Framework for Network Traffic Identification by Using Network Intrusion Detection and Prevention Systems

P. Ravali[1], Panduri Bujji Babu[2], M. Madhavi Latha[3], M. Pradeep[4]

Asst. Prof, CSE Department, Princeton Inst. of Engg and Technology for Women, Hyderabad[1]

CSE Dept. St. Mary's Women's Engineering College, Guntur[2]

Asst. Professor, CSE Dept. Princeton Institute of Engineering & Technology for Women. Hyderabad[3]

Asst. professor, CSE dept. Mallareddy College of Engineering for Women, Hyderabad[4]

Abstract: This paper presents an investigation, involving experiments, which shows that current network intrusion, detection, and prevention systems (NIDPSs) have several shortcomings in detecting or preventing rising unwanted traffic and have several threats in high-speed environments. Precise organization traffic recognizable proof is a significant reason for network traffic checking and information investigation, and is the way to work on the nature of client administration. In this paper, through the examination of two organization traffic ID strategies in light of machine learning and profound parcel review, an organization traffic distinguishing proof strategy in view of machine learning and profound bundle examination is proposed. This strategy utilizes profound parcel assessment innovation to distinguish most organization traffic, diminishes the responsibility that should be recognized by machine learning. This paper presents an investigation, involving experiments, which shows that current network intrusion, detection, and prevention systems (NIDPSs) have several shortcomings in detecting or preventing rising unwanted traffic and have several threats in high-speed environments. It shows that the NIDPS performance can be weak in the face of high-speed and high-load malicious traffic in terms of packet drops, outstanding packets without analysis, and failing to detect/prevent unwanted traffic. A novel quality of service (QoS) architecture has been designed to increase the intrusion detection and prevention performance. Our exploration has proposed and assessed an answer involving an original QoS setup in a multi-facet change to sort out parcels/traffic and equal procedures to build the bundle handling speed. The new engineering was tried under various traffic velocities, types, and errands. The trial results show that the design works on the organization and security execution which is can conceal to 8 Gb/s with 0 bundles dropped. This paper likewise shows that this number (8Gb/s) can be improved, yet it relies upon the framework limit which is constantly restricted.

Keywords: Intrusion detection, traffic identification, MDIP, network security, open source, quality of service, security.

I. INTRODUCTION

With the quick advancement of organization innovation, network clients are requesting increasingly high speed and nature of organization administrations. Hence, it has become one of the difficulties in the field of organization activity and support the executives to oversee and control different organization business traffic through compelling specialized implies, recognize various administrations, give different quality confirmation, and meet clients' business needs. Network traffic recognizable proof gives a successful specialized means to recognize traffic of various applications. By characterizing, distinguishing and separating the use of organization traffic, the traffic of various applications can be partitioned to furnish clients with customized network benefits and further develop the organization administration quality and client fulfillment.

Data innovation (IT) influences pretty much every part of current life. Today, different gadgets are accessible to meet clients' necessities, for example, high machine processor speed, and quick organizations. Close by our rising reliance on IT, there has tragically been an ascent in security episodes. Dangers and assaults might go from taking individual data from a PC or organization server to taking the most highly classified data put away on a Security Insight Administration (Sister). Besides, programmers can sneak around on clients' internet based buys by snooping on their Visa subtleties, or, significantly more alarmingly, wellbeing basic frameworks can be compromised. Complex assaults and dangers have made the execution of safety frameworks really testing. Programmers have advanced alongside the complexity of the IT business. For instance, programmers exploit the improvements in PC processors and organization velocities to expand the volume and speed of vindictive traffic that could comprise a Disavowal of Administration (DoS) or Dispersed Refusal

DOI: 10.17148/IJARCCE.2023.12409

of Administration (DDoS) assault [1]. Network security is in this manner critical and has formed into an industry pointed toward further developing applications and equipment stages to recognize and stop network dangers. Perhaps of the most settled idea in data security is a guard top to bottom methodology which uses a multifaceted underlying model, in which rewalls, weakness evaluation devices (against infections and worms), and IDPS (Interruption Recognition and Counteraction Frameworks) are utilized to forestall any unfriendly undertakings on network frameworks and servers. The Organization Interruption Discovery and Anticipation Framework (NIDPS) has been intended to act as the last place of safeguard in the organization design.

