iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: V Month of publication: May 2019

DOI: https://doi.org/10.22214/ijraset.2019.5484

www.ijraset.com

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

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

Adaptive Wildcard Rules for TCAM Management using Cache Replacement Algorithms

Miss. Shrutika Kusekar[1], Prof. Hyder Ali Hingoliwala[2]
[1]PG student, [2]Associate Professor, Computer Department, JSPM's JSCOE, Handewadi, Pune

Abstract: Software-defined networking (SDN) could be a style which means to form systems lightsome and elastic. SDN could be a technology that allows cloud computing and network engineers and directors to retort quickly to dynamic business necessities through a centralized management console. The target of SDN is to reinforce network management by empowering ventures and specialist suppliers to react quickly to dynamic business conditions. Ternary Content Available Memory (TCAM) capacity downside is a very important issue in Software-Defined Networking. Rule caching is associate degree economical technique to resolve the TCAM capability downside. Here, we tend to utilize cover-set methodology to unravel the rule dependency downside and propose a wildcard-rule caching algorithmic program to cache rules into TCAM. Additionally, we tend to additionally propose a cache replacement algorithmic program considering each temporal and spatial locality. Per the simulation results, our wildcard-rule caching algorithmic program and cache replacement algorithm have higher performance than previous works in terms of caching magnitude relation and hit ratio, severally.
Keywords: Software Defined Networking, Wild-rule caching, Ternary Content Addressable Memory.

I. INTRODUCTION

Software-Defined Networking may be a greenhorn network architecture that has a world read of network state for network directors to manage network services. SDN controller maintains the flow tables within the switches to abide by with network policies. The flow tables are enforced by the TCAM within the fashionable switches. TCAM will hunt a packet's header and compare the matching patterns of the packet to the match field of all rules within the flow table in parallel. Though TCAM will forward packets quick, there are solely 2-20K flow table sizes in Commodity SDN switches that are a lot of but RAM based storage. Therefore, the TCAM capability downside is associate degree important issue in SDN. Software-defined measure has the capacity to enable synchronal, dynamically instantiated, and measure tasks. In this approach, Associate in Nursing SDN controller orchestrates these measure tasks at multiple abstraction and temporal scales, supported a world view of the network. Samples of measure tasks embrace distinctive flows extraordinary a given threshold and flows whose volume changes considerably. In an exceedingly cloud setting, every tenant will issue distinct measurement tasks. Some cloud services have an oversized range of tenants, and cloud suppliers already provide straightforward per-tenant measurement services.

A content cell in TCAM is often in one among 3 states: don't care (X) and binary states (0, 1). A content cell transitions to the don't-care state once its associated mask bit is enabled. A masked content cell continually produces a match result regardless of compare-signal and functions sort of a wild card. Full TCAM has Associate in nursing freelance mask cell for every content cell. However, in several applications, a mask cell are often shared with multiple

Memory cells in multiple words to scale back chemical element space prices. Since TCAM at the same time searches entire TCAM memory cells to search out matching entries with compared signals, the entire memory array is accessed and compared in parallel. Such parallel execution consumes vital power. The intensive power consumption typically erases the advantage of the high-speed search operation and prohibits use of TCAM devices. Therefore, power reduction is that the key call criteria when exploitation TCAM devices in several applications. There are two vital areas in TCAM style that decision for innovative power-saving techniques. These are compare-data and match line. Because the range of word entries increase, the match-line power consumption becomes a lot of vital than the compare data power consumption.

One of the rule caching techniques is wildcard-rule caching. Wildcard-rule caching might keep additional TCAM space than exact-match rule matching. However, there are different priorities between completely different rules per the network policies. If 2 wildcard rules overlap with one another and we solely cache the lower-priority one within the TCAM, the packets matching the overlapping field house of those 2 rules will improperly match lower-priority rule.

