2019-06

User Profiling Based on Application-Level Using Network Metadata

Shaman, F

http://hdl.handle.net/10026.1/15108

10.1109/isdfs.2019.8757503
2019 7th International Symposium on Digital Forensics and Security (ISDFS)
IEEE

User Profiling Based on Application-Level Using Network Metadata

Faisal Shaman[1,2]; Bogdan Ghita[1]; Nathan Clarke[1] and Abdulrahman Alruban[1]
[1]*Centre for Security, Communications and Network Research*
University of Plymouth, Plymouth, United Kingdom
[2]*Faculty of Computers and Information Technology, University of Tabuk, Saudi Arabia*
{faisal.shaman, bogdan.ghita, nathan.clarke, abdulrahman.alruban}@plymouth.ac.uk

Abstract— **There is an increasing interest to identify users and behaviour profiling from network traffic metadata for traffic engineering and security monitoring. Network security administrators and internet service providers need to create the user behaviour traffic profile to make an informed decision about policing, traffic management, and investigate the different network security perspectives. Additionally, the analysis of network traffic metadata and extraction of feature sets to understand trends in application usage can be significant in terms of identifying and profiling the user by representing the user's activity. However, user identification and behaviour profiling in real-time network management remains a challenge, as the behaviour and underline interaction of network applications are permanently changing. In parallel, user behaviour is also changing and adapting, as the online interaction environment changes. Also, the challenge is how to adequately describe the user activity among generic network traffic in terms of identifying the user and his changing behaviour over time. In this paper, we propose a novel mechanism for user identification and behaviour profiling and analysing individual usage per application. The research considered the application-level flow sessions identified based on Domain Name System filtering criteria and timing resolution bins (24-hour timing bins) leading to an extended set of features. Validation of the module was conducted by collecting NetFlow records for a 60 days from 23 users. A gradient boosting supervised machine learning algorithm was leveraged for modelling user identification based upon the selected features. The proposed method yields an accuracy for identifying a user based on the proposed features up to 74%**

Keywords- user profile, user behavioural, user identification, network traffic analysis, supervised learning, network security

I. INTRODUCTION

The number of internet users has reached more than five billion across the world, and it is growing continuously [1]. A recent report published by Cisco Inc. in 2016 presented that the traffic data generated was at a level of 7 Exabytes per month, due to reach an expected monthly data volume of 49 Exabyte in 2021 [2]. Due to the massive usage of computer systems and applications, as well as their increased complexity, user identification and behaviour profiling from generic network traffic have become critical parts of network and traffic management [3]. Primarily for network administrators and security investigators to identify security breaches and enforce the organisation policy as well as provide more intelligent routing decisions for the traffic transiting the infrastructure [4]. User profiling based on the features extracted (source to destination packet size, inter-arrival time) from network traffic metadata encourages the ISP to know the user and how this is reflected in the organisation's security and their policy. User identification and behaviour profiling are the translation of each user activity and include a network footprint of the user interaction. Understanding and identifying subjects from network traffic metadata and profiling their behaviour is a challenging task for researchers as user behaviour while having a common and constant component, also includes slight variations and even the nature of online applications interaction changes over time [5]. In addition, while users can indeed be linked through their authentication profiles with the IP addresses they have allocated, an IP-agnostic solution allows for both a reduction of cross-layer monitoring of users as well as detection of possible intrusion/misuse. This study examines user identification and behaviour profiling by analysing generic network traffic and aiming to profile and identify user behaviour based on their timing and application usage footprint instead of relying exclusively on the IP addressing information [6].

Further in this context, it is worth reminding that both traditional methods of identifying applications, using port-based techniques or Deep Packet Inspection cannot be applied anymore due to the ports randomisation or tunnelling [7] in the case of the former and encryption for the latter [8]. Recent studies [9], [10]focused upon using statistical flow analysis for user identification and behaviour profiling, by extracting features from the flow-level to be able to characterise different users with preservation of user privacy and deal with the encrypted traffic. This relies heavily on the quality of the extracted features and the efficiency of the training phase [11].

The approach proposed in this paper continues the line of research by introducing a novel flow-level set of statistical features set based on the timing of application sessions. The application sessions in turn, are derived from the flow interarrival times and reverse DNS queries. The method aims to improve the accuracy of identifying users and profiling them based on their unique behaviours. The rest of the paper is organised as follows: Section 2 describes the state-of-the-art in traffic classification and user identification and a description with existing limitations of existing approaches. Section 3 explains the proposed method and discusses the rationale for selecting different flow features for this work. Section 4 evaluates the effectiveness of the proposed method by using a supervised machine-learning algorithm with the extracted features. Finally, section 5 concludes the paper and includes possible future work.

