iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** VIII **Month of publication:** August 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

An Open Platform for Internet Traffic Classification in TIE

M.Thanigavel[1], R.M. Mallika[2], T.Sujilatha3
[1,2,3]CSE Dept, GKCE Sullurpeta, Nellore, A.P, India

Abstract: ***An open source internet traffic classification systems designed for both experimental and operational use, in this paper we are using traffic identification engine it's also a open source tool, TIE was developed in 2008 to promote sharing common implementations and data in network field, here the accurate traffic identification and insightful measurements form the foundation of network business intelligence and network policy control. Without identifying and measuring the traffic flowing on their networks, CSPs are unable to craft new subscriber services, optimize shared resource utilization, and ensure correct billing and charging .First and foremost, CSPs must understand their use cases, as these determine tolerance for accuracy. It is likely less of a problem if reports show information that is wrong by a small margin, but it can be catastrophic if subscriber billing/charging is incorrect or management policies are applied to the wrong traffic. So-called embedded solutions typically make do with simplistic approaches. Faced with such variation, CSPs must understand the technologies, trade-offs (e.g., completeness and false positives), and deployment challenges (e.g., routing asymmetry; tunnels and encapsulation; encryption, obfuscation, and proxies) that exist in the context of traffic classification, and only with this detailed understanding can they ask the right questions in order to truly understand what a vendor is providing, and any limitations that would otherwise be hidden.***
Keywords: ***TIE-Traffic Identification Engine, CSP-Communication Service Provider, Traffic Classification, Open Source Platforms, Network and Traffic Monitoring.***

I. INTRODUCTION

In this paper, we briefly describe TIE's components and functionalities by detailing some of the design choices focused on multi-classification, comparison of approaches, and online traffic classification.

A. What is TIE?

In order to compare different classification approaches, TIE proposes a unified representation of classification results. It defines IDs for application classes (applications) and associates them with group classes (groups), which include applications offering similar services. Such mapping enables the comparison of techniques working at different granularities (e.g., applications vs. groups) or, for instance, the comparison of traffic classifiers which have application-level protocol classes using a coarser granularity. Moreover, several application sub-classes (sub applications) are associated with each application, in order to discriminate related traffic flows serving different purposes (e.g., signaling vs. data, Skype voice vs. Skype chat, etc.).

B. Operating Modes

TIE can be run in three operating modes, each one corresponding to a different overall behavior.

1) *Offline Mode:* a flow is classified only when it expires or at the end of TIE execution. This mode is useful for evaluating classification techniques when no timing constraints apply, or when classification requires observing flows for their entire lifetime.
2) *Real Time Mode*: a flow is classified as soon as enough information is collected, thus implementing online classification. This mode can be used for policy enforcement (QoS, admission control, billing, firewalling, etc.).
3) *Cyclic Mode:* flows are classified at regular time intervals (e.g., each 5 minutes) and the results are stored into separate output files related only to the corresponding interval, which is useful to build live traffic reports.

All working modes can be applied to both live traffic and traffic traces. Among them, the real time mode is the one imposing most constraints and heavily influencing the whole design of the TIE engine.

C. Architecture Overview And Functionalities

TIE is written in C, targeting Unix-like operating systems, currently supporting the Linux, FreeBSD and Mac OS X platforms. The software constitutes of a single executable and a set of classification plugins dynamically loaded at run time. Moreover, the TIE framework includes a collection of utilities Distributed with the source code to post-process output files.

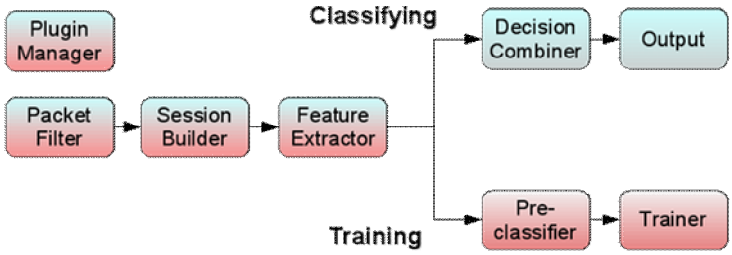


Figure 1: TIE Architecture

The TIE engine processes packets in five stages, with the last two varying depending if TIE is used for classifying traffic or for training machine learning classifiers.