Wildcard rule caching faces many major challenges.as a caching system, reducing the rate of cache miss and avoiding

©IJRASET: All Rights are Reserved

Memory overflow are vital. We conferred the preliminary style of a completely unique reactive wildcard rule caching system named CAB in. The main idea of CAB is to partition the geometric illustration of the rule set, or the sector house of packet headers, into several little logical structures named buckets, and to associate every bucket with one or multiple rules consistent with its location within the field space. Mistreatment the flow table pipeline supported by OpenFlow switches, buckets work as filters for packets to match the foundations. By caching the bucket with associated rules, CAB is ready to make sure correct packet forwarding.

CAB's rule caching policy and multi-table style have bonded correct packet forwarding. During this section, we further introduce the adaptive Cache Management (ACME). The goal of ACME is modify the buckets with traffic awareness so that flow table usage is usually decreased below a dynamic path. Once the initial set of buckets is generated and also the massive rules are known and preloaded, the switch is prepared to serve incoming traffic. ACME then collects the matching pattern from each the CAB controller and the switch and adjusts the buckets to accommodate the dynamic path and maintain a decreased flow table usage at the switch. The matching pattern contains the traffic info relating to that rules and buckets are cached and that rules are matched, which provides ACME with traffic-awareness. ACME generates the initial bucket set based mostly solely on the rule set as a result of the system at the start has no historical traffic data. To accommodate any attainable incoming traffic pattern, ACME bounds the utmost variety of flow entries (i.e., bucket and rules) that are put in or replaced within the switch for somebody flow setup question. This avoids fierce churns on the flow table usage. Therefore, the initial bucket generation needs every bucket to contain no over N associated rules, wherever N may be a pre-determined parameter. According to CAB's rule caching policy, the utmost variety of entries that are put in or replaced within the switch for any individual question is finite by N + one.

ACME maintains the bucket set through a decision-tree based structure named bucket tree. The initial bucket set is generated by repeatedly partitioning the sector area into smaller hyper-rectangles. The bucket tree records the method
to partition the whole field area into smaller hyper-rectangles, and finally into buckets.

II. RELATED WORK

Sanghyeon Baeg [1] 2008, Power consumption is that the most crucial issues in match lines styles for low-power ternary content-addressable memory. Within the projected match-line design, the match line gift in every TCAM word is divided into four segments and is by selection pre-charged to cut back the match-line power consumption. The match lines that are partly charged are evaluated to work out the ultimate comparison result by sharing the fees deposited in numerous elements of the divided segments.

B. Heller et al, [2] 2010, Built ElasticTree, introduces energy proportion in today's wide traffic which is there in data-center. They're going to seemingly primarily decrease this quickly developing vitality price. Compare multiple ways for locating minimum-power network [20]. Framework is vitality proficiency and best execution. System worked close to its ability can build the likelihood of born and deferred bundles.

A.R. William Curtis et al, [3] 2011, DevoFlow proposition permits directors to focus on simply the streams that issue for his or her administration issue. DevoFlow handles most miniaturized scale streams within the data plane, and consequently permits United States of America to create the foremost out of switch re-sources. DevoFlow takes care of issue by allowing a clonable playing card principle to settle on a yield port. Multipath steering to statically stack balance movement with no utilization of the control-plane. These procedures do not spare a lot of vitality on elite systems.

P. Porraset al, [4] 2012, Incorporates many vital parts that are necessary for enabling security applications in Open Flow networks as well as role-based authorization, rule reduction, conflict analysis and policy synchronization. FortNOX may be a vital initial introduce enhancing the protection of Open Flow systems. It shows the attainableness and quality of our nom Delaware plume set guideline decrease approach [18]. It unable to handle dynamic matching method.

Zahid Ullah et al, [5] 2012, Hybrid divided static random could also be a memory style throughout that access memory-based ternary content out there memory (HP SRAM-based TCAM), that involves TCAM usefulness with normal SRAM, where we tend to tend to square measure eliminating the transmitted disadvantages of normal TCAMs. Electromagnetic unit SRAM-based TCAM could also be a way throughout that they logically dissects normal TCAM table throughout a hybrid approach.

H. Kim and N. Feamster et al, [6] 2013, Designed and enforced Procera, associate event-driven network management framework supported SDN. to boot utilize the OpenFlow convention to impart between the Procera controller and also the hidden system switches. It provides higher perceivability and command over undertakings for acting system. This SDN will improve common network management tasks [19]. Procera experiences the characteristic deferral caused by the communication of the management plane and also the info plane.