Pioneers have made equipment IDPS to handle a large number of bundles simultaneously [10], [11], however there are restrictions in the capacity to perform specific programming undertakings. What's more, restricted memory size is an issue for equipment based NIDPS arrangements. Moreover, equipment based NIDPS offer a high scope of handling speeds yet are exorbitant. Programming arrangements are famous on the grounds that they are less expensive and offer more exibility than equipment arrangements. This paper centers around open-source programming arrangements. PC organization and web security face expanding difficulties and many organizations depend on NIDPS to get their information sources and frameworks. The need to guarantee that the NIDPS can stay aware of the rising requests because of expanded network utilization, higher speed organizations and expanded noxious movement, makes this a fascinating area of examination and inspired this review.

II. RELATED WORK

Network traffic identification refers to the identification of bidirectional TCP or UDP flows generated by network communication according to the types of network applications (WWW, FTP, P2P, etc.) in the Internet based on TCP/IP protocol[1]. At present, there are three commonly used methods: Identification based on port matching; Identification based on deep packet detection(DPI); Identification based on machine learning method.

Table 1 Searching Result of the Key Words.

Name of search engine	Key word searches conducted	Results of search Subhead
Google Scholar	Traffic recognition with neural networks	32,500
Google Scholar	Network traffic recognition neural	31,700
Google Scholar	Network traffic classification recognition neural network	19,100
Google Scholar	Supervised network traffic classification recognition neural network	4,560
Google Scholar	Supervised network traffic classification recognition neural network after 2005	1,870

Network traffic identification refers to the identification of bidirectional TCP or UDP flows generated by network communication according to the types of network applications (WWW, FTP, P2P, etc.) in the Internet based on TCP/IP protocol [1]. At present, there are three commonly used methods: Identification based on port matching; Identification based on deep packet detection (DPI); Identification based on machine learning method.

Identification Method based on Port Matching: The distinguishing proof technique in light of port matching is to group the application types that are not utilized by the non-obligatory port number suggested by the IANA (The Web Relegated Numbers Authority) [2]. In the beginning phase of Web advancement, HTTP, FTP and different applications can be recognized by port numbers like 80 and 21. At the point when P2P applications were simply arising, BitTorrent and other P2P applications can likewise be recognized by ports, for example, 6346-6347 and 6881-6889. Be that as it may, with the rise of countless new organization administrations and the top to bottom improvement of P2P innovation, an enormous number of arising applications start to involve port cover and dynamic port innovation to cross firewalls or keep away from other obstructing techniques. Hence, the port-based distinguishing proof strategy has extraordinary restrictions, and the ID results are exceptionally wrong.

Identification Method based on Deep Packet Inspection: Every application has its own different component fields, which can be explicit strings or spot successions. The DPI (profound bundle examination) innovation identifies the heap content of IP parcel during network association or information transmission as indicated by the strategy for design coordinating, and decides the kind of use as per different burden satisfied identification innovation can rapidly distinguish the application sort of organization stream and isn't impacted by port changes. Sen et al. [3] concentrated on the acknowledgment of P2P streams in view of the attributes of payload in information bundles, and checked the exactness vigor and continuous execution of the strategy.

Identification Method based on Machine Learning: Because of various application conventions, network information stream has various attributes as far as information stream term, parcel length, bundle transmission recurrence and bundle rate. As per these attributes of organization stream, the ID innovation in information mining can be utilized to accomplish great traffic distinguishing proof through AI. Bayesian recognizable proof [4], support vector machine (SVM)[5], C4.5[6] and other AI calculations in light of stream measurable attributes have been brought into the use of organization traffic ID.

III. COMPARISON OF NETWORK TRAFFIC IDENTIFICATION METHODS

First and foremost, the port-based traffic distinguishing proof technique doesn't require confounded estimation and examination, and its execution guideline is straightforward. It can meet the necessities of quick distinguishing proof of fast organization. Notwithstanding, because of the improvement of new organization applications, particularly the rise of P2P applications, the vast majority of them utilize irregular ports and disguise ports to safeguard their organization interchanges, which lessens the exactness of port-based traffic ID technique, which has been steadily dispensed with by history.

The recognizable proof strategy in view of component field can distinguish the powerful fields in the heap without depending on the port Settings of the application. It can well distinguish network streams and explicit organization applications, and the location precision is high. This strategy can distinguish network traffic rapidly by just recognizing the initial not many explicit parcels of organization traffic. In any case, since this strategy relies upon the component field of the application convention, it can perceive known applications and can't perceive new applications. Furthermore, this technique can't recognize the organization traffic of burden encryption.