978-1-7281-2827-6/19/$31.00 ©2019 IEEE

II. Related Work

Historically, the field of user identification and behaviour profiling from generic network traffic information includes a number of different methods and techniques. The first option, a port-based monitoring and profiling is not an option anymore because of the randomly port numbers utilised by different applications are either randomised or tunnelled (towards web-based interfaces), leading to a typical accuracy of less than 70% versus the other available methods (Deep packet inspection and statistical) [12], [13]. It has been argued that the low-accuracy associated with port-based technique can be solved using the Deep Packet Inspection (DPI), which is the most powerful technique on the traffic classification fields as the results showed that the accuracy was very high, up to 95%. However, when dealing with encrypted data, the deep packet inspection techniques can only access the header and metadata of the examined packet [14]. Therefore, this limits the amount of information that such a technique can analyse which in turn affects the identification performance.

The research community has therefore moved towards using statistical methods for instance to overcome the above limitations [9], [15]. A reasonable accuracy of up to 85% was achieved by applying statistical features-based methods such as flow inter-arrival time and packet size to identify users who generated the examined network traffic [16]. The user behaviour profile to be identified from the statistical application levels which have noisy traffic. For instance, when a user concurrently browses multiple websites, his/her behaviour would convolute multiple patterns, increasing the complexity of the user identification task when using the application level [17].

In addition, a number of recent studies [18], [19] have used behavioural profiling in identifying computer network users using DNS information and the volume of traffic, summarised by the number of connections in addition to the statistical overall traffic parameters , by collecting the daily reverse DNS queries and identifying the user sessions with an accuracy of up to 72%. However, the accuracy of user identification based on DNS is also affected by the duration of observation as investigated in [19], which has an accuracy of 73% and 90% with a duration of 65 days and seven days respectively. This is indeed counterintuitive, as the accuracy on the 65 days is lower than the accuracy of seven days. This is potentially due to the slight changes in both user behaviour and application characteristics over time, which jointly may introduce noise on the data. A variety of studies have examined user behaviour profiling from different perspectives such as identification to distinguishing users [18], [20]. The techniques are primarily utilised to identify a user by storing previous user activities to be able to decide whether the examinee user is legitimate. However, the use of the behavioural-based technique by observing the interaction of the client with network applications such as the average packet size while uploading a video on YouTube [21].

Another group of studies have been conducted to explore the possibility of applying user behavioural profiling to increase the level of security in computer networks. Indeed, the early studies in this field have employed an anomaly-based detection to determine any abnormal behaviour [22]. It can be argued that using behavioural profiling can help in differentiating users for various purposes in different performance based on the statistical features extracted from the generic network traffic and the different activities that could be provided to build an accurate user profile[6], [21] .

As a result, user behaviour profiling is an appropriate solution in associating with changing of the user behaviour and application over time in a computer network.

To sum up, each method has its strength and limitation based on different circumstances. Relying only in IP addresses or port-based approaches to tag individual may not be useful enough in analysing network traffic.

III. Proposed Method

This study focuses on extracting and analysing a flow-level feature set that allows identifying user behaviour through its network activity footprint as shown in Figure 1. A set of features is utilised to investigate the users' identification and their daily Internet usage based on a filtered applications session (as explained in subsection B.2). For training and annotating purposes, the used applications are identified based on DNS queries lookup [23]. The raw network traffic is analysed in terms of representing user's daily usage by using a combination of features based on the session, timing and flow DNS filtering. The concept of user session can be described as a group of continuous flows with characterised by a flow inter-arrival time (i.e. the time between two consecutive flows) lower than a pre-defined threshold.

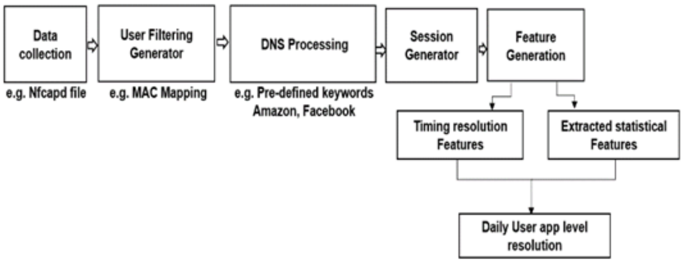

Data collection
e.g. Nfcapd file
User Filtering Generator
e.g. MAC Mapping
DNS Processing
e.g. Pre-defined keywords Amazon, Facebook
Session Generator
Feature Generation
Timing resolution Features
Extracted statistical Features
Daily User app level resolution

Fig.1: Proposed User Identification and Behaviour Profiling Methodology

The threshold value is determined by conducting a preliminary analysis that computes the inter-arrival time distribution among the flows. Using session characteristics as a discriminator is based on the fact that user behaviour differs between users (for instance browsing of Facebook varies from user to user in timing and contents). Accordingly, the session is the measure of the variability of user behaviour changing based on timing resolution bins extracted from start/end time of each application sessions (for instance, 24-hour resolution). To validate the method an experiment was carried out using a dataset captured from the University of Plymouth, the Centre for Security Communications and Network Research (CSCAN) lab for 23 users. The raw network traffic was stored as Netflow records using nfdump [24]. The stored flows were pre-processed using Python scripts to filter users based on the MAC/IP address mapping and applications based on reverse DNS queries, and to create additional statistical features. Finally, the dataset was statistically summarised to produce daily user application level records. The newly extracted features were fed into a gradient boosting machine learning algorithm to create a user profile. More details are explained in the next subsections.