M. Yu, L. Jose et al, [7] 2013, OpenSketch empowers an easy and good approach to collect estimation data. It utilizes data plane estimation natives hooked in to ware switches Associate in nursingd a variable management plane therefore directors will while not

©IJRASET: All Rights are Reserved

abundant of a stretch execute variable estimation calculations. It's easy, economical thanks to management switches [16]. Sketches additional versatile in supporting varied mensuration tasks. Delay of every mensuration pipeline element is massive.

Weirong Jiang et al, [8] 2013, Random access memory i.e. (RAM)-based Ternary Content available Memory i.e.(TCAM) design is style for economical implementation on progressive FPGAs. We have a tendency to provides a formal study on RAM-based TCAM to disclose the concepts and also the algorithms behind it. To face the temporal order challenge, we have a tendency to propose a standard design consisting of arrays of small-size RAM-based TCAM units.

Jacobson et al, [9] 2014, Novel management plane design referred to as OpenNF that addresses these challenges through careful API style. OpenNF allows applications to decide on affordable selections in meeting their destinations. NF software package is often Up-to-Date. System has High performance on network observance.

M. Moshref et al, [10] 2014, DREAM permits operators Associate in Nursingd cloud tenants to flexibly specify their measure tasks in an extremely network, and dynamically allocates TCAM resources to those tasks supported the resource-accuracy. User-specified high level of accuracy. DREAM can support plenty of synchronized tasks. DREAM needs to dismiss nearly 0.5 the assignments, and drop concerning tenth.

N. Katta et al, [11] 2014, CacheFlow system could also be a system that "caches" the foremost in vogue rules at intervals the insufficient TCAM, at intervals that they want forward to package to handle the little amount of "cache miss" traffic. But, we've got a bent to can't blindly apply existing cache-replacement algorithms, because of dependencies between rules with overlapping patterns.

Naga Katta et al, [12] 2014, Instead of creating long dependency chains to cache smaller groups of rules inside that linguistics of the network policy square measure preserve. There square measure main four varieties criteria for it. Property that mixes the foremost effective of hardware and software package switches. Fine-grained rule caching that places common rules inside the TCAM, despite dependencies on less-popular rules. Ability that alter progressive changes to the rule caching as a result of the policy changes.

Bo Yan, Yang Xu, Hongya et al, [13] 2014, CAB is style to partition the sphere area into logical structures referred to as as buckets. That buckets are carries with it all the associated rules. By exploitation CAB, we have a tendency to resolve the rule dependency downside together with little storage overhead. whereas as per examination , CAB reduces the low setup requests by creating use of order of magnitude, once more saves management information measure by a 0.5, and considerably scale back average ow setup time.

R. Wei et al. [14] 2016, Propose a brand new technique known as Block Permutation (BP) to scale back the quantity of TCAM entries needed to represent a classifier. The BP procedure altogether enhances the pressure rate true being what it's [17]. They pack the parcel grouping rules place away in TCAMs. TCAM will likewise be connected to alternative instrumentality usage primarily based applications.

J. Sheu, and Y. Chuo et al. [15] 2016, we utilize cover-set technique to unravel the rule dependency drawback and propose a wildcard-rule caching algorithmic rule to cache rules into TCAM. In step with the simulation results, our wildcard-rule caching algorithmic rule and cache replacement algorithm have higher performance of caching magnitude relation and hit ratio, severally.

III. PROPOSED ALGORITHM

A. Description of the Proposed Algorithm

1) Bucket Generation

a) In this module, the initial bucket is generated supported the rule set. It needs every bucket to contain no quite N associated rules, wherever N may be a pre-determined parameter. in step with CAB's rule caching policy, the utmost range of entries that are put in or replaced within the switch for any person question is finite by N + one.

b) The initial bucket set is generated by repeatedly partitioning the sector house into smaller hyper-rectangles. The bucket tree records the method to partition the complete field house into smaller hyper-rectangles, and eventually into buckets.