The machine learning identification methods technique in light of the stream measurements highlights involves the identification methods innovation in information mining to acknowledge traffic identification methods through the machine learning strategy, which defeats the troubles that can't be addressed by the initial two techniques, is liberated from the impact of port changes and convention highlight changes, and can recognize new applications. In any case, this sort of technique in view of machine learning in both Bayesian distinguishing proof in light of SVM (support vector machine) identification method strategy, can't recognize explicit application, need, contingent upon the kind of various parcel stream to recognize traffic location somewhat falls behind, and effectively impacted by stream length, with under a specific long stream the misdiagnosis rate is high. Furthermore, the precision of this identification methods strategy is handily impacted by powerful organization changes and traffic property set, and the inconvenience of this sort of technique is that it is computationally serious and not reasonable for ongoing traffic recognizable proof of high velocity organization. In light of the examination and correlation of the above traffic distinguishing proof strategies, an organization traffic identification methods strategy in view of machine learning and DPI innovation is proposed by the rule of component field based ID technique and stream measurements based machine learning strategy.

IV. PREVENTING MALICIOUS PACKETS

In this experiment, TCP/IP food traffic was sent at differing speeds (see Table 2) with 255 malicious UDP packets (threads) also sent at 1 microsecond (1 mSec) intervals. Snort was set to prevent UDP threads by using two rule conditions (TTL and content) as follows: reject udp any any ->any any (msg: ``Prevent Malicious UDP Packets"; ttl: 120; content:j' C2 48 60 AE 97 4F 4B C3 'j; Sid: 100007;). Use of these options will prevent any UDP malicious packet that is matched with the TTL value equal to 120 and a data pattern inside the malicious packet with content ``.H`..OK.". The hexadecimal number (`C2, 48, 60, AE, 97, 4F, 4B, C3'), which the rule contained, is equal to the ASCII characters (`., H0,,.,., O, K,.'). As shown in Table 2, When 255 malicious UDP packets were sent at a speed of 1 mSec and TCP/IP food traffic at 100 bytes per second (Bps), Snort prevented 100% of the total UDP packets that it analyzed. As the food traffic (speed) was increased to 10000 bytes per second (10000Bps), Snort prevented less than 51% of the total malicious packets analyzed (see Table 2). The number of missed malicious packets increased when the speed increased. The experiment shows that, when the speed was 60000 Bps, Snort only prevented less than 18% of 100% of the malicious packets analyzed (see Table 2).

Table 2. Snort-NIDPS reaction to prevent malicious packets.

Flood traffic(Bps) with 255 udp malicious packets in (1mSec)	Number of packets analsed	Eth packets received of packets analysed	Ip4 analysed of Eth packets analysed	ICMP packets analysed	TCP packets analyzed	UDP malicious packets analysed	UDP malicious packets reject	%malicious packets prevent
100 Bps	267032	100.00%	89.066%	28	995	236795	236795	100.00%
1000 Bps	266863	100.00%	99.991%	7	3572	263260	26320	100.00%
10000 Bps	329926	100.00%	99.988%	522	114260	215104	108107	50.258%
60000 Bps	335143	100.00%	99.992%	784	147518	186814	186814	17.564%

V. MACHINE LEARNING AND DPI TECHNICAL IDENTIFICATION METHODS

DPI technology

DPI innovation is a sort of innovation in light of element field recognition. By profoundly perusing the IP parcel load content and redesigning the application layer data, the substance of the whole application layer is gotten. Then, the information stream content is checked and distinguished by the current element library, in order to recognize the particular application information. Profound bundle assessment requires the hardware to have the option to rapidly examine, identify and rearrange application information to keep away from extreme deferral to the application.

DPI innovation by and large comprises of two sections, one is examining calculation, the other is highlight library. The examining calculation is to match the substance and element library of IP bundle load word by word. The string matching calculation normally utilized in DPI innovation is AC calculation, WUMANBER calculation and SBOM calculation [7]. DPI identification is like component matching in enemy of infection programming. The antivirus programming matches the filtered current document to its own infection library word by word. Assuming a similar trademark code is found, the sort and name of the infection not set in stone.

This approach can precisely recognize network streams in light of the element library, and can be exact to the particular application to which the organization streams have a place, with high location exactness. In any case, DPI can't distinguish application traffic that has not yet been kept in the element library, lingers behind the arrival of new applications, and doesn't perceive scrambled network information streams.

