMetric Key generation for Multivariate Gaussian distribution classifier were 94%, 95% and 84% for first, second and third feature set respectively. This proves that our results improved statistically over the previous results.

V. Conclusion

In this paper, we explored the employed of hardware characteristic features to identify electronic devices performed the comparison analysis of classifiers for the prediction of identifying the device. The device identification technique is compared to four alternative classifiers. Experimental result show that different classifiers behave differently on the same dataset. From the analysis, we observed that proposed model MVGD performed better than all others for device identification. And our Shamir's Secret Sharing results based on ICMetric key generation for proposed model MVGD are quite promising. Overall, this paper outlines the method of analysis and mathematical implementation using proposed model multivariate Gaussian distribution.

References

[1] R. Tahir and K. McDonald-Maier, "Improving Resilience against Node Capture Attacks in Wireless Sensor Networks using ICMetrics," in Emerging Security Technologies (EST), 2012 Third International Conference on, 2012, pp. 127–130.

[2] E. Papoutsis, G. Howells, a. Hopkins, and K. McDonald-Maier, "Key Generation for Secure Inter-Satellite Communication," Second NASA/ESA Conf. Adapt. Hardw. Syst. (AHS 2007), pp. 671–681, Aug. 2007.

[3] B. Ye, G. Howells, and M. Haciosman, "Investigation of Properties of ICMetrics in Cloud," in Emerging Security Technologies (EST), 2013 Fourth International Conference on, 2013, pp. 107–108.

[4] R. Tahir, H. Hu, D. Gu, K. McDonald-Maier, and G. Howells"Resilience against brute force and rainbow table attacks using strong ICMetrics session key pairs," in Communications, Signal Processing, and their Applications (ICCSPA), 2013 1st International Conference on, 2013, pp. 1–6.

[5] A.Hopkins, K. Mcdonald-Maier, and G. Howells, "Device to generate a machine specific identification key." Google Patents, 2013.

[6] R. Tahir, H. Hu, D. Gu, K. McDonald-Maier, and G. Howells, "A scheme for the generation of strong cryptographic key pairs based on ICMetrics," in Internet Technology And Secured Transactions, 2012 International Conference For, 2012, pp. 168–174.

[7] Y. Kovalchuk, H. Hu, D. Gu, K. McDonald-Maier, D. Newman, S. Kelly, and G. Howells, "Investigation of Properties of ICMetrics Features," in Emerging Security Technologies (EST), 2012 Third International Conference on, 2012, pp. 115–120.

[8] Y. Kovalchuk, K. McDonald-Maier, and G. Howells, "Overview of ICMetrics Technology-Security Infrastructure for Autonomous and Intelligent Healthcare System.," Int. J. U-& E-Service, Sci. Technol., vol. 4, no. 3, 2011.

[9] G. Howells, E. Papoutsis, A. Hopkins, and K. McDonald-Maier,"Normalizing Discrete Circuit Features with Statistically Independent values for incorporation with in a highly Secure Encryption System," in Adaptive Hardware and Systems, 2007. AHS 2007. Second NASA/ESA Conference on, 2007, pp. 97–102.

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

[11] "Global Cost of Cybercrime Exceeded $600 Billion in 2017, Report Estimates," Security Intelligence.

[12] "Pegasus (spyware)," Wikipedia, Mar. 26, 2021.

[13] "Meltdown and Spectre," Meltdownattack.com, 2013.

[14] A. Singh, N. Thakur and A. Sharma, "A review of supervised machine learning algorithms," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2016, pp. 1310-1315.

[15] S. Ray, "A Quick Review of Machine Learning Algorithms," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 35-39.

[16] D. Goodin, "Plundering of crypto keys from ultrasecure SGX sends Intel scrambling again," Ars Technica, Jun. 09, 2020.

[17] K. Harmer, G. Howells, W. Sheng, M. Fairhurst and F. Deravi,"A Peak-Trough Detection Algorithm Based on Momentum," 2008 Congress on Image and Signal Processing, Sanya, China, 2008, pp. 454-458.

[18] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detectio system through feature selection analysis and building hybrid efficient model," Journal of Computational Science, vol. 25, pp. 152–160, Mar. 2018.

[19] D. Reynolds, "Gaussian Mixture Models," in Encyclopedia of Biometrics, 2015, pp. 827–832.

978-1-7281-5546-3/19/$31.00 ©2019 IEEE