2) Rule Preloading

a) This module preloads the large rules and excludes them from the reactive rule caching process. A rule to be 'larger' when it is associated with a larger number of buckets. By sorting the rules by the number of associated buckets, the top *K* rules are identified as large rules to be preloaded.

©IJRASET: All Rights are Reserved

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

b) Memory reserved for rule preloading or the setting of *K* affects cache performance. With more rules being preloaded, the control bandwidth usage by rule caching tends to be smaller. However, less flow table space is available for rules to be cached on demand, and flow table overload is more likely to happen.
c) If we reserve less memory for rule preloading, the system reacts better to bursty flow arrivals, while the control channel load tends to be higher. Essentially the choice of *K* reflects a trade-off between the tolerance of traffic burstiness and the control channel load.

3) Bucket Adjustment
a) This module adjusts buckets with traffic-awareness that saves table memory in 2 ways that. Victimization Optimized buckets that reduces the quantity of buckets that require to be cached.
b) The quantity of rules being cached however not matched by any packets is a smaller amount within the optimized case. Each bucket adjustment may be a series of merging operations that is followed by variety of cacophonous operations.

4) Rule Cache and Replacement
a) In the switch, each bucket is cached as a wildcard flow entry and can be stored in a bucket filter, while the rules are kept in the rule table. The caching is triggered when the switch receives Flow-Mod messages carrying corresponding buckets or rules.
b) Evict inactive entries and save memory for new entries. The timeout mechanism is used to evict inactive entries. Whenever a bucket is installed, the CAB controller assigns the same hard timeouts for both the bucket and its associated rules.

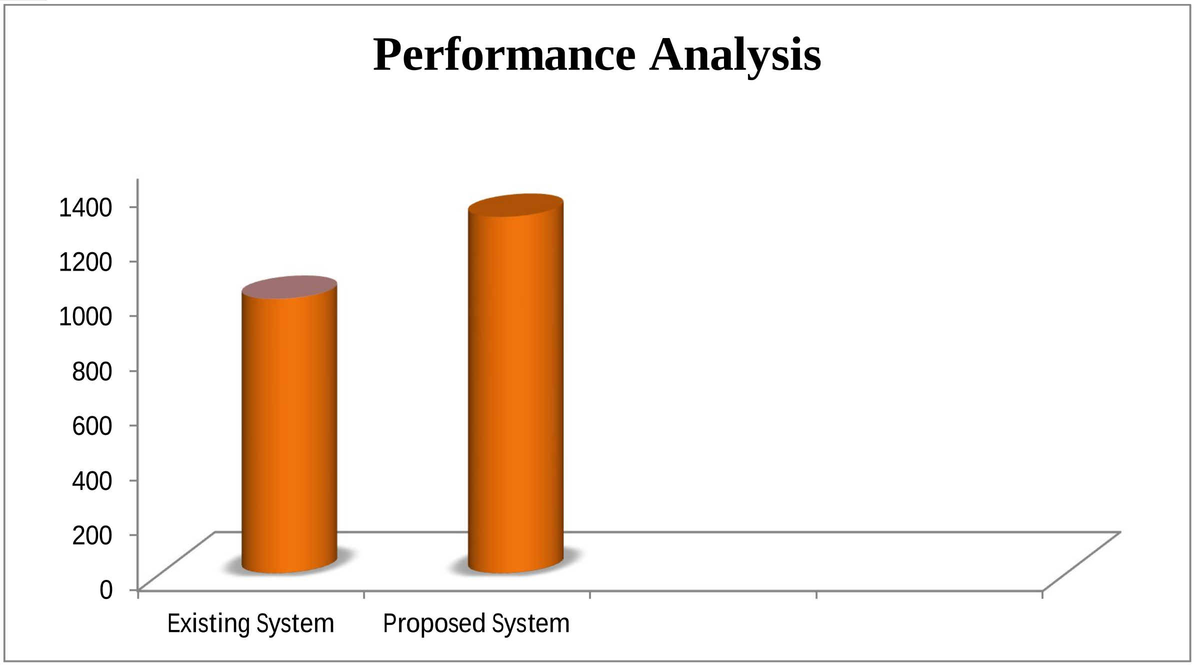
B. PSEUDO Code
1) Bucket Generation
a) Input: Rule Set R
b) Output: Initial Bucket Set B
c) Steps
i) Partition a hyper-rectangle node into no of child nodes
ii) for each node in hyper-rectangle
iii) Check child node < N
iv) Add node as bucket
v) Otherwise
vi) Partition the generated child nodes.
2) Optimal Bucket Generation
a) Input: Bucket Set B
b) Output: Optimal Bucket B'
c) Steps
i) Perform merging operation until No positive merging gain
ii) Post-Order Traversal
iii) Accept merging child node
iv) Update optimal bucket if necessary
v) Perform splitting operation until No positive splitting gain
vi) Pre-Order Traversal
vii) Compute partition gain
viii) Accept Split
ix) Update optimal bucket if necessary