Machine learning identification methods

The center of the organization traffic distinguishing proof strategy in view of AI is that PC projects can continually work on their presentation with the amassing of opportunity for growth, in order to finish responsibilities that can't be finished by customary techniques. In network traffic recognizable proof, this sort of earlier information can be various qualities of organization traffic and administrative data of individuals. Choosing proper AI calculation can really take advantage of earlier information to finish traffic identification method. The progression of organization traffic ID technique in light of AI is displayed in "Fig. 1".Firstly, the preparation informational index is utilized to prepare the distinguishing proof model, and afterward the recognizer is laid out as indicated by the preparation model. After the recognizer is laid out, the ID of traffic can be understood.

DOI: 10.17148/IJARCCE.2023.12409

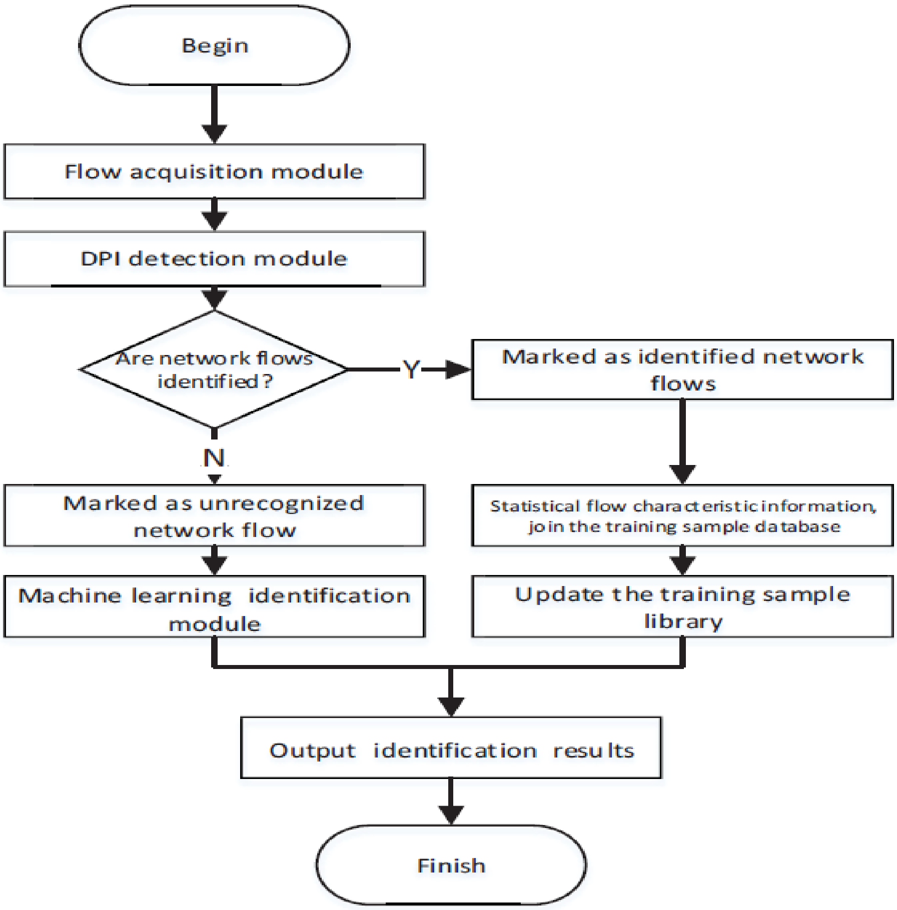
Begin
Flow acquisition module
DPI detection module
Are network flows identified?
Y
Marked as identified network flows
N
Marked as unrecognized network flow
Statistical flow characteristic information, join the training sample database
Machine learning identification module
Update the training sample library
Output identification results
Finish

Fig. 1. Network traffic identification process based on DPI and machine learning

VI. NOVEL NIDPS ARCHITECTURE

Critical analyses were done for the experiments presented in sections II(A) and II(B). The figure show that performance of NIDPS throughput is affected when NIDPS is exposed to a high-volume and speed of traffic; more packets will be dropped and left outstanding as the speed of traffic increases. Figure 1 shows that the NIDPS's detection performance decreased when the traffic speed increased. There were more missed alerts and missed logs for packets as the speed of traffic increased. Figure 1 shows that the NIDPS prevention performance decreased when traffic speed increased. When traffic moves through the network interface card (NIC) to the NIDPS node, the packets are stored in the buffer until the other relevant packets have completed transmission to processing nodes. In the event of high-speed and heavy traffic in multiple directions, the buffer will fill up. Then packets may be dropped or left outstanding [15].