A. *Data Collection*

The dataset was collected for 23 users for a period of 60 days (starting from May 8th, 2018 till July 8th, 2018) based on (ethical approval) approved by a University committee from the student network within the Centre for Security, Communications and Network Research (CSCAN) at Plymouth University, to ensure that the collected data captures most of the user's patterns such as the used applications and variability in their behaviour over time. During this period, the participants accessed the Internet through the university network and performed their normal daily routine such as browsing and downloading on the Internet. Participants were not asked to follow a protocol, and they merely use their device(s) in their typical fashion. The data was collected during their browsing of the internet and was stored in NetFlow file format, together with the MAC/IP mapping to ensure that IP changes due to DHCP allocation do not affect the accuracy. The top eleven applications were selected based on the statistical procedure, which was computed by implementing the reverse DNS queries keywords for all users to count the connections for each application and choose the top connected and used applications and websites on the lab (i.e., Amazon, Google, Instagram, Facebook, LinkedIn, Yahoo, MSN, Unknown, Stack overflow, TeamViewer, and IEEE). Therefore, these applications were added to the session generator for the applications filtering and labelling purposes. Users were filtered by using the MAC address mapping to label the data related to each use.

B. *Data pre-processing*

The collected data were pre-processed by generating the bidirectional network traffic information. The raw network traffic was generated in several steps regarding getting the most relevant flow-level features to identify the users based on the application sessions and timing resolution criteria. The next subsections explain the undertaken steps to pre-process the raw network traffic to extract desired features.

1) Acquiring Raw Network Traffic

The collected data were initially analysed by *nfdump* tool to generate the daily raw network traffic for all users in the research centre. In addition, the flow records were expanded to get specific bidirectional NetFlow data records including date start/end time, IP source, IP destination, in packets (source to destination packets), in byte (source to destination bytes), out packets (destination to source packets), out byte (destination to source bytes), bps (bits per second), pps (packets per second), and bpp (bytes per packets).

2) Media Accesses Control and IP Source Mapping

DHCP maintained the monitored network for the data collection. Therefore the Netflow collection was accompanied with IP/MAC mapping, to ensure that profiling is allocated to the correct host even if the IP addresses change. Since the MAC address of every hardware is unique, this makes the MAC addresses instead of IPs more reliable to separate the data related to each client for the training purposes only. Table 1 shows a sample of MAC addresses along with its corresponding IP to keep tracking of the IP assignment. The mapping table is used to ensure that there is no IP conflict occurs through collecting the raw traffic data.

Table 1: MAC Address and IP Source Mapping

Timestamp	**Media Access control**	**IP source**
1526029632	b86b23eb1d7f	192.168.200.170

3) Domain Name Lookup

The associated domain names are resolved for each Netflow record using a bash script [25]; this is in line with the use of reverse DNS queries were in several previous studies for tracking user behaviour and activity [17], [19]. The DNS lookup utility [23] was utilised on a bash script to initialise the application name (domain name) for each queried flow, by converting IP destination to the domain name. The converted domain name was added as a new attribute (DNS queries) to the Netflow records attributes to be analysed on the next process of this study as shown in Table 2. Therefore, the primary aim of using the DNS lookup utility in this study is to determine which flow belongs to which application that facilitates the automated application flow filtering process.

Table 2: Extracted Features after Domain Name Lookup Process

No.	**Attribute**	**Explanation**
1	date & time	Date and start /end time
2	IP dst	IP destination
3	in pkt	source to the destination number of packets transmitted
4	DNS queries	The reverse DNS quarry as 'bbc-vip016.cwwtf.bbc.co.uk.'
5	in byte	source to destination bytes
6	out pkt	destination to source number of packets transmitted
7	out byte	destination to source bytes
9	Bps	source to destination bits per second
10	Pps	source to destination packets per second
11	Bpp	source to destination bytes per packets

a) Application Flow Filtering Based on the Domain Name

The flows were filtered and separated into groups (applications set) based on pre-defined keywords related to the 11 selected popular applications (i.e., Amazon, Google, Instagram, Facebook, LinkedIn, Yahoo, MSN, Unknown, Stack overflow, TeamViewer, IEEE). Reverse DNS query results are classified as unknown if the DNS lookup utility generator could not return any value from the given IP destination address. The applications flow traffic connections that were filtered and combined in data frames (similar to matrix data object) in terms of representing the usage and automate the way of dealing with the raw network traffic for each client's duration as the data related to each user separated in the previous steps. Furthermore, the filtered data frame is used in the session generator step for the feature's analysis.