IV. SIMULATION RESULTS

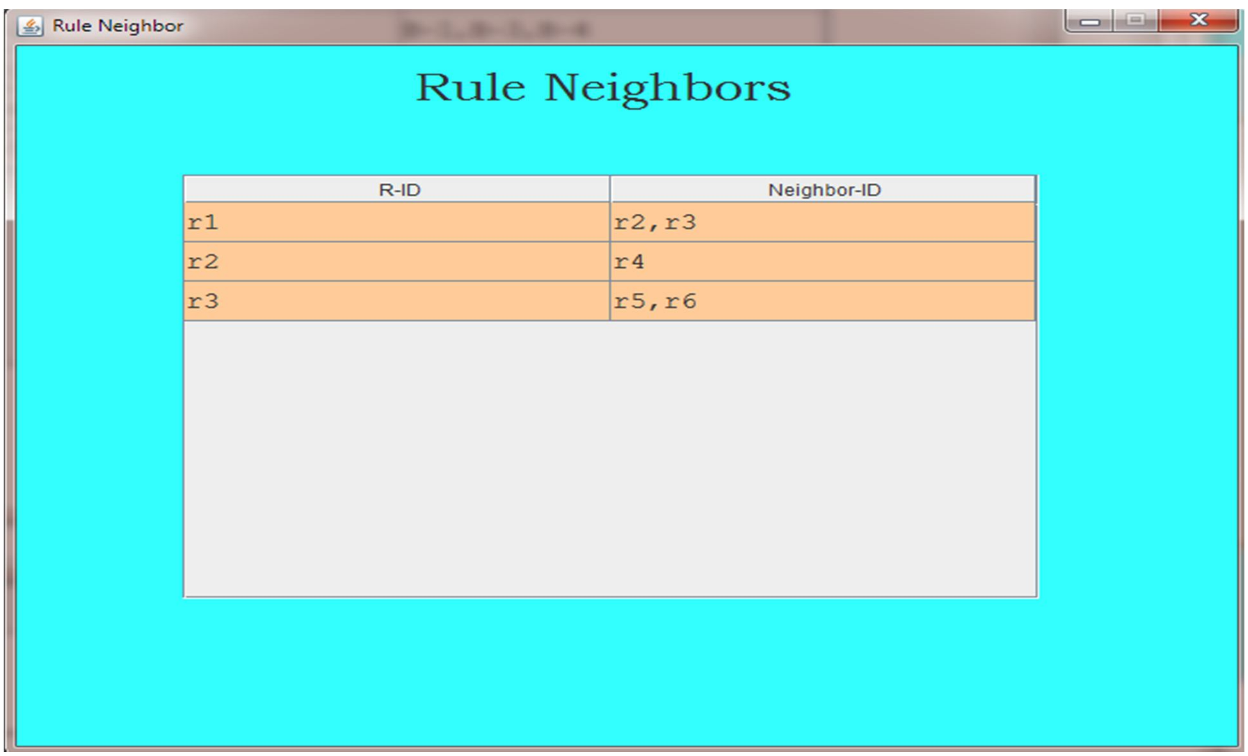
Wildcard-rule caching proposed a novel trump card rule storing calculation and a reserve substitution calculation to make utilization of TCAM space effectively. TCAM can look into a bundle's header and think about the coordinating examples of the parcel to the match field of all standards in the stream table in parallel. These special case rule storing calculation continue reserving an arrangement of imperative standards into TCAM until there is no TCAM space. This reserve substitution calculation takes fleeting and spatial activity territories into thought, which could make hit proportion high. Besides, the proposed reserve substitution calculation could have higher hit proportion than the other existing calculations.