In this case, there is no security concern about the packets dropped; the packets are dropped outside the system. The existence of outstanding packets that are waiting or have not been processed by a security system (i.e. NIDPS node) affects the system efficiency however. Packets can also be lost in a host-based IDPS. Most software tools use a computer program such as the kernel, which manages input/output (I/O) requests from software and decodes the requests into instructions to direct the CPU's data processing. When traffic moves from the interface (NIC) through the kernel's buffer to the processor space, where most of processing nodes are executed, the packets will be held in the kernel buffer before being processed by the CPU. When some nodes experience a high-volume of data, the buffer will fill up and packets may be dropped. Implementing QoS methods, such as queueing, memory reservation, congestion-management, and congestion avoidance techniques, can yield preferential treatment to priority traffic according to its relative importance. Furthermore, QoS technology ensures that network performance is more predictable, and that bandwidth utilization is more effective. QoS is used to improve performance in high speed network events. QoS can be configured on physical interfaces such as ports and switch virtual interfaces (SVI) .

DOI: 10.17148/IJARCCE.2023.12409

In our review, QoS has been utilized to design a clever engineering to further develop generally speaking organization traffic and security execution. The framework (switch) interface has been arranged to have two info lines and four result lines. The lines' boundaries were summoned to permit lines to deal with traffic collectively of bytes. These heap a bunch of bundles similarly among the lines and separation traffic into equal streams to expand the pace of parcel handling. The framework then, at that point, utilizes equal NIDPS hubs to expand the NIDPS throughput execution and investigations each departure line independently to decide if it is liberated from vindictive codes. A class map and a strategy map were made for each info line. The class map perceives and classes a particular kind of traffic for each info line, while the strategy map controls and coordinates as far as possible for each info line and applies the breaking point to all points of interaction. The data transfer capacity, limit, support, memory reservation, and need (line and traffic) were designed for all entrance and departure lines to treat and control traffic to assist with forestalling clog or complete disappointment through over-burden. One line was arranged as a sped up line. It got focused on QoS administrations and different lines were not overhauled until the data transmission of focused on line arrived at its breaking point.

VII. WORKING MODEL FOR NOVEL ARCHITECTURE

NIDPSs process packets which are carried by IP protocols, e.g. UDP, TCP and ICMP. The IP protocols are checked by NIDPS rules based on a signature database (known signature/ attacks). however, to get the best NIDPS performance, the NIDPS should be implemented in a system which can manage the layer 3 network protocol (IP layer). In our study, a layer 3 switch has been used to support and improve NIDPS performance. The switch supports QoS configuration as well as Differentiated services (DiffServ) architecture.

The parallelization of data (traffic) that was distributed through ingress and egress queues into critical and noncritical is viewed as multiple traffic parallelism (MTP). Critical pre-processing of traffic is performed on queues to create particular groups of packets (threads) before the traffic is examined by an ingress queue algorithm. Non-critical preprocessing occurred after the packets had been matched to ingress and egress queues policies. The NIDPS node component can be parallelized in an either non-functional or functional manner. Component level parallelism is dened as function parallelism of the NIDPS processing node. Individual components of NIDPS were isolated, and each output queue was given its own processing element. The NIDPS node was configured from a single node NIDPS to a multi-node NIDPS. Each node was conjured to check for a certain type of packet (e.g. UDP, TCP and ICMP) and was able to access discrete parts of a centralized, common rule base to order to carry out its task. he kernel buffer parameters for each NIDPS node was configured as each output queue rate.

When traffic arrives at the ingress interface of system, packets will be classified through a class map that will enable packets to be processed as a group of bytes defined by a policy and ACLs that were matched with DSCP values. A policy map was made to specify required action for each class. The following procedures constitute the method: Classify the traffic with a class map for SVI and ports. Set ACLs rules depending on the kind of traffic/attacks to be detected or prevented. In our experiment, we detect and also prevent UDP malicious packets which came with random high-speed traffic. We allowed UDP traffic to be processed in a separate egress interface (queue) and then analyzed by a parallel NIDPS node. The other traffic (e.g. TCP, ICMP, etc.) was processed in the other egress queues. Organize a rate-limit for the system ingress interface processing speed (Setting a set group of packets in bytes) for the class traffic. The rate depends on the maximum limit of SVI bandwidth including memory. In our system we set ``1.124 million" bytes (nearly 1Gb of packets) for the set of classes because the maximum limit for each interface in our system is 1Gbps.