b) Session Generator

The filtered applications' data frames are then analysed and divided into sessions using a predefined flows inter-arrival time threshold, assuming that packets in any flow are relatively uniformly spread over the duration of the flow [26]. The flows inter-arrival time is denoted by τ (i.e., τ = *the start time of the second flow – the start time of the first flow*) after converting the date and time to epoch timestamp. The session parameters (Extracted Statistical and Timing Resolution Features section c) were calculated based on a flows inter-arrival time threshold based on two conditions: the flows are part of the same session when the τ is less than the threshold (i.e., 10 seconds) and the new session starts when the τ is higher than the threshold (i.e., 10 seconds). Furthermore, this procedure is applied in all filtered application data frames in order to divide each application to a set of sessions by generating features based on the session concept

C. *Features Generation Process*

The features set generation, and their discriminative strength is paramount in maximising the accuracy of the user identification. Two types of features, statistical features and session timing-based resolution features were extracted for the dataset. The session timing-based resolution features (hour session start and end) is determined and included within features sets to add another diminution to the features spacing while it could provide the user-dependent pattern.

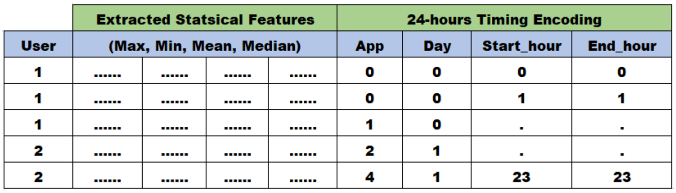
1) Extracted Statistical Features: These features were derived directly from the Netflow records (nfdump) (e.g., source to destination packets and source to destination size of packets as shown in Table 2). Besides, there were other features that were not derived by nfdump, and a calculation was applied to get the complete form of bidirectional flow records data, e.g., destination to source bits per second (bps), destination to source packets per second (pps), destination to source bytes per packets (bpp). Additional features were derived or computed from the above (e.g., Transmitted_ data rate, Received _to_ transmitted packets). Also, statistical measures were utilised on the extracted statistical flow session features no 1 to 14 (i.e., maximum, minimum, mean and median as shown in Table 3).

Table 3: Bidirectional Features Sets

NO.	Features	Explanation
class	User ID	User1, User2, User n
1	In pkt	Session Source to the destination packet
2	In byte	Session Source to destination byte
3	Out pkt	Session Destination to the source packet
4	Out byte	Session Destination to source byte
5	bps	Session source to destination bits per second
6	pps	Session source to destination packets per second
7	bpp	Session source to destination bytes per packet
8	D2s bps	Session destination to source bits per second
9	D2s pps	Session destination to source packets per second
10	D2s Bpp	Session destination to source bytes per packet
11	Transmitted _ data rate	Session transmitted data rate
12	Received _data rate	Session received data rate
13	Received_to_transmitted packets	Session Received to transmitted packets
14	Received _to transmitted _data	Session received to transmitted
15	Start time	Session start time
16	End time	Session end time
17	Number_of_ connections	Session Number of connections
18	Day_ of_ the_ week	Date encoded from (0-7)
19	Application	Application name encoded (0-10)
20	Start_ hour	Integer encoded from (0-23)
21	end_ hour	Integer encoded from (0-23)
22	Start/end hour	Start / end hour integer from (0-23) represented on (0-1) timing bins

2) Session Timing Resolution Features: The timing based features were extracted based on the start and end time of the sessions that are proposed by this study, and it includes two types of features relating to the user activity characteristics: session activity and application usage features number 15 to 22 as shown in Table 3.

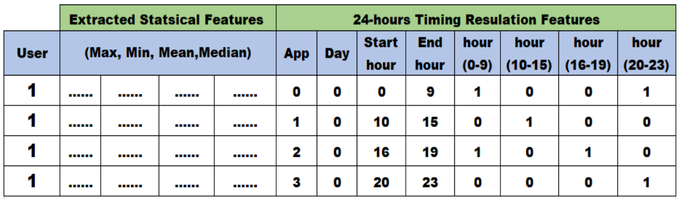
a) Daily User Session Encoding: Once the session was generated as defined on section 2 and features sets (Extracted Statistical Features, Timing Resolution Features) were extracted for each application's session based on the process explained on the previous sections. Then, the start/end time (hour) was extracted into a separated Feature as an integer that represents the hour (0-23), as shown in Figure 2. The 24-hour was encoded in terms of combining the start and end timing resolution for the whole sessions related to one application, to represent the daily usage as explained in the next section. Furthermore, the feature that represents applications was encoded into an integer based on the initialised application name to be able to operate with many machines learning which require input as numeric rather than labels, by converting each categorical value into numerical (0-10). Also, the day feature is encoded from (0-7) to represent the day of the week. The data can be described as nominal features, e.g., applications name or numerical, e.g., 0, 1 and 2. While some classification algorithms can work with nominal features, such as the Decision Tree or the Random Forest, almost all can work on numerical ones, such as the Support Vector Machine or the Multilayer Perceptron. This makes it necessary to encode the nominal to numerical features.