©IJRASET: All Rights are Reserved

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com



The above graph shows the performance analysis of the existing and proposed system.



Rule dependency problem is solved in proposed system efficient Rule ID is calculated with Neighbor-ID.

V. CONCLUSION AND FUTURE WORK

Wildcard-rule caching is Associate in nursing economical technique to handle TCAM capability downside. Though there's rule dependency downside in wildcard-rule caching, we will utilize cover-set methodology to unravel this downside. Our k-HNS formula will cache a collection of rules into TCAM and has higher capability to make the set with higher total weight for every rule. On the opposite hand, our NSR formula each considers temporal and spacial localities that build the cache hit magnitude relation high.

©IJRASET: All Rights are Reserved

REFERENCES

[1] Low-Power Ternary Content-Addressable Memory Design Using a Segmented Match Line, Sanghyeon Baeg, IEEE Transactions On Circuits And Systems—I: Regular Papers, Vol. 55, No. 6, July 2008.

[2] Elastictree: Saving energy in data center networks, B. Heller et al., in Proc. NSDI, vol. 3. 2010, pp. 19–21.

[3] DevoFlow: Scaling flow management for high performance networks, A.R. Curtis, Comput. Commun. Rev., vol. 41, no. 4, pp. 254–265, Aug. 2011.

[4] A security enforcement kernel for OpenFlow networks, P. Porraset al, in Proc. 1st Workshop Hot Topics Softw. Defined Netw., 2012, pp. 121–126.

[5] Hybrid Partitioned SRAM-Based Ternary Content Addressable Memory, Zahid Ullah, Kim Ilgon, and Sanghyeon Baeg, IEEE Transactions On Circuits And Systems—I: Regular Papers, Vol. 59, No. 12, December 2012.

[6] Improving network management with software defined networking, H. Kim and N. Feamster, IEEE Commun. Mag., vol. 51, no. 2, pp. 114–119, Feb. 2013.

[7] Software defined traffic measurement with OpenSketch, M. Yu, L. Jose, and R. Miao, in Proc. NSDI, vol. 13. 2013, pp. 29–42

[8] Scalable Ternary Content Addressable Memory Implementation Using FPGAs, Weirong Jiang, IEEE Architectures for Networking and Communications Systems December 2013.

[9] OpenNF: Enabling innovation in network function control, A. Gember - Jacobson et al, in Proc. ACM Conf. SIGCOMM, 2014, pp. 163–174.

[10] DREAM: Dynamic resource allocation for software-defined measurement, M. Moshref, M. Yu, R. Govindan, and A. Vahdat, in Proc. ACMConf. SIGCOMM, 2014, pp. 419–430.

[11] “Rule-Caching Algorithms for Software-Defined Networks,” N. Katta, O. Alipourfard, J. Rexford, and D. Walker, available online at https://www.cs.princeton.edu/~jrex/ papers/cacheflow-long14.pdf, 2014.