1) *Packet Filter:* This stage captures link-layer frames – or reads them from file – and filters them according to configurable rules. It is based on the well-known Libpcap library and its filtering capabilities are implemented by using both the Berkeley Packet Filters and additional user-space filtering rules (e.g., selecting traffic within a specified time range).
2) *Session Builder:* This stage organizes network traffic into sessions, i.e., the flow objects to be classified. We defined a generic concept of session to support the various types of traffic flow objects adopted in literature
   a) *Flow:* defined by the {SRC_{IP} , SRC_{port}, $DEST_{IP}$, $DEST_{port}$, transport protocol} tuple and an inactivity timeout, with a default value of 60 seconds;
   b) *Bi-flow:* defined by the {SRC_{IP} , SRC_{port}, $DEST_{IP}$, $DEST_{port}$, transport protocol} tuple, where source and destination can be swapped, and the inactivity timeout is referred to packets in any direction;
   c) *Host:* containing all the packets a host generates or receives. A timeout can be optionally set.

Although bi-flows can be considered a computationally efficient approximation of TCP connections (they just require a lookup on a hash table for each packet), some applications may need a more accurate identification of their lifetime. Hence, TIE implements computationally-light heuristics based on TCP flags that, applied to bi-flows, yield to a better approximation of TCP connections, avoiding the segmentation of TCP connections into several bi-flows in presence of long periods of silence (e.g., Telnet, SSH). This stage keeps track of sessions using a chained hash table, and – to properly work with high traffic volumes – it includes a Garbage Collector component responsible for periodically releasing the resources related to classified and expired sessions.

3) *Feature Extractor.* This stage is responsible for collecting the features required by the classification plug-in, and it is triggered by the Session Builder for every incoming packet. As reported in Table 1.a, for each session it provides (i) basic features (al-ways available to classifiers) and (ii) advanced features (extracted on-demand). In order to optimize computational efficiency, advanced features are collected only if requested by a command-line option and if a skip-session flag is not set (this flag avoids processing additional packets when enough packets have already been inspected). While we included support for features based on the most common classification techniques (port-based, flow-based, payload-based, etc.), TIE can be easily ex-tended to extract new features based on definitions already published in literature or to support new techniques. In order to rapidly experiment with techniques implemented by external tools, this stage can optionally dump for each session the corresponding classification features along with the label as-signed by a classifier (e.g., a payload based classifier can used to establish ground truth). TIE supports dumping features directly in some common for-mats, such as the arff format used by WEKA – one of the most used tools in the field of machine-learning classification.
4) *Decision Combiner:* When TIE is used to classify traffic, the fourth stage of the TIE engine consists in a multi-decisional engine made of a Decision Combiner (hereinafter DC) and one or more Classification Plug-in (hereinafter classifiers) implementing different classification techniques.

II. INTRODUCTION TO INTERNET TRAFFIC CLASSIFICATION

Traffic classification goes beyond identification (i.e., determining what the traffic is) and extends into extracting pertinent information (e.g., video resolution, media type, CDN of origin, etc.) and measuring characteristics (e.g., duration, counting events, determining QoE, etc.); however, not all solutions are created equal.

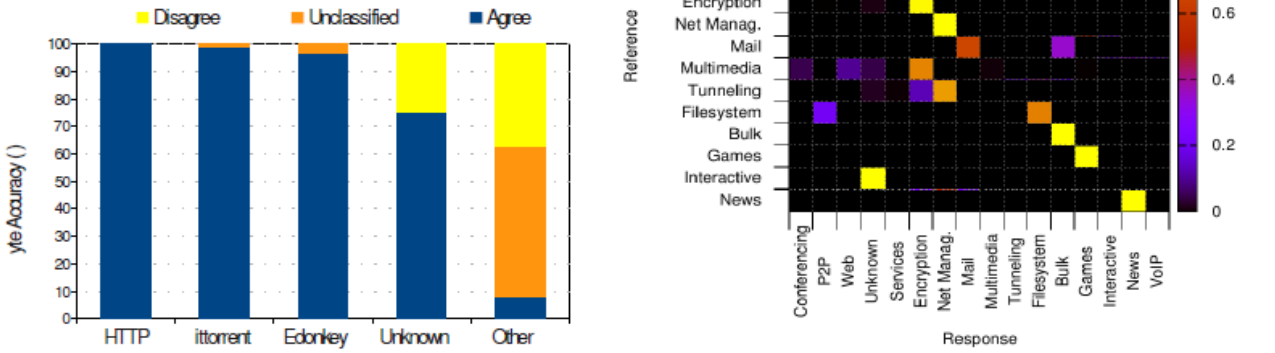
A. *Traffic Identification*

In general, traffic will be described as being one or more of these types:

1) *Protocol:* a strict set of rules and formats that define how two or more elements share Information (the information flow could be one way or bidirectional). Examples include UDP, TCP, HTTP, RTMP, SIP, FTP, and SMTP1.
2) *Application:* traffic associated with a particular software program. Examples include Skype, Netflix, PPStream, and games.
3) *Website:* all the web pages that are part of a particular web domain and all content that is exchanged with a particular domain (whether or not the content corresponds to a web page)
4) *Service:* a more general term that can include websites like Twitter and Face book, cloud services like Sales force, online storage, and many others.
5) *Provider:* typically used to differentiate a brand within a type of traffic. For instance, many different video providers use RTMP, and many different voice services rely on SIP.