	Extracted Statsical Features				24-hours Timing Encoding			
User	(Max, Min, Mean, Median)				App	Day	Start_hour	End_hour
1	......	......	......	......	0	0	0	0
1	......	......	......	......	0	0	1	1
1	......	......	......	......	1	0	.	.
2	......	......	......	......	2	1	.	.
2	......	......	......	......	4	1	23	23

Fig.1: Daily user session resolution

b) Daily User app level resolution: The daily user application level time resolution features were encoded into (0, 1) timing bins as shown in Figure 3 to combine all sessions related to the applications filtered and pre-processed by representing the user daily usage behaviour. Also, to gain a higher user's daily application's activity resolution, the mean of each application's session extracted statistical features was calculated. This allowed summarising the activity of a user for one day in a single record. In addition, in this stage, the start/end hour was converted to binary encoding to represent the daily app level time resolution. For instance, if Amazon (0) used from an hour (0-9) and (20-23) per day, all these hour bins will be given 1s, and other bins will take 0s. Furthermore, if Facebook (1) is used based on an hour (10-15), this hour bins will take 1's, and other bins will take 0's.



	Extracted Statsical Features				24-hours Timing Resulation Features							
User	(Max, Min, Mean,Median)				App	Day	Start hour	End hour	hour (0-9)	hour (10-15)	hour (16-19)	hour (20-23)
1	......	......	......	......	0	0	0	9	1	0	0	1
1	......	......	......	......	1	0	10	15	0	1	0	0
1	......	......	......	......	2	0	16	19	1	0	1	0
1	......	......	......	......	3	0	20	23	0	0	0	1

Fig.2: Daily User App Level Resolution

IV. Evaluation

Gradient boosting is a useful practical supervised machine learning for different predictive tasks, and it can dependably provide more accurate results than the straight single machine learning models which are inspired by the gradient boosting framework of [27], which has been previously applied to solve classification and regression problems and more recently to train conditional random fields. The boosting supervised machine learning was utilised to build a series of small decision trees based on the collected data and each tree attempts to correct errors from the previous stage. During the last few years, many practical studies were published, which use decision trees as the base learning for gradient boosting [26], [27]. Furthermore, the algorithm can optimise any differentiable loss function by using a gradient descent approach [28]. This approach builds the trees sequentially to sum an individual tree consecutively, which provide the best solution under different conditions. In addition, the Z-score was applied to the dataset to normalise the numeric data, excluding the binary bins features for higher accuracy on the end classification model [29]. The data were split randomly into two sets; 70 % of the data were used to train the gradient boosting classifier while 30% of the data were used for testing between all user's data. The classifier performance was evaluated with different metrics derived from the four parameters: True positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). The evaluation parameters (accuracy, precision, recall, and F1 score) were calculated based on error rates, which are represented in the confusion matrix according to the following equation:

- Accuracy: it is the one that predicts the overall accuracy of the model.

$$Accuracy = \frac{\mathrm{TP+TN}}{\mathrm{TP+FP+FN+TN}} \qquad (1)$$

- Precision: it gives the fraction of the classifier prediction that is true.

$$Precision = \frac{\mathrm{TP}}{TP+FP} \qquad (2)$$

- Recall: The percentage of true results out of all results estimated by the classifier.

$$Recall = \frac{TP}{\mathrm{TP+FN}} \qquad (3)$$

- F1 score: it is a metric conducted by calculation of the Precision and Recall and more useful than accuracy in case of uneven multiclass distribution and if the FP and FN are very different [9].

$$F1\ score = \frac{\mathrm{Recall}*\mathrm{Precision}}{\mathrm{Recall+Precision}} \qquad (4)$$

A. Experimental Results:

The results as shown in Table 4 that the flows generic network traffic analysis can produce a notable result in terms of user identification and behaviour profiling. In comparison with previous studies [9], [17], in [9] the achieved accuracy was 73% by using flow network analysis approach, while our study achieved up to 74% level of accuracy. Therefore, different aspects affected the accuracy between our study and compared studies for instance (volume of traffic, the environment of collecting the flow network traffic). Table 4 shows the accuracy of all feature's sets to up to 74%. The accuracy of set 2 which represents the second 30 days of the data exceeded the accuracy of set 1 and set 3 which represent the first 30 days and all the 60 days of the data. Therefore, set 1 and set 3 data were affected by less traffic generated by the experimental lab users (due to a holiday) while set 2 traffic generated by users were normal, and these affect the volume of interactions of the examined users in those sets. The highest accuracy on set 2 improves the proposed measurement features that were affected by the periods of collecting data and user access limitation, which was observed on the volume of traffic data between all sets.