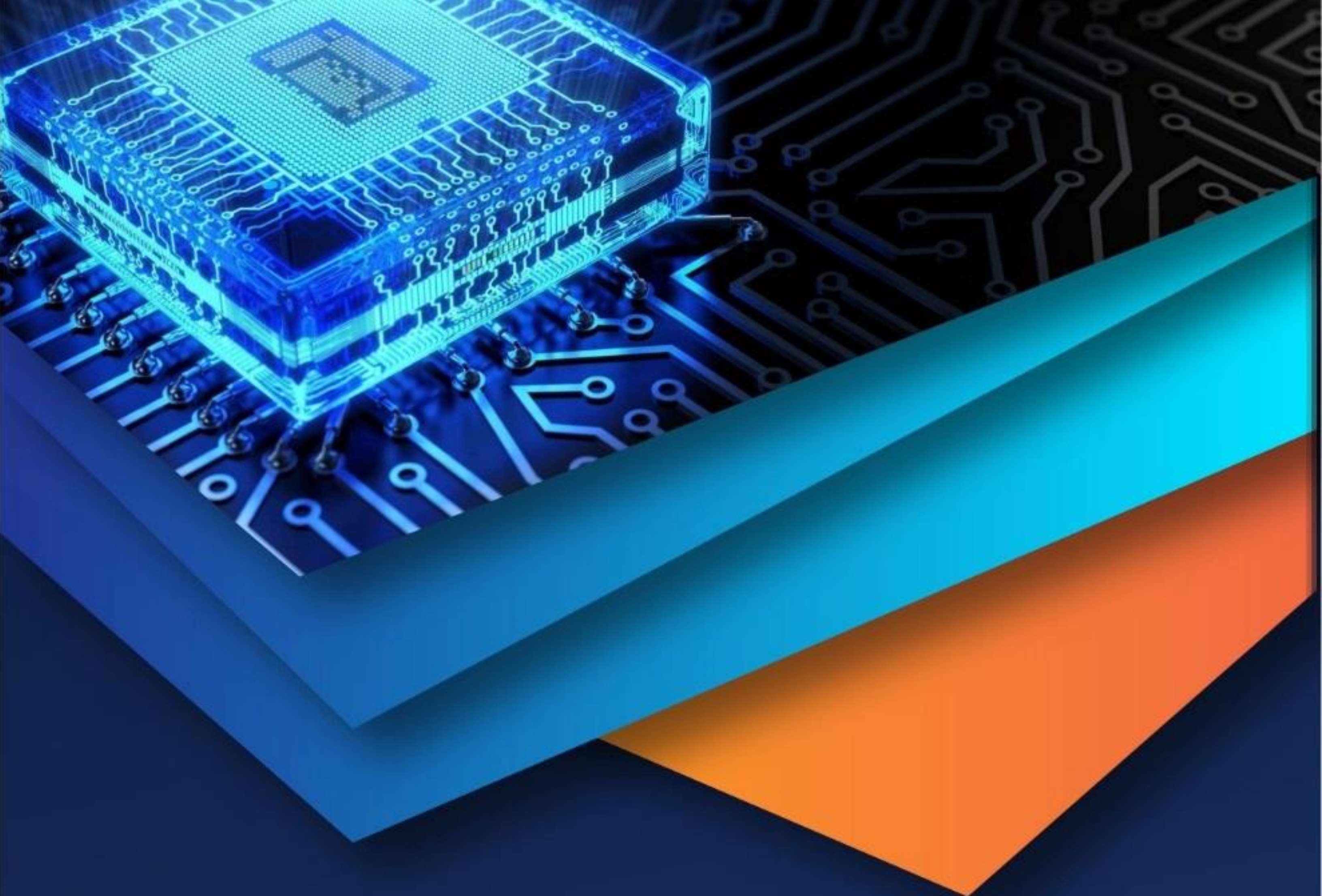
[12] Infinite CacheFlow in Software-Defined Networks, Naga Katta, Omid Alipourfard, Jennifer Rexford and David Walker, ISBN: 978-1-4503-2989-7, August 2014.

[13] CAB: A Reactive Wildcard Rule Caching System for Software-Defined Networks, Bo Yan, Yang Xu, Hongya Xing, Kang Xi, H. Jonathan Chao, Association for Computing Machinery, August 2014.

[14] Finding non-equivalent classifiers in Boolean space to reduce TCAM usage, R. Wei, Y. Xu, and H. J. Chao, IEEE/ACM Trans. New., vol. 24, no. 2, pp. 968–981, Apr. 2016.

[15] Wildcard Rules Caching and Cache Replacement algorithm in Software-Defined Networking, J. Sheu, and Y. Chuo, IEEE Trans. On Networking and Service Management, pp. 19-29, Hsinchu, Taiwan, March 2016.

©IJRASET: All Rights are Reserved



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)