Classifier	Accuracy on applications		Accuracy on categories of apps	
	sessions	bytes	sessions	bytes
Portload	74.24%	97.83%	73.88%	97.45%
Port	19.57%	25.12%	15.95%	23.89%

(a) Overall accuracy of *Portload* and *Port* plugins



(b) *Portload* vs *L7*: byte-accuracy on applications generating most of the traffic (left); confusion matrix with applications grouped into categories (right)

Figure 2: Classification accuracy as obtained by comparing *Port* and *PortLoad* against *L7*.

TIE can be used as an enabling technology for rapid development and effective comparison of traffic classification approaches in terms of accuracy. In, we used it to develop and evaluate a lightweight payload-based classifier called Port Load, which inspects only the first 32 payload bytes of the first packet in both directions of each session. We compared Port Load to (i) a port-based classifier (the Port plug-in, based on Coral Reef signatures) and (ii) a DPI classifier (L7 plug-in, based on L7-Filter). Such comparison has been conducted on a full-payload traffic trace of 40 GB captured at University of Napoli, Italy. To perform such comparison, we first launched TIE with only the L7 plug-in enabled, in order to use its results as a reference. We then executed TIE respectively enabling the Port Load and the Port plug-in. By running the tie stats utility on the generated output files, we obtained the related confusion matrices, from which we evaluated the (expected) loss in accuracy when moving from DPI to Port Load and port approaches. As shown in Figure 2.a, Port Load reported an overall accuracy and byte-accuracy of about 74% and 97% respectively, showing very good results on heavy flows. Figure 2.b (right) represents the confusion matrix (with applications grouped into categories) of Port Load against L7. The warm colors on the main diagonal denote a good accuracy on most (categories of) applications, whereas few cells outside of it show application categories that are not well identified by Port Load. Figure 2.b (left) summarizes the classification results for the applications with the largest byte-counts. Each bar (corresponding to a class) represents the percentage of bytes on which Port Load respectively agreed, disagreed, or returned

unknown, with respect to L7. TIE allowed us to conclude that the Port Load approach is a valid alternative to DPI when the main objective is to obtain high byte-accuracy (97%) and lower unknown percentage (40% less).
TIE has been widely used by the traffic classification community, by both academic and commercial organizations in this paper, we first provide a brief overview of the evolution of traffic classification and of the challenges addressed in the last years in this research field. We then describe the main components and functionalities of TIE by detailing some of our design choices, also driven by such analysis of the state of the art. It is also common to see sub-classifications that add further granularity to a classification. For instance, YouTube might be designated as HD or non-HD, or Bit Torrent might be distinguished by being encrypted or not. Frequently, sub-classifications are used to add clarity rather than in response to convey a particular technical distinction.

There are many other terms that are important in the context of traffic classification, including:
Library: the list of traffic types that are supported (i.e., identified and measured) by a solution Identifying and Measuring Internet Traffic: Techniques and Considerations
Content Type: typically refers to a finer level of classification of traffic as being video, text, images, audio, etc.
False Positive: traffic that is incorrectly identified as being of Type B; the 'positive' identification of the traffic as being of Type B is false
False Negative: traffic of Type A (that is supposed to be recognized) that is not identified as Type A; the 'negative' identification (i.e., "this is not Type A") is false
Unrecognized Traffic: traffic that is not identified as belonging to any of the supported types
Over-the-Top (OTT): traffic that is on a CSP's network that does not originate from a service provided by the CSP
State-full: requiring awareness of or maintaining a finite number of states
Data Traffic and Control Traffic (alternatively called 'data channel' and 'control channel'): data traffic is the actual payload or content being exchanged, whereas control traffic governs that exchange; for instance, in a video stream the control traffic will include a feedback loop to convey user instructions (e.g., play, pause, seek) and transport quality information
Signature: a pattern corresponding to a known traffic type against which observed traffic types are compared. In the most basic definition, a signature is a regular expression that is applied to packets. In the most advanced definition, a signature can be a state full technique that monitors state changes within data and control traffic to extract information required for further identification (e.g., where the next data flow will appear) or simply requested (e.g., the provider of a video)

B. Techniques

Many techniques are applied, alone or in combination, to identify traffic and extract relevant fields. It's not uncommon for vendors to use the term 'signature' to mean any and all techniques. Increased reliability and accuracy is typically achieved at the cost of greater processing complexity.