Table 4: Users' Traffic Classification Results

Set	No. users	No. days	Accuracy	Precision	Recall	F 1 score
Set 1	23	1st /30	68%	64%	63%	63%
Set 2	23	2nd /30	74%	75%	73%	73%
Set 3	23	60	72%	67%	65%	65%

The classification comparison was implemented by the gradient boosting using the set 2 features as shown in Table 5. The comparison indicates the extracted statistical features and session timing resolution features by employing them to the classifier separately. The session timing resolution features indicated the highest usage score of up to 65% compared to the extracted statistical features which were up to 61%. The session timing resolution attributes were scored the highest usage among all users. Also, the set 2 features were applied to random forest feature importance, which indicated a good performance between all features to identify users.

Table 5: Classification Performance for Each Feature type

Feature type	Accuracy	Precision	Recall	F1 score
Timing resolution features	65%	62%	60%	60%
Extracted statistical features	61%	59%	53%	55%
Both	74%	75%	73%	73%

Therefore, the top 10 features are represented in Figure 4, in which the first top 4 features (app_encoded, end_hour, start_ hour, number_of_connections) were scored the highest usage between extracted statistical and proposed session timing resolution features based on the whole dataset. The features importance analysis applied to the proposed timing resolution features indicated by the highest usage of the features was because that the proposed timing features enhanced the classifier to identify and discriminate users. Therefore, the highest score achieved with the two-feature type (Timing resolution features and extracted statistical features), which indicated that the module was being enhanced by the proposed features to differentiate between user's traffic samples. The app_ encoded feature enhanced the module to identify users who indicate that the encoding criteria applied on the features on the feature extracting step. Also, the start_ hour and end_ hour features scored the second and third top highest between all features which indicate the importance of the 24-hour timing resolution features to identify users from their investigated traffic samples.

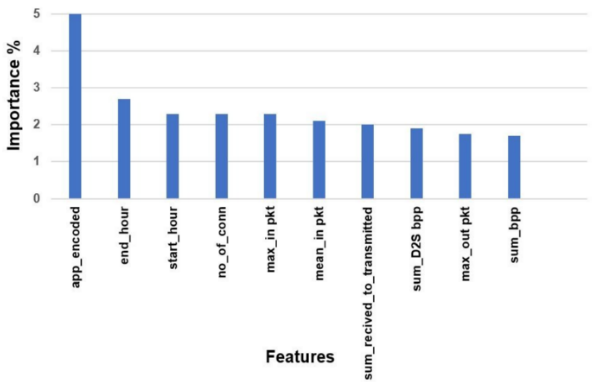


Fig.3: Random Forest Feature Importance

B. Confusion Matrix

The most straightforward way to evaluate the performance of the classifier is based on a confusion matrix especially when the model has more than two classes. Table 6 illustrates a confusion matrix for all users to show the correct and incorrect prediction of each class based on the test data for set 2 features set. The performance of the classification model is ideally high between all classes ranging from 48 -100%. The labels are indicated with users' id from (user 1- user 23) as illustrated in Table 6 for the predicted and true labels. The highest score of the actual class predict 97% as the TP for user 10, and 3% of FN recorded for users (4 and 12), which indicate the ability of the module to identify users with a high score.

Furthermore, user 6 scored the second highest accuracy between all users which is 91% of TP classified samples, there were 9% FN misclassified recorded for user 23. Also, user 3 was recorded as the third highest score on the module with 90% TP and 14% misclassified attributed to users (2 and 20). User 23 recorded the lowest accuracy with 48% TP and 52% FN misclassified attributes to users (2, 4,7,13 and 18) on the module because of the number of traffic samples are the lowest between all users on the module. The reason of the small number of user 23 traffic samples are the number of days which affect the TP for this user comparing to others as there is no traffic for the whole 60 days depends on his usage which is lower than other users on the module.

Table 6: Confusion Matrix (Features Set 2)