VIII. EVALUATION OF NOVEL NIDPS ARCHITECTURE

The experiments that were described in section II are repeated, but here the novel architecture is implemented to test performance in terms of throughput with the support of the proposed solution (QoS and parallel technologies). Each experiment tested Snort NIDPS throughput when analyzing traffic such as TCP/IP headers and then detecting or preventing unwanted traffic (UDP malicious packets) arriving at a high-speed. As shown in Figure 1, when malicious UDP packets were sent at a speed of 1 mSec with different TCP/IP ood traffic at 16 to 60000 bytes per second (Bps), Snort NIDPS started effectively but overall it missed detecting up to 65% of malicious packets that system received (see Table 1). Furthermore, it was unable to prevent all unwanted packets. The experiment shows that, when the speed was 60000 Bps, Snort prevented less than 18% of the malicious packets analyzed. When QoS architecture was implemented, Snort NIDPS detected almost 100% of malicious packets that system received. The experiment results show that Snort NIDPS performance increased greatly when QoS is used. It prevented almost100% of malicious packets that it analyzed.

This section discusses the proposed solution and compares it to related research in parallelism in intrusion detection. Vasiliadis et al. [5] proposed a new model for a multiparallel IDS architecture (MIDeA) for high-performance processing

DOI: 10.17148/IJARCCE.2023.12409

and stateful analysis of network traffic. Their solution offers parallelism at a subcomponent level, with NICs, CPUs and GPUs doing specialized tasks to improve scalability and running time. They showed that processing speeds can reach up to 5.2Gbps with zero packet loss in a multi-processor system. Jiang et al. [6] proposed a parallel design for NIDS on a TILERAGX36 many-core processor. They explored data and pipeline parallelism and optimized the architecture by exploiting existing features of TILERAGX36 to break the bottlenecks in the parallel design. They designed a system for parallel network traffic processing by implementing a NIDS on the TILERAGX36, which has a 36 core processor.

Table 3. Snort-NIDP reaction to detect malicious packets.

Packet sent (TCP/IPFlood traffic(Bps) with 255 UDP malicious packets in (1mSec)	Eth packets received	Ip4 analyzed of Eth packets analyzed	ICMP packets analyzed	TCP packets analyzed	malicious UDP packets analyzed	malicious UDP packets Alerts	malicious UDP packets logged	%Malicious packets Alerts and logged
16 Bps	100%	99.174%	99	1680	999866	999866	999866	100.00%
32 Bps	100%	99.693%	105	4751	899338	894351	894351	99.44%
200 Bps	100%	99.899%	1511	200015	759092	757877	757877	99.84%
1200 Bps	100%	99.999%	1130	565025	433681	430081	430081	99.17%
4800 Bps	100%	98.376%	1003	799012	200995	199789	199789	99.40%
60000 Bps	100%	99.881%	1339	973755	27560	27491	27491	99.75%

The framework was planned by two techniques: initial a crossover equal design was utilized, consolidating information and pipeline parallelism; and furthermore a half breed load-adjusting plan was utilized. They exploited the parallelism presented by joining information, pipeline parallelism and various centers, utilizing both rule-set and ow space apportioning. They demonstrated the way that handling velocities can deal with and reach up to 13.5 Gbps for 512-bytes. Jamshed et al. [2] introduced the Kargus framework which takes advantage of high handling parallelism by adjusting the example coordinating jobs with multi-center computer processors and heterogeneous GPUs. Kargus adjusts its asset utilization relying upon the info rate, to save power.

The examination shows that Kargus handles up to 33 Gbps of ordinary traffic and accomplishes 9 to 10 Gbps in any event, when all bundles contain assault marks. The two methodologies depicted in this section are not straightforwardly practically identical as far as throughput as various quantities of processors are utilized in each. In any case, the tests demonstrate the way that high gains can be made by parallelizing NIDPSs to battle issues of higher paces and expanding traffic. Our examination utilizes a multi-facet switch alongside equal innovation to further develop parcels handling execution which builds the capacity to deal with various velocities and information volumes. Further improvements happen while lining is joined with equal processor advances. The methodology of this study has shown how parallelism at a more elevated level of granularity, which is easier to carry out, can likewise make noteworthy enhancements for security execution with regards to throughput and the quantity of dropped bundles. By utilizing 2 machines associated with two connection points, our NIDPS handled up to 8 Gbps with 0 drop for 1KB bundles. This number can be expanded up to 32Gbps which is the full framework limit forward transfer speed by carrying out additional hubs of NIDPS.