This list introduces some popular techniques, in order of ascending reliability/accuracy:

1) *Port Number:* this approach simply looks at the port number of the traffic and concludes that the traffic is of the type commonly associated with this port. Because of the certainty of false Positives due to many traffic types taking random ports, this approach should not be used in any circumstances in which reliable identification is needed.
2) *Regular Expression:* a byte pattern that is (assumed/expected to be) a unique identifier for a particular traffic type. The longer a regular expression, the less chance of there being a false positive due to matches against random data. Identification typically requires that one or more regular expressions be applied across multiple packets and flows.
3) *Tracker:* a state-full technique that monitors state changes within data and control traffic both to extract information required for further identification (e.g., where the next data flow will appear) and to provide addition information in general.
4) *Analyzer:* similar to a tracker, but with complete protocol awareness; that is, an analyzer can extract any and all meaningful pieces of information due to a complete understanding of a protocol. In the previous example of adaptive video, a tracker would be sufficient to determine from the control traffic where the data traffic would appear, but an analyzer is required to

extract the resolution and codec information.

C. Additional Data and Measurements

Beyond simply identifying traffic, what additional data can be extracted or determined and what measurements can be made? For instance, additional data can include: service tier, IP address, MAC address, content provider, client device, media stream type, media container, video resolution, video codec, audio codec, operating system, browser, session protocol, and transport protocol, to name a few fields popular with CSPs. If policy management is an objective, then a CSP needs to know if this data is actionable in real-time (i.e., can the data serve as a condition that triggers an action).

Measurements can add immense value, particularly for business intelligence, and can include:

Duration of a video or audio stream
Voice or video quality of experience
Counts of the number of events
Tracking of "top" items (e.g., most frequently requested URLs, most popular video providers, etc.)
Summations (e.g., adding up a number of observed or measured values)

Some vendors even include the ability for a CSP to define their own custom measurements to answer questions as they are asked. Once again, if policy management is an objective, then a CSP needs to know if this data is actionable in real-time.

Traffic Category	Description	Examples
Storage	Large data transfers and online storage services	FTP, NNTP, PDBox, Rapidshare,Mega, Dropbox
Gaming	Console and PC gaming	Nintendo Wii, Xbox Live ,Playstation Network, World of Warcraft
Marketplaces	Marketplaces for application and content downloads and software updates	Google Play Store, Apple iTunes ,Windows Update
Administration	Protocols used to administer the network	DNS, ICMP, NTP, SNMP
File-Sharing	File-sharing applications, whether peer-to-peer or direct	BitTorrent, eDonkey, Ares, Pando,Foxy
Communications	Applications, services, and protocols that allow email, chat ,voice, and video communications	Skype, ICQ, SIP, MGCP, IRC,FaceTime, WhatsApp, Gmail, SMTP
Real-Time Entertainment	Applications and protocols that allow 'on-demand' entertainment	Adaptive or progressive audio and video peercasting Place shifting specific streaming sites and services
Tunneling	Protocols and services that allow remote access to network resources , or provide encryption or encapsulation	SSL, SSH, L2TP,Remote Desktop,VNC, PC Anywhere
Social Networking	Websites and services focused on enabling interaction and sharing	Facebook, Twitter, Habbo, Bebo
Web Browsing	Web protocols and specific websites	HTTP, WAP browsing

Table 1: Traffic Classification and Identification with its examples

III. TECHNOLOGY REQUIRED:

Before traffic identification signatures and techniques can even be applied, or in the course of applying such techniques, a number of technical hurdles must be overcome. To be a truly viable solution, it is necessary that all of these challenges be addressed.

A. State-full Protocols

Many types of traffic can only be positively identified if the recognition technology has complete awareness of protocol state.

B. Related Flows and Sessions

In many cases, a positive identification is only possible if the recognition solution can correlate and apply signatures to the same asset across multiple transactions issued into the same, or different, connections.

C. Routing Asymmetry

By design, all broadband networks exhibit routing asymmetry of one form or another; that is, traffic packets relating to the same flow can take different routes through the network. CSPs must make certain that any solutions they are considering can accurately identify and measure all types of traffic in all network configurations, and CSPs should take heed that this is certainly not always the case.

D. Tunnels and Encapsulation

A significant portion of traffic that will be inspected for identification is contained within tunnels (e.g., GTP, GRE, L2TP, Q-in-Q, and IP-in-IP) or encapsulation (e.g., MPLS, EoMPLS, and VLAN). For maximum utility, the identification solution must be able to inspect (and apply policy control) within the tunnels and the encapsulation.