1	67	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0
2	0	85	0	0	5	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	5	0	0
3	0	6	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0
4	0	0	0	70	0	0	0	0	0	0	20	0	10	0	0	0	0	0	0	0	0	0	0
5	0	0	0	5	58	0	0	0	6	0	0	0	0	0	5	21	0	0	0	5	0	0	0
6	0	0	0	0	0	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9
7	0	0	0	0	0	0	75	0	8	0	0	0	0	0	0	0	17	0	0	0	0	0	0
8	5	0	0	0	5	0	0	72	0	0	0	0	3	0	0	0	0	10	0	0	5	0	0
9	0	0	0	0	0	0	7	0	75	0	0	0	0	0	0	0	7	0	0	7	0	4	0
10	0	0	0	2	0	0	0	0	0	97	0	1	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	3	0	0	0	0	0	0	87	0	0	0	0	0	7	0	0	0	3	0	0
12	0	0	0	11	0	0	0	0	0	0	0	89	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	7	0	0	7	0	0	57	0	0	8	0	0	0	21	0	0	0
14	0	0	0	14	0	0	8	0	0	0	7	0	0	71	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	15	0	0	71	0	0	14	0	0	0	0	0
16	0	0	0	11	0	0	6	0	6	0	0	0	6	0	0	68	0	0	3	0	0	0	0
17	0	0	0	4	4	0	4	0	0	0	6	0	0	4	0	0	78	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	88	2	0	0	0	0
19	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	18	0	65	0	0	8	0
20	0	0	0	8	0	0	0	0	8	0	0	0	0	0	7	0	0	0	0	71	0	0	6
21	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0	0	0	10	0	0	60	0	0
22	0	0	15	0	0	0	0	10	0	0	0	2	0	0	0	0	0	15	0	0	0	58	0
23	0	12	0	15	0	0	10	0	0	0	0	0	10	0	0	0	0	5	0	0	0	0	48
Uid	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

TRUE LABEL

PREDICTED LABEL

V. Discussion

The experiment results indicated that the nature of the features derived from flow-level generic network traffic is unique, thereby using it to identify and build a user behavioural profile is a promising solution to help the security administrator to make an informed decision about different perspectives. In addition, the proposed features and the analysis of the user traffic information can enhance user identification and behaviour profiling. Moreover, the experiment showed that by utilising session timing resolution and extracted statistical features, the system was able to identify users and represent the usage of applications to up to 74 % level of accuracy as shown in Table 4. Since set 2 achieved a high level of accuracy, it is a clear indication that there is discriminative information exists in the user proposed features predicted as a right classification module. The confusion matrix description of the high attributed level of the truly predicted classes indicates the level of unique information among different users and application usage. Therefore, obtaining a precise user statistical and session timing resolution pattern lead to an accurate module that helps to identify profile users.

Also, observation and calculation of the user behaviour activity among different applications provide a robust approach to identify relevant traffic. When combined with the reverse DNS queries, user MAC address mapping and session timing resolution (an approach uses the reverse DNS quarries to initialise applications and then identify the user data that uses session information to tag all traffic from that user) provides a very successful approach to the target upon the traffic that is most relevant.

The analysis relied on MAC addresses instead of IP addresses to ensure host consistency, as the (DHCP) changes IP address among users during the time. The system successfully identified the individual user with accuracies of 48 - 100% as demonstrated in Table 6. Therefore, the accuracy differs among users due to the volume of traffic and the period of collecting the data, as some of the users scored higher accuracy compared to other users. The explanation of predicted label being high was attributed to the level of uniqueness of users in an application. The applications analysis of this approach was identified based on reverse DNS queries which were implemented relying on DNS lookup utility, which is a good objective in case millions of records needs to be investigated. Moreover, an automated way of dealing with the real traffic used in this approach and provide the ability to deal with any number of users on the investigated network.

VI. Conclusion and Future Work

The present work proposes a method for user identification and behaviour profiling from generic network traffic. The resulted classification accuracy shows that the proposed features based on application-level flow sessions could be utilised to discriminate among users with an accuracy of up to 74%. A supervised machine-learning algorithm was employed to evaluate the analysis algorithm with real data collected from the

Centre for Security, Communications and Network Research (CSCAN) at Plymouth University to investigate the proposed approach.

Apart from the future work, implement different timing resolutions such as (quarter _of_ hour) features to see the effect of the new features and different distribution analysis will be applied on the sessions flow inter-arrival time, to investigate the impact of different thresholds and its effect on system performance. Additionally, more experimental work and analysis will be utilised to examine the effect of each users features based on variance and similarity based on the natures of features.

REFERENCE

[1] J. Yang, Y. Qiao, X. Zhang, H. He, F. Liu, and G. Cheng, "Characterizing user behavior in mobile internet," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, 2015 , pp. 95–106.

[2] Cisco, "Solutions - Cisco's 2017 Visual Networking Index (VNI) Infographic - Cisco," 2017. [Online]. Available: http://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/vni-infographic.html. [Accessed: 03-Jul-2017].

[3] T. Bakhshi and B. Ghita, "User Traffic Profiling: In a Software Defined Networking Context," in *2015 Internet Technologies and Applications (ITA)*, 2015, pp. 91–97.

[4] T Bakhshi, B Ghita, User-centric traffic optimization in residential software defined networks, *2016 23rd International Conference on Telecommunications (ICT),* (2016), pp. 1-6.

[5] T Bakhshi, B Ghita, Traffic Profiling: Evaluating Stability in Multi-device User Environments, *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)* (2016), pp. 731-736.

[6] G. Alotibi, N. Clarke, F. Li, and S. Furnell, "User profiling from network traffic via novel application-level interactions," in *2016 11th International Conference for Internet Technology and Secured Transactions, ICITST (2016*), pp. 279-285 .