Seller organizations are intending to foster security answers for safeguard the undertaking organization. Gear has been intended to meet availability speed and burden guidelines. The upgrades in the throughput of NIDPS displayed in this examination are accomplished by matching the ASA (Versatile Security Apparatus) Cisco hardware [10] with numerous executions of Grunt. The standards of the technique proposed in this exploration could be applied to other hardware mixes where comparative offices are advertised. To sum up, our exploration varies from past examination as far as the engineering utilized. The examination researches what QoS including DiffServ innovation and parallelism can have mean for in fast and weighty rush hour gridlock networks utilizing an industry standard switch and standard work area processors. This arrangement is a more open approach to getting great outcomes as it very well may be enacted at a more elevated level, specifically at the degree of designing the CISCO switch programming and repeating Grunt on standard machines. Further enhancements could be made on the off chance that better execution gear was utilized. Cost is for the most part a significant concern. The plan proposed in this exploration helps the organization security prerequisites for minimal price.

Table 4: Novel NIDPS architecture reaction to prevent malicious packets

Flood traffic(Bps) with 255 UDP malicious packets in (1mSec)	Number of packets analyses	Eth packets received of packets analyzed	Ip4 analyzed of Eth packets analyzed	ICMP packets analyzed	TCP packets analyzed	UDP malicious packets analyzed	UDP malicious packets reject	%malicious packets prevent
100 Bps	1013836	100.00%	90.076%	228	262037	738999	738999	100.00%
1000 Bps	1502809	100.00%	99.891%	117	823198	678401	678400	99.999%
10000 Bps	1993125	100.00%	99.889%	522	1161022	830578	830578	100.00%
60000 Bps	2505935	100.00%	99.998%	384	1725830	779641	779641	100.00%

IX. EXPERIMENTAL RESULTS AND ANALYSIS

Flow identification algorithm in CentOS system implementation, using Wireshark capture data in the campus local area network, then carries on the processing, only keep BitTorrent, PPStream such type of P2P traffic, and belongs to the WWW HTTP traffic, finally the flow identification method based on DPI and traffic identification method based on this model to analyze traffic data.

Flow identification algorithm in CentOS system implementation, using Wireshark capture data in the campus local area network, then carries on the processing, only keep BitTorrent, PPStream such type of P2P traffic, and belongs to the WWW HTTP traffic, finally the flow identification method based on DPI and traffic identification method based on this model to analyze traffic data.

As shown in Table 1, the dpi-based identification method is significantly less sensitive to PPStream traffic than to BitTorrent traffic. This is because BitTorrent P2P file sharing software is open source and its protocol features can be easily found through analysis of its programs and application protocols. DPI technology can be used to identify the corresponding network traffic. For private commercial applications of PPStream, only the protocol features can be obtained by analyzing network packets and decompiling. The accuracy is limited to some extent, which leads to the reduction of identification recognition rate of these traffic. To identify the network flows through the machine learning method which cannot be recognized by DPI, the traffic of BitTorrent and PPStream is judged as P2P traffic, which makes up for the deficiency of DPI recognition. As shown in “table 2”, the traffic generated by BitTorrent and PPStream is judged to be P2P traffic. The identification method adopted in this study has significantly improved the identification of P2P traffic like BitTorrent and PPStream by combining machine learning algorithm and DPI technology to detect network traffic, thus improving the overall identification rate of network traffic.

Table 5. Traffic Identification Results Based on DPI Algorithm

Protocol name	Actual flow /Byte	Identified traffic /Byte	Traffic identification rate	Actual connection number	Number of connections identified	Connection identification rate
BitTorrent	1070660210	96894749	90.5%	255	240	94.1%
WWW	82945	80290	96.8%	20	20	100%
PPStream	36055060	26680744	74%	250	181	72.4%

Table 6. Traffic Identification Results Based on DPI Algorithm

Protocol name	Actual flow /Byte	Identified traffic /Byte	Traffic identification rate	Actual connection number	Number of connections identified	Connection identification rate
P2P	1180246025	109998929	93.2%	482	453	94.0%
WWW	82945	80290	96.8%	20	20	100%