E. Devices and Tethering

In this era of the Internet of Things, there is no practical limit to the number of devices that can have an IP address. The increasing number and diversity of connected devices brings opportunity to CSPs who can identify trends and can, in turn, create services that cater to these unique demands. Identifying and Measuring Internet Traffic: Techniques and Considerations With respect to traffic classification, the rich array of connected devices impose a number of requirements.

F. Client and Access Devices

First, it is important to differentiate between *client device* and *access device*:

1) A *client device* is the device that originates packets on the network
2) An *access device* is the device that connects to the access network and owns the IP connectivity session

Consider Figure 3, below. Within the home network, there are many client devices (e.g., laptop, tablet, mobile phone), and the diagram could have included many others (e.g., game console, smart thermostat, etc.), but there is only a single access device (i.e., home router). The home router connects to the CSP's network, but the client devices actually originate packets. In the mobile network, things can become a bit blurred. Typically, any device that connects to the mobile network is an access device and, in most cases, the mobile device is also a client device. However, in the case of tethering, a clear split is made: in this case, the mobile phone serves as an access device (as a Wi-Fi hotspot), while the tethered laptop is the client device. Beyond simply differentiating between client and access devices, the next consideration is the information and insight available. For instance, is detailed measurements (e.g., application usage, video duration, quality of experience) available per-device? What can be gleaned about the device identity (e.g., manufacturer, model, operating system, browsers, etc.)? , The richer this information, the richer the insight and, potentially, the richer the subscriber services that can be enabled.

G. Tethering Detection

Many CSPs want to offer tethering services as add-ons to existing data plans, but to do so they need to be able to detect and manage tethered devices. The most robust plans require policy control platforms that can apply separate policy to the tethered and access (i.e., hotspot) devices.

H. Network Address Translators

In the home network and in the case of tethering, and in other network environments (e.g., public Wi-Fi hotspots), the client devices exist behind a network address translator (NAT). The NAT serves as a Identifying and Measuring Internet Traffic: Techniques and Considerations Single point of access connection and, in effect, 'hides' the devices that are behind it. Detecting and identifying individual devices behind a NAT are a complex task and one of which very few solutions are capable.

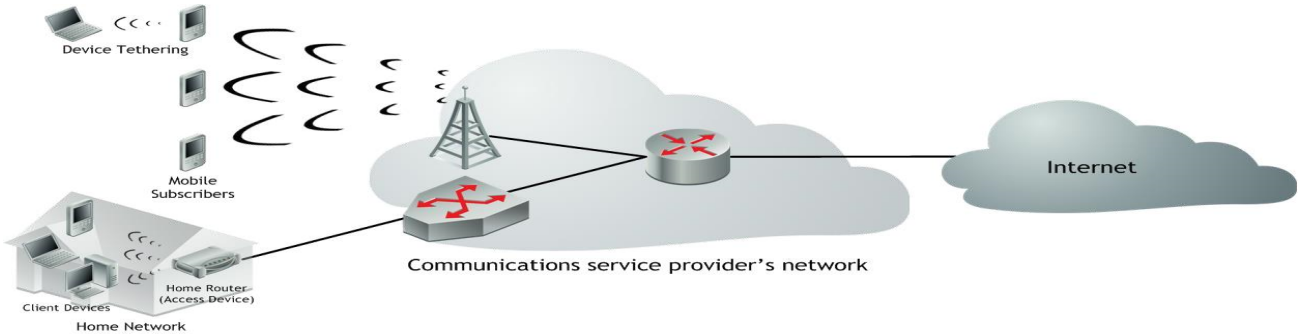


Figure 3 - Client and access devices in fixed and mobile access networks

I. Real-Time Network Policy Control

All of this device information is useful from a business intelligence and strategy perspective, but becomes much more vital when it is available in real-time for network policy control. Naturally, passive/offline post-processing cannot be incorporate into real-time decisions and enforcement, but some solutions perform the device differentiation and tethering detection in real-time. It is important for CSPs to ask detailed questions in order to accurately understand a particular vendor's capabilities.

J. Encryption, Obfuscation, and Proxies

Information can be hidden or guarded in many ways, and two widespread mechanisms in the context of traffic classification are:

1) *Encryption:* encoding information such that it can only be read by an authorized party
2) *Obfuscation:* hiding or disguising information to prevent detection

It is important for CSPs to keep in mind that encryption does not mean something is undetectable or unidentifiable, it just means that the content is private. Because most encrypted traffic relies on accepted standards (e.g., IPSEC, TLS), it is generally easy to detect, although capabilities do vary by solution vendor. Obfuscation measures vary widely, and are typically used to avoid detection and policy management. Early approaches randomized ports and moved information around within packets in order to overcome relatively simple pattern recognition algorithms, finally, it is worth explicitly noting that techniques to encrypt and obfuscate traffic are evolving rapidly, so it is vitally important that CSPs understand and assess their solution vendors' capabilities to adapt to these changes. For instance, can software updates provide new capabilities in the field, or will a hardware upgrade be required? Can the traffic classification solution combine multiple techniques (e.g., measurements, analyzers, and heuristics).