[7] F. Dehghani, N. Movahhedinia, M. R. Khayyambashi, and S. Kianian, "Real-Time Traffic Classification Based on Statistical and Payload Content Features," in *2010 2nd International Workshop on Intelligent Systems and Applications*, 2010, pp. 1–4.

[8] M. Finsterbusch, C. Richter, E. Rocha, J. A. Müller, and K. Hänßgen, "A survey of payload-based traffic classification approaches," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 2, 2014 , pp. 1135–1156.

[9] M. V. Vinupaul, R. Bhattacharjee, R. Rajesh, and G. S. Kumar, "User characterization through network flow analysis," in *Proceedings of the 2016 International Conference on Data Science and Engineering, ICDSE 2016*, pp. 1-6 .

[10] A. Ulliac and B. V. Ghita, "Non-intrusive Identification of Peer-to-Peer Traffic," 2010 Third International Conference on Communication Theory, Reliability, and Quality of Service, Athens/Glyfada, 2010, pp. 116-121

[11] T Bakhshi, B Ghita, "On internet traffic classification: A two-phased machine learning approach", Journal of Computer Networks and Communications, 2016

[12] J. Erman, A. Mahanti, and M. Arlitt, "Internet Traffic Identification using Machine Learning," *Global Communications Conference 2006. GLOBECOM '06. IEEE*, 2006, pp. 1–6.

[13] M. E. Kounavis, A. Kumar, H. Vin, R. Yavatkar, and A. T. Campbell, "Directions in Packet Classification for Network Processors," *Vol. 2 Morgan Kaufmann Series in Computer Architecture and Desien*, 2004 , pp. 273–298.

[14] PéterMegyesi, G. Szabó, and S. Molnár, "User behavior based traffic emulator: A framework for generating test data for DPI tools," *Computer Networks*, vol. 92, 2015 , pp. 41–54.

[15] N. Melnikov, "Cybermetrics: User Identification through Network Flow Analysis," *Ifip International Federation For Information Processing*, 2010 , pp. 167–170.

[16] H Oudah, B Ghita, "Network Application Detection Using Traffic Burstiness", *World Congress on Internet Security WorldCIS-2017* (2017), pp. 23 – 28.

[17] C. M. McDowell, "Creating Profiles From User Network Behavior," 2013.

[18] C. Banse, D. Herrmann, and H. Federrath, "Tracking users on the Internet with behavioral patterns: Evaluation of its practical feasibility," *IFIP Advances in Information and Communication Technology*, vol. 376 AICT, 2012 , pp. 235–248.

[19] M. Kirchler, D. Herrmann, J. Lindemann, and M. Kloft, "Tracked Without a Trace: Linking Sessions of Users by Unsupervised Learning of Patterns in Their DNS Traffic," in *the 2016 ACM Workshop on Artificial Intelligence and Security*, 2016, pp. 23–34.

[20] X. Gu, M. Yang, J. Fei, Z. Ling, and J. Luo, "A Novel Behavior-Based Tracking Attack for User Identification," in *2015 Third International Conference on Advanced Cloud and Big Data*, 2015, pp. 227–233.

[21] G. Alotibi, F. Li, N. Clarke, and S. Furnell, "Behavioral-Based Feature Abstraction from Network Traffic," *Iccws 2015-The Proceedings of the 10th International Conference on Cyber Warfare and Security 2015,* pp.1-9.

[22] S. H. Oh and W. S. Lee, "An anomaly intrusion detection method by clustering normal user behavior," *Computers and Security*, vol. 22, no. 7, 2003, pp. 596–612.

[23] D. lookup Utility, "DNS lookup utility UBUNTo," *DNS lookup utiliy*, 2005. [Online]. Available: http://manpages.ubuntu.com/manpages/bionic/man1/host.1.html.

[24] P. Haag, "nfdump and NfSen," 2006. [Online]. Available: http://nfdump.sourceforge.net/.

[25] Internet Software Consortium, "host(1) - Linux man page." [Online]. Available: https://linux.die.net/man/1/host.

[26] E. Garsva, N. Paulauskas, G. Grazulevicius, and L. Gulbinovic, "Packet inter-arrival time distribution in academic computer network," *Elektronika ir Elektrotechnika*, vol. 20, no. 3, 2014 , pp. 87–90.

[27] C. X. Zhang, J. S. Zhang, and G. Y. Zhang, "An efficient modified boosting method for solving classification problems," *Journal of Computational and Applied Mathematics*, vol. 214, no. 2, 2008, pp. 381–392.

[28] Y. Yang, "Web user behavioral profiling for user identification," *Decision Support Systems*, vol. 49, no. 3, 2010 ,pp. 261–271.

[29] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, "Data preprocessing for supervised learning," *International Journal of Computer Science.*, vol. 1, no. 2, 2006 , pp. 111–117.