DOI: 10.17148/IJARCCE.2023.12409

X. CONCLUSION

Another engineering for NIDPS sending was planned, executed and assessed. There has as of late been gigantic advancement in PC networks with respect to their capacity to deal with various paces and information volumes. Because of this quick turn of events, PC networks are currently more defenseless than any other time in recent memory to rapid assaults and dangers. These can bring significant hardship to PC organizations and frameworks. Network interruptions can be arranged at different levels. Some rapid assaults can be delegated being hard to identify or forestall. It will turn out to be perpetually challenging to examine expanding volumes of traffic because of the quick changes in innovation that are speeding up. As of late, different open-source apparatuses have opened up to cover security necessities for network frameworks and clients. In this paper, the presentation of an open source NIDPS has been assessed with regards to rapid and volume assaults. The motivation behind the assessment was to decide the exhibition of the NIDPS under rapid traffic when limited by off-the-rack equipment, and afterward track down ways of further developing it.

This study zeroed in on the shortcoming of such security frameworks, for example NIDPS in fast organization network. We proposed an answer for lessening this shortcoming and introduced an original engineering in NIDPS improvement that uses QoS and equal advancements to coordinate and further develop network the board and traffic handling execution to work on the exhibition of the NIDPS. With our clever engineering, Grunt's presentation improved uniquely, permitting more parcels to be checked before they were conveyed into the organization. The presentation (investigation, recognition and counteraction pace) of Grunt NIDPS expanded to over close to 100%. By utilizing 2 machines (computers) associated with two 1Gb points of interaction, Grunt NIDPS handled up to 8 Gbps with 0 drop. This number can be expanded up to 32Gbps which is the full framework limit forward transfer speed by executing more hubs of NIDPS. The exploration zeroed in on laying out a specialized arrangement with a hypothetical establishment. This data sums up the issue and arrangement and consequently empowers the proposed way to deal with be applied all the more effectively to foundations that are different to the proving ground utilized in this exploration.

REFERENCES

[1] M. D. Samani, M. Karamta, J. Bhatia, and M. B. Potdar, ``Intrusion detection system for DoS attack in cloud,'' International Journal of Applied Information Systems (Foundation of Computer Science), vol. 10, no. 5. New York, NY, USA: FCS, 2016.

[2] W. Bul'ajoul, A. James, and M. Pannu, ``Improving network intrusion detection system performance through quality-of-service configuration and parallel technology,'' J. Comput. Syst. Sci., vol. 81, no. 6, pp. 981999,2015.

[3] Anne James, "A New Architecture for Network Intrusion Detection and Prevention", VOLUME 7, 2019,IEEE Access.

[4] Ravindra Changala, "Intrusion Detection System Using Genetic Algorithm" published in International Journal of Emerging Trends in Engineering and Development [IJETED], Impact Factor 2.87, ISSN NO:2249-6149, Issue 2,Vol. 4 May 2012.

[5] W. Bul'ajoul, A. James, S. Shaikh, and M. Pannu, ``Using Cisco network components to improve NIDPS performance,'' Comput. Sci. Inf. Technol.,pp. 137-157, Aug. 2016.

[6] Y. Naouri, and R. Perlman, (2015). ``Network congestion management by packet circulation,'' U.S. Patent 8 989 017 B2, Mar. 24, 2015.

[7] M. K. Testicioglu and S. K. Keith, ``Method for prioritizing network packets at high bandwidth speeds,'' U.S. Patent 15 804 940, Nov. 6, 2017.

[8] Ravindra Changala, "Diminution of Deployment Issues in Secure Multicast System with Group Key Management" published in International Journal of Computer Application (IJCA), Impact Factor 2.52, ISSN No. : 2250-1797, Volume 2, Issue 3, June 2012.

[9] H. Jiang, G. Zhang, G. Xie, K. Salamatian, and L. Mathy, ``Scalable high-performance parallel design for network intrusion detection systems on many-core processors,'' in Proc. 9th ACM/IEEE Symp. Archit. Netw. Commun. Syst. Piscataway, NJ, USA: IEEE Press, 2013, pp. 137-146.

[10] M.-J. Chen, Y.-M. Hsiao, H.-K. Su, and Y.-S. Chu, ``High-throughput ASIC design for e-mail and web intrusion detection,'' IEICE Electron.Express, vol. 12, no. 3, pp. 1-6, Jan. 2015.

[11] J.-M. Kim, A.-Y. Kim, J.-S. Yuk, and H.-K. Jung, ``A study on wireless intrusion prevention system based on snort,'' Int. J. Softw. Eng. Appl.,vol. 9, no. 2, pp. 112, 2015.