IV. TIE AND THERE RESEARCH COMMUNITY

Starting from the first release of TIE in 2009 (avail-able upon request by email), the platform has been cited in more than 35 publications and, through several collaborations, it has been extended to support new classification features and schemes – including the combination of multiple classifiers – and to run techniques already available in WEKA (see Sec. 4.3). Since 2011, when a more recent version of TIE was released, TIE has been downloaded more than 150 times according to statistics collected at the official website (unique downloads of distinct users who filed a request through a web form). Download requests originated from universities (62%), companies (30%), and individuals (8%).

V. CONCLUSION

Here by we conclude that accurate traffic identification and insightful measurements form the foundation of network business intelligence and network policy control is required, without identifying and measuring the traffic flowing on their networks, CSPs are unable to craft new subscriber services, optimize shared resource utilization, and ensure correct billing and charging. Traffic classification goes beyond identification and extends into extracting information and measuring characteristics. However, not all solutions are created equally. Many techniques exist to identify traffic and extract additional information or measurement quantities, ranging from relatively simple to extremely complex state, the advance techniques that can provide the most comprehensive information and actionable utility are processor-intensive and therefore only available on best-of-breed DPI and policy control platforms. So-called embedded solutions typically make do with simplistic approaches.

we plan to further extend TIE by: (i) investigating the optimal combination strategy and set of classifiers to generate reliable ground truth while preserving privacy; (ii) investigating strategies for multi-threaded classification, exploiting a) offloading techniques offered by recent traffic capturing engines such as multi-queue adapters and multi-line buses between NICs and CPU cores, b) GPU extensions, c) NUMA capabilities, etc..

VI. ACKNOWLEDGEMENT

I like to thank our PRINCIPAL, HOD and OTHER FACULTIES for their valuable comments and helpful suggestions, to make this paper publish and also they help for my academic growth.

REFERENCES

[1] Dainotti, A. Pescap´e, and K. C. Claffy, “Issues and future directions in traffic classification,” Network, IEEE, vol. 26, no. 1, 2012, pp. 35–40.
[2] A. Finamore, M. Mellia, M.Meo, M.Munafo,and D.Rossi,“Experiences of internet traffic monitoring,” Network, IEEE, vol. 25, no. 3, 2011, pp. 8–14.
[3] F. Gringoli, L. Nava, A. Este, and L. Salgarelli, “Mtclass: Enabling statistical traffic classification of multi-gigabit aggregates on inexpensive hardware,” Wireless Commu-nications and Mobile Computing Conference (IWCMC), 2012 8th International, 2012, pp. 450–455.
[4] A. Este, F. Gringoli, and L. Salgarelli, “On-line svm traffic classification,” Wireless Communications and Mobile Com-puting Conference (IWCMC), 2011 7th International, 2011, pp. 1778–1783.
[5] S. Lee, H. Kim, D. Barman, S. Lee, C.-k. Kim, T. Kwon, and Y. Choi, “Netramark: A network traffic classification benchmark,” SIGCOMM Comput. Commun. Rev., vol. 41, no. 1, Jan. 2011, pp. 22–30.
[6] M. Canini, W. Li, A. W. Moore, and R. Bolla, “Gtvs: boost-ing the collection of application traffic ground truth,” Traffic Monitoring and Analysis. Springer, 2009, pp. 54–63.
[7] F. Gringoli, L. Salgarelli, M. Dusi, N. Cascarano, F. Risso, and kc claffy, “Gt: picking up the truth from the ground for internet traffic,” Computer Communication Review, vol. 39, no. 5, 2009, pp. 12–18.
[8] A. Dainotti, W. De Donato, A. Pescape, and P. Salvo Rossi, “Classification of network traffic via packet-level hidden markov models,” IEEE Global Telecommunications Confer-ence, 2008, pp. 1–5.
[9] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, “Traffic classification through simple statistical fingerprinting,” ACM SIGCOMM CCR, vol. 37, no. 1, Jan. 2007, pp. 7– 16.
[10] L. Salgarelli, F. Gringoli, and T. Karagiannis, Comparing Traffic Classifiers,SIGCOMM Comput.Commun. Rev., vol. 37, no. 3, 2007, pp.65–68.
[11] K.G.S. Venkatesan and M. Elamurugaselvam, ―Design based object oriented Metrics to measure coupling & cohesion‖, International journal of Advanced & Innovative Research, Vol. 2, Issue 5, PP. 778 – 785, 2013.
[12] A. Dainotti, W. de Donato, and A. Pescapé, ―TIE: A Community-Oriented Traffic Classification Platform,‖ Traffic Monitoring and Analysis, Springer, 2009, pp. 64–74.
[13] T. T. Nguyen and G. Armitage, ―A Survey of Techniques for Internet Traf-fic Classification using Machine Learning,‖ IEEE Commun. Surveys & Tuto-rials, vol. 10, no. 4, 2008, pp. 56–76.
[14] http://thanigavelm.blogspot.in/2014/01/latest-technologies-in-computer-science\Internet Traffic Classification in TIE.html
[15] L. Bernaille, R. Teixeira, and K. Salamatian, ―Early Application Identifica-tion,‖ Proc. 2006 ACM CoNEXT Conf., 2006, p. 6.
[16] G. Aceto et al., ―Portload: Taking the Best of Two Worlds in Traffic Classi-fication,‖ IEEE INFOCOM Wksps., 2010, pp. 1–5.
[17] N.Williams, S.Zander, and G.Armitage, A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification vol. 36, no. 5, Oct. 2006, pp. 7–15.
[18] G. Szabó et al., ―Traffic Classification over Gbit Speed with Commodity Hardware,‖ IEEE J. Commun. Software and Systems, vol. 5, 2010.
[19] A. Callado et al., ―Better Network Traffic Identification Through the Inde-pendent Combination of Techniques,‖ J. Network and Computer Applica-tions, vol. 33, no. 4, 2010, pp. 433–46
[20] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, ―Blinc: Multilevel Traf-fic Classification in the Dark,‖ ACM SIGCOMM Comp. Commun. Review, vol. 35, no. 4, 2005, pp. 229–40

REFERENCE TABLE

Category	Description	Availability	Packets inspected
Basic	number of upstream/downstream packets	always	every packet
	number of upstream/downstream packets carrying payload		
	amount of upstream/downstream bytes		
	duration		
	source/destination port		first packet
	transport layer protocol		
Advanced	inter-packet time among the first n packets	on-demand	first n packets
	packet and payload size of the first n packets		
	first n bytes in the first packet with payload (per direction)		first packet per direction
	stream of payloads up to b bytes		packets necessary to collect b payload bytes
	upstream/downstream payload sizes stats (min, max, mean, variance)		every packet
	upstream/downstream inter-packet times stats (min, max, mean, variance)		
	full packet content (headers + payload)		
	upstream/downstream round-trip times stats (min, max, mean, variance)	on-demand (with biflow sessions only)	

(a) Supported basic and advanced per-session features.

Table 2: Breakdown of TIE extensible functionalities

Label	Technique	Category	Training
NB	Naive Bayes	Bayesian	Confusion Matrix
MV	Majority Voting	Vote	
WMV	Weighted Majority Voting		
D-S	Dempster-Shafer	Dempster-Shafer	
BKS	BKS	Behavior Knowledge Space	BKS
WER	Wernecke		BKS & Confusion Matrix
ORA	Oracle	Oracle	*n.a.*
PRI	Priority-based	*n.a.*	

(b) Implemented combination algorithms.

Classification Plugin	Features based on	Classification approach	Collaborations and contributions from the community
Port	Protocol ports	Port-based	Developed by UNINA, signatures from CAIDA
L7	Payload	Deep payload inspection	Developed by UNINA, code and signatures from Linux L7-filter
PortLoad	Payload	Lightweight Payload Inspection [8]	Developed by UNINA
GMM-PS	First few packet sizes	Gaussian Mixture Models [7]	Developed by UNINA
HMM	Packet size and inter-packet time	Hidden Markov Models [18]	Developed by UNINA
FPT	Packet size and inter-packet time	Statistical [19]	Joint work between UNINA and University of Brescia in the context of the RECIPE research project
Joint	Packet size and inter-packet time	Nearest Neighbor [20]	Joint work: UNINA, CAIDA, Seoul National University
OpenDPI	Payload	Deep payload inspection	Joint work: UNINA, TU Munchen
WEKA/arff	Any information	Machine Learning [21]	Developed by UNINA in collaboration with more than six research groups (e.g., THALES Communication and Security, Tokyo Institute of Technology, etc.)

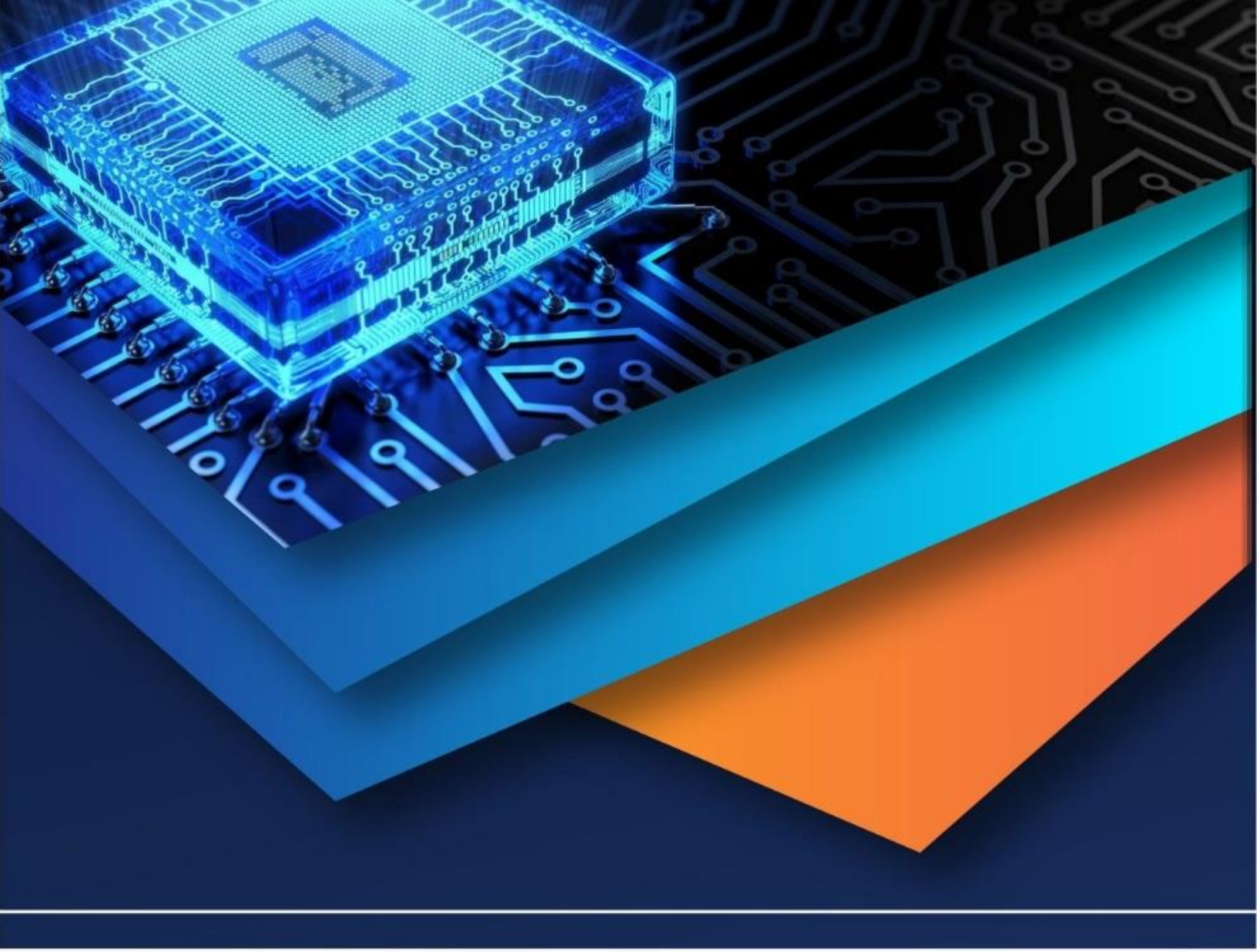
(c) Classification plugins developed since 2008.

AUTHORS

Mr M.THANIGAVEL, Pursuing my M.Tech (CSE) in GOKULAKRISHNA COLLEGE OF ENGINEERING, Sullurpeta , and my area of interest is NETWORKING ,DISTRIBUTING SYSTEM & CLOUD COMPUTING, E-mail id: thaniga10.m@gmail.com

Mrs. R.M. MALLIKA, ASST PROFESSOR in the Department of CSE at GOKULA KRISHNA COLLEGE OF ENGINEERING, Sullurpeta, and my area of interest is DMDW, COMPILER DESIGN, OS, SPM & NETWORK etc., E-mail id: mallika.521@gmail.com.

Miss T.SUJILATHA., ASST PROFESSOR in the Department of CSE at GOKULA KRISHNA COLLEGE OF ENGINEERING, Sullurpeta, and my area of interest is networking, Cloud computing, WSN and DMDW etc., E-mail id: illu.suji@gmail.com.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY