International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 07 | July 2022 www.irjet.net p-ISSN: 2395-0072

IMPROVING DDOS DETECTION IN IOT DEVICES

PONRADHA N[1], Mrs.R.SAHILA DEVI[2], M.E.,

[1]PG student, Rohini College of Engineering & Technology, Kanyakumari.
[2]Associate Professor, Rohini College of Engineering & Technology, Kanyakumari

---

ABSTRACT

Distributed denial of service (DDoS) attacks stay testing to mitigate in existing systems, recalling for home associations that include different Internet of Things (IoT) contraptions. DDoS traffic revelation model that includes a supporting system for determined model trees for different IoT device classes. Specifically, a substitute type of the model will be made and applied for each device class, since the qualities of the association traffic from each contraption class could have honest variation(s). As a context oriented examination, we get a handle on how devices in a normal smart home environment can be characterized into four unmistakable classes (and in our novel situation, Class 1 - outstandingly raised level of traffic consistency, Class 2 - raised level of traffic consistency, Class 3 - medium level of traffic consistency, and Class 4 - low level of traffic consistency). Disclosures from our appraisals show that the precision of our proposed approach is some place in the scope of 99.92% and 99.99% for these four contraption classes. With everything taken into account, we show the way that we can use contraption classes to help us even more truly perceive DDoS traffic.

Keywords: DDoS, IoT, mitigation, cloud computing.

1. Introduction

Distributed denial of service (DDoS) assaults have been a bad dream for big business tasks, accessibility, and security. After the development of current processing standards like distributed computing, these assaults saw significant changes in scale, techniques, points, and targets. The benefits gave Distributed denial of service (DDoS) assaults have been a bad dream for big business tasks, accessibility, and security. After the rise of current registering ideal models like distributed computing, these assaults saw significant changes in scale, techniques, points, and targets. The benefits given by distributed computing are accessible to the two casualties and the aggressors characterized by distributed computing are accessible to the two casualties and the assailants [1]. A comprehensive solution to DDoS attacks requires covering the global effects over a wide area of autonomous system (AS) domains on the Internet. Obviously, the global-scale defense is too costly for a real-life implementation. Even the Cyber Defense Technology Experimental Research (DETER) testbed can only emulate partial Internet activities. To implement an efficient defense system, we must leverage the network topology and use distributed traffic monitoring and detection. In reality, we build a DDoS defense system over a limited number of network domains serviced by the same Internet service provider (ISP). These ISP network domains cover the edge networks where the protected systems are physically connected [2]. At the beginning phase of a DDoS assault, the traffic changes are challenging to distinguish on the grounds that low traffic variances are not discernible. Observing Internet traffic at the singular stream level is cost restrictive to cover every single imaginable stream. In the mean time, the worldwide traffic in a wide region network is hugely enormous to perform ongoing discovery of organization oddities really. Latest works target countering DDoS assaults by battling the hidden vector, which is generally the utilization of botnets. A botnet is an enormous organization of compromised machines (bots) constrained by one element (the expert). The expert can send off synchronized assaults, like DDoS, by sending requests to the bots by means of a Command and Control channel. Tragically, recognizing a botnet is likewise hard, and effective arrangements might expect to partake effectively to the botnet itself [3], which raises significant moral issues, or to initially identify botnet-related malevolent exercises (assaults, diseases, and so on), which might defer the relief. Distributed computing is a model that permits organizations to send venture applications that, if appropriately planned, can scale their figuring assets on request. Organizations can either send their own applications on Infrastructure as a Service (IaaS) or Platform as a Service (PaaS) arrangements, or they can purchase prepared to-involve applications that utilization the Software as a Service (SaaS) [4] model. which raises significant moral issues, or to initially identify botnet-related malevolent exercises (assaults, diseases, and so on), which might defer the relief. Distributed computing is a model that permits organizations to send venture applications that, if appropriately planned, can scale their figuring assets on request. Organizations can either send their own applications on Infrastructure as a Service (IaaS) or Platform as a Service (PaaS) arrangements, or they can purchase prepared to-involve applications that utilization the Software as a Service (SaaS) [4] model.

A key element that has prompted the early reception of public distributed computing is the utility estimating model, which oversees the expense of registering assets consumed. Like public utilities, for example, gas and power, cloud buyers just compensation for the assets (stockpiling, transfer speed, and PC hours) they consume and for the time they utilize such assets. As per the terms of arrangement of the cloud specialist co-op (CSP), cloud buyers are liable for all computational expenses caused in their rented process conditions, whether or not the assets were consumed sincerely. Normal use cases for organizations that have taken on open distributed computing incorporate site and Web application facilitating and web based business. Like any Internet-confronting presence, these cloud based administrations are helpless against circulated refusal of-administration (DDoS) assaults. Such goes after are notable, and the related dangers have been well-informed. Here, we investigate a more unobtrusive assault on Web-based administrations facilitated in the cloud. Given the pay-more only as costs arise estimating, cloud-facilitated Web administrations are defenseless against assaults that look to take advantage of this model. An aggressor (for instance, a botnet) can play out a false asset utilization (FRC) assault by consuming the metered transmission capacity of Web based administrations, expanding the cloud shopper's monetary burden[5]. Distributed computing is at present perhaps the most advertised data innovation region and has become one of the quickest developing sections in IT industry. Because of the adaptability, pay per use, flexibility, versatility, and different characteristics guaranteed by this worldview, it acquired the interest of huge associations and corporates for facilitating their administrations onto the cloud. Nonetheless, the capacity to answer security dangers and occasions is recorded as one of the main pressing concerns of worry in distributed computing.

Distributed computing permits us to increase our servers and to serve an enormous number of solicitations for a help. The presentation of asset rich distributed computing stages, where adopters are charged in view of the utilization of the cloud's assets, referred to as "pay-as-you-use" or utility registering, has changed the Distributed Denial of Service (DDoS) chase down issue in the cloud to a monetary one. This new kind of assault focuses on the cloud adopter's financial assets, and is alluded to as Economic Denial of Sustainability (EDoS) assault [6]. All in all, the assault is making the cloud unreasonable by blurring the cloud charging system to charge the cloud client's bill for the assault's exercises. A notable strategy taken by EDoS assaults is to remotely control zombies to easily (with low rate to try not to set off security cautions) flood a designated cloud administration by undesired solicitations. Because of such undesired solicitations, and in light of the cloud versatility thought, the help use will be increased to fulfill the on-request demands. Furthermore, in view of the "pay per use" idea, a cloud adopter's bill will be charged for those undesired solicitations, prompting administration withdrawal or liquidation. What makes this more shocking is that it is incredibly hard to specifically channel the vindictive traffic without influencing the help overall. This likewise implies that any proposed moderating method should be exceptionally wise; any other way, the actual procedure could be used by the aggressors as a wellspring of EDoS assault.

In this paper, spread out a mathematical model in view of queueing speculation to formalize and take apart the low-rate DDoS attack circumstance in compartment based cloud climate. Considering the results of these examinations, we raise the characteristics and deficiencies of the compartment based cloud climate in defeating the low-rate DDoS attack and propose a strong DDoS help framework consenting to the components of compartment based cloud climate.

We research the probability that involving the new features in compartment based cloud climate defeats the low-rate DDoS attack. we raise the characteristics and deficiencies to direct low-rate DDoS attack in the holder based cloud climate. We spread out a mathematical model considering queueing speculation to formalize the low-rate DDoS attack circumstance in holder based cloud climate and analyze the restriction of holder based cloud climate in defeating against low-rate DDoS attack. Coordinated by this model, we propose an exceptional balance part to improve and organize the resource dissemination and the amount of holders for easing the low-rate DDoS attack.

1.1 Related Works

Aman bakshi et al.[7] lways creating cloud idea, issues are emerging from this "brilliant arrangement" in the endeavor field. Keeping interlopers from going after the cloud foundation is the main sensible thing the staff, the board and organizers can predict. Notwithstanding organization size or volume and extent of the cloud, this paper makes sense of how move IT virtualization system could be utilized in answering a refusal of administration assault. In the wake of getting a terribly unusual spike in inbound rush hour gridlock, designated applications could be quickly moved to virtual machines facilitated in another server farm.Wanchun Dou et al.[8] proposed Solidly talking, the technique is sent by two periods, i.e., non-assault period and assault period. All the more uncommonly, genuine parcels are gathered in the non-assault period, for removing trait matches to create an ostensible profile. With the ostensible profile, the CBF strategy is advanced by working out the score of a specific parcel in the assault period, to decide if to dispose of it or not. Finally, broad recreations are led to assess the plausibility of the CBF technique.

Wanchun Dou et al.[9] proposed Appropriated Denial of Service assault (DDoS), particularly HTTP, XML or REST based DDoS assaults might be extremely hazardous and may give exceptionally destructive impacts to accessibility of administrations and all purchasers might get impacted simultaneously. Another explanation is that on the grounds that the distributed computing clients cause their solicitation in XML and afterward to send this solicitation utilizing HTTP convention and fabricate their framework communicate with REST convention (like Amazon EC2 or Microsoft Azure) thus XML assault more helpless. So the compromise coming from conveyed REST assaults are more and simple to carry out by the aggressor, however to security master undeniably challenging to determine. Mohd Nazri Ismail et al.[10] proposed another model to recognize flooding based DoS assault in cloud climate has been recommended comprising three stages. (1) The first-stage is to display the typical traffic design for pattern profiling and (2) the subsequent stage is the interruption identification cycles and (3) at last anticipation stage.

Wanchun Dou et al.[11] proposed Hadoop Map Reduce and Spark to accelerate information handling by isolating and handling information streams simultaneously. With a certifiable informational collection, we directed true trials to assess the viability of our created network observing and danger recognition framework as far as organization checking, danger discovery, and framework execution. Our experimental information shows that the proposed framework can productively screen network exercises, track down strange ways of behaving, and identify network dangers to safeguard basic foundation frameworks. Massimo Ficco et al.[12] proposed a strategy to orchestrate stealthy attack patterns, which exhibit a slowly-increasing-intensity trend designed to inflict the maximum financial cost to the cloud customer, while respecting the job size and the service arrival rate imposed by the detection mechanisms.

Oswaldo Olivo et al.[13] proposed that Torpedo can success- fully detect second-order DoS vulnerabilities in widely used web applications written in PHP. Once our tool discovers a vulnerability, it also performs symbolic execution to generate candidate attack vectors. Gaurav Somani et al.[14] proposed a clever moderation system, DARAC, which pursues auto-scaling choices by precisely separating between genuine solicitations and aggressor traffic. Aggressor traffic is distinguished and dropped in view of human conduct examination based discovery. We likewise contend that the greater part of the arrangements in the writing, don't give a lot of consideration to the help quality to genuine solicitations during an assault.

Gaurav Somani et al.[15] proposed the isolated environment is another virtual machine that is similar to the original one. At the meantime, execution of the operating system and tagged application keeps continuously both in new environment and original one. Siqin Zhao et al.[16] proposed a novel resource containment approach to enforce the victim's resource limits. Our real-time experimental evaluations show that the proposed approach results in reduction in the attack reporting time and victim service downtime by providing isolated and timely resources to ensure availability of other critical services.

youhuizi li et al.[17] proposed PINE, a performance isolation optimization solution in container environments, which can adaptively allocate the storage resource for each service according to their performance behaviors through dynamical resource management and I/O concurrency configuration. Ping Du et al.[18] proposed a cloud-based attack defense system called CLAD, which is running on cloud infrastructures as a network service to protect Web servers. Gaurav Somani et al.[19] proposed DDoS attacks on cloud services, where having the same attack features, two different services show completely different consequences, due to the difference

in the resource utilization per request. Manoj Singh Gaur et al.[20] proposed a novel resource containment approach to enforce the victim's resource limits. Operating System (OS) level "internal collateral damage", in which the other critical services are also affected.

To solve the above problems, we developed dynamic DDoS mitigation mechanism to defeat the low-rate DDoS attack in container-based cloud environment. The article is organized as follows: section 1 presents introduction and related works. Section 2 discusses the proposed method DDoS attack in container-based cloud environment. Section 3 discusses the experiments and results, and section 4 describes the conclusion of the research work.

1.2 Our Contribution

In this work, environment consists of 41 assorted SHIoT gadgets, and the supporting correspondence framework and programming equipment stage are likewise arrangement to empower traffic assortment that can be utilized to prepare DDoS identification models. Notwithstanding the essential information gathered in this examination, we likewise utilized optional information, including a bigger number of different SHIoT gadgets (i.e., more noteworthy gadget heterogeneity). The Fortinet AP 221C remote passageway, the Cisco 2960 Catalyst 48 PoE (Power over Ethernet) switch,

the HP Pavillion dm1, and Microsoft HP 10 10.0.17134 form 17134 workstations have been set up to catch traffic utilizing port reflecting, x64 processor engineering, AMD E-350, 1600MHz two centers, 4 GB RAM) with Wireshark programming device variant 2.6.3 introduced. The switch's actual correspondence ports (FA0/1 and FA0/3) to which the remote passageway and IoT center point for the Phillips Hue gadget are associated are designed for port reflecting. These ports are set up as sources, which guarantees that all traffic to and from them is reflected (planned) to the objective contact port (FA0/2). A traffic assortment workstation is associated with this port. With an authentic traffic profile of a SHIoT gadget, it is vital to have a dataset that incorporate DDoS traffic. These two sets structure the reason for fostering a powerful model for identifying network traffic inconsistencies, for example, DDoS traffic created by SHIoT gadgets.

2. Methods

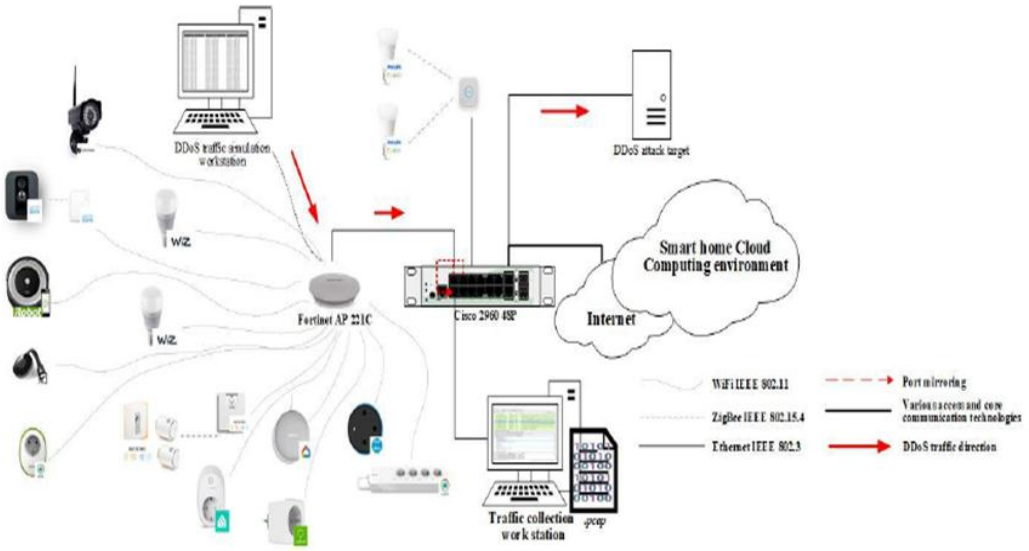

Fig. 1 System Architecture

2.1 Defining legitimate traffic profiles for classes of SHIoT device

SHIoT is a dynamic and inescapable climate, where new client IoT devices with different functionalities are persistently familiar with the market. To encourage a DDoS traffic disclosure model considering as of late portrayed SHIoT contraption classes, it is essential to describe a certifiable traffic profile of each and every device class. In the improvement of any irregularity recognizable proof model considering controlled AI procedures, it is vital to have a lot of data that will address certifiable traffic and a lot of data that will address nonsensical traffic.

2.2 Formation of data sets for the development of DDoS traffic detection models

The SHIoT contraption classes described by the investigation enable the ID of the class relationship of the device considering the traffic stream made by the contraption. This moreover engages the development of a bona fide traffic profile considering the way that each traffic stream alloted to one of the four described classes by the request model ends up being fundamental for a set that tends to a genuine traffic profile of a comparable class. To cultivate a model for perceiving (nonsensical) DDoS network traffic, the Logistic Model Trees (LMT) procedure was used. For the execution of the methodology and data taking care of, we used the WEKA programming device, as well as instructive assortments that address profiles of common traffic coming about in light of the SHIoT contraption gathering model and educational assortments of misguided DDoS traffic. Also similarly as with any AI model new development, the goal is to use those independent components whose change greatestly influences changing the dependent part. Moreover basic to diminish those components can incite model tendency. Thusly, moreover with the improvement of the SHIoT device portrayal model, independent components z1 to z7 address traffic stream identifiers and contain information on the source and objective IP addresses, shows used, and traffic stream age time wiped out from the fundamental datasets.

2.3 The working principle of the developed model for detection of illegitimate DDoS network traffic

Made by the created model of absurd DDoS traffic area occurs in two phases. The essential stage is a fundamental for later distinguishing proof of DDoS traffic in the ensuing stage and remembers the portrayal of SHIoT contraptions for perspective on made traffic stream. The multi-class gathering model results show the way that the SHIoT contraption can be described into one of four predefined classes concerning the traffic streams it makes with an accuracy of 99.79%. The lower level of traffic consistency is achieved by the contraption's strategy for movement, similar to a raised level of client association, playback of sound/video content, and the like.This achieves a more incredible LMT model that can't be summarized to the root center point, yet it includes 11 centers or 6 terminal center points. An essential backslide model is portrayed on each piece of the decision tree polishing off with the terminal center point. In the ongoing case, this suggests that the LMT model contains an amount of five growing concentrations and six vital backslide models. A LMT model containing a decision tree and related vital backslide models with picked relevant independent features and related coefficients. The working rule of the made model for acknowledgment of nonsensical DDoS network traffic . Made by the made model of nonsensical DDoS traffic area occurs in two phases. The chief stage is a fundamental for later revelation of DDoS traffic in the ensuing stage and remembers the request for SHIoT contraptions for light of delivered traffic stream.

3. Results and discussion

Due to the qualities of low-rate DDoS assault, the harmless and the noxious solicitations have comparable ways of behaving in rush hour gridlock. Subsequently, we assume the organization traffic is steady and consistent whether under nonattack situations or assault situations. Furthermore, as a general rule, the solicitations couldn't be in every way malignant demands enduring an onslaught scenarios. suppose the appearance pace of solicitations to the microservice follows the Poisson dispersion under nonattack situations.

3.1 Performance of Unknown Users

To unknown requests, the remainder of assets will be allocated to them in the wake of meeting the prerequisite of whitelist demands. What's more, we endeavor to make the normal remaining time of obscure solicitations adequate under the situation that the extent of malevolent solicitations as high as could really be expected. To accomplish this objective, we lead an investigation of the relationship between the quantity of compartments and normal remaining time of request under various assault qualities. To examine the accuracy and viability of the relief component, we lead a set of reenactment examinations to gauge the normal remaining season of solicitations after DDoS alleviation instrument improving under various assault qualities. What's more, as the correlation bunch, we run the recreation explores different avenues regarding something similar designs to acquire the remaining season of solicitations in the most pessimistic scenario without improvement. In each trial, we direct multiple times reenactments and afterward take the normal as the end-product. The subtleties of the examination results are displayed in Fig. 2.

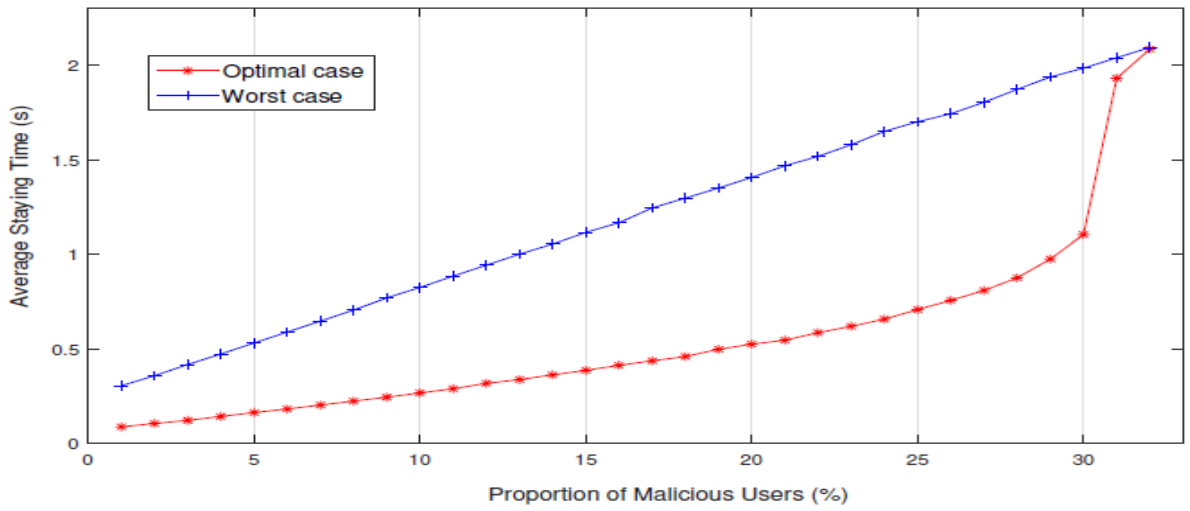

Fig. 2: Average waiting time of unknown users in scenarios with a different number of containers and different attack strengths.

3.2 Effectiveness on Complex DDoS Scenarios

Now and again, the low-rate DDoS attack isn't shipped off independently besides, mixes in with the flood-based DDoS attack. To survey the practicality of our mitigation framework in these confounded circumstances, we impersonate a flood-based DDoS attack and add it into the past attack cases which in a manner of speaking contain the low-rate DDoS attack. Specifically, the ordinary attack speed of the flood-based DDoS attack is 5,000 requesting each second, which is the on numerous occasions occupations than the regular case. Additionally, the evaluation results are shown in Fig. 3. The alleviation component, the QoS of whitelist solicitations can in any case be kept up with in the typical level under the combination DDoS assaults. Likewise, with the assault strength of the low-rate DDoS assault expanding, the QoS of obscure solicitations has a level of progress contrasted with the cases without our DDoS moderation instrument.However, the QoS of the unknown requests is still far away from the acceptable level, because the micro service only has limited resources to face the massive DDoS requests. In order to maximize the effectiveness of our mitigation mechanism, combining it with the traffic filtering mechanism will have greater capacity to defeat the DDoS attacks in these complex scenarios. In this case, the traffic filtering mechanism is a complement for our mitigation mechanism to filter the malicious requests from flood-based DDoS attack.

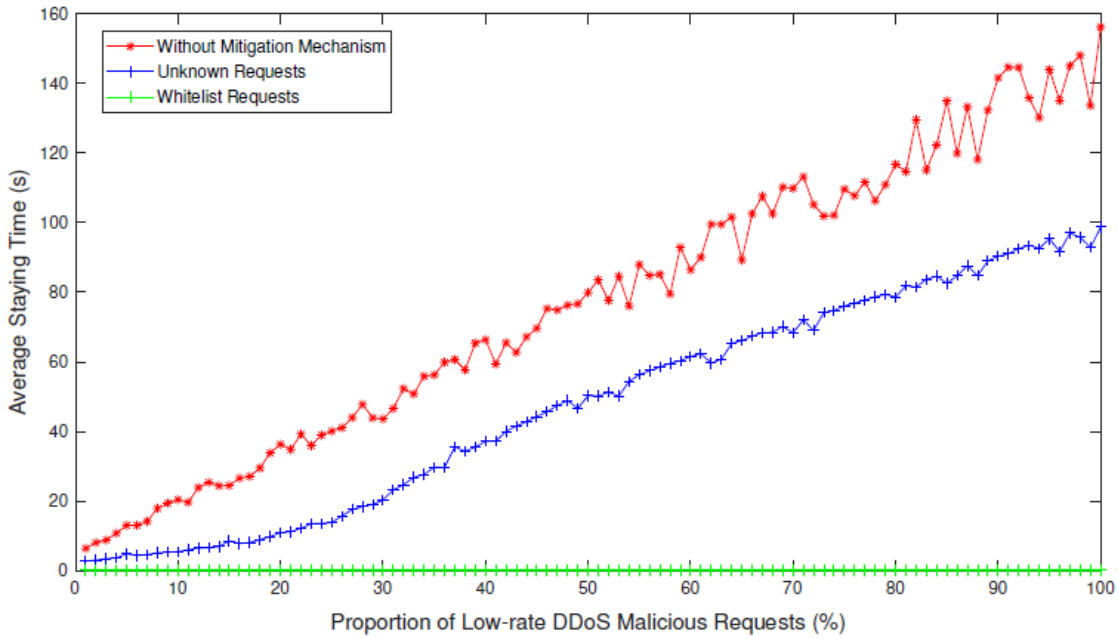

Fig. 3: Average waiting time of requests in complex DDoS scenarios with or without the DDoS mitigation mechanism.

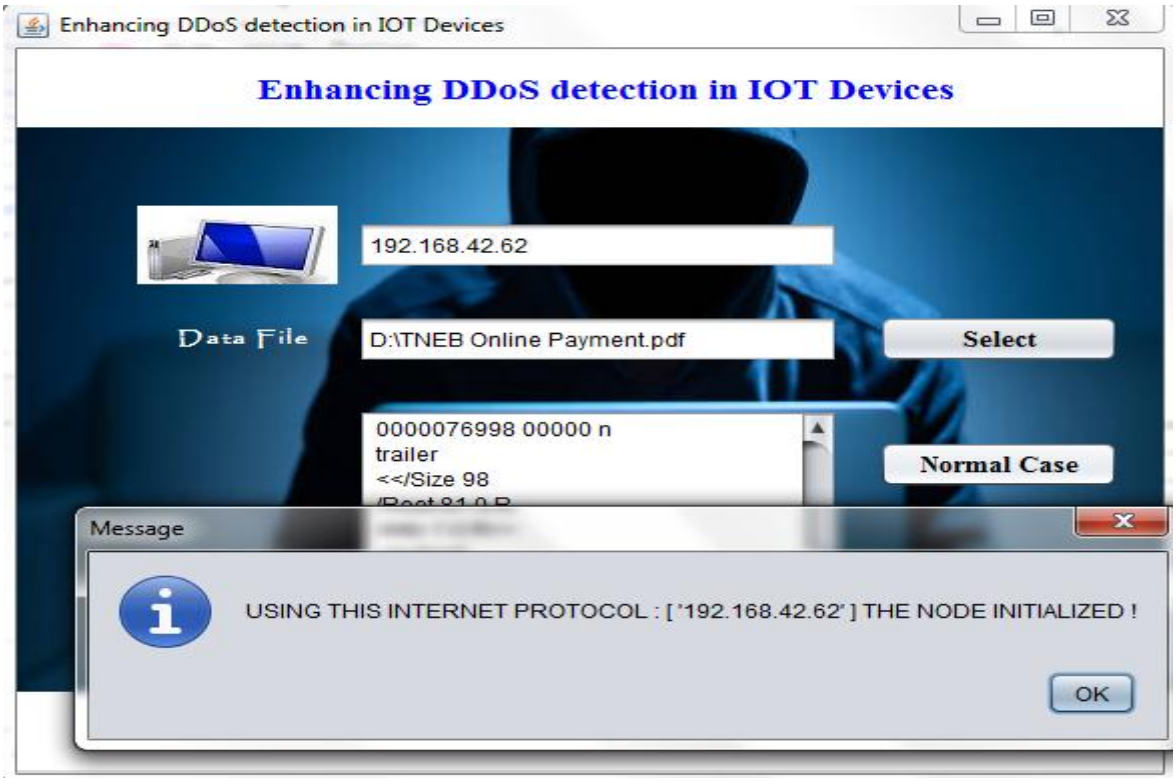

Fig 4. Send file using normal case

Fig 5. File send using DDOS case

4. Conclusion

The DDoS recognizable proof model presented in this paper diverges from the all around average organization traffic peculiarity acknowledgment draws near. Earlier methodologies are by and large established on making a real traffic profile that is supposed to apply to each and every terminal device. Such a philosophy is cognizant in conditions containing normal devices whose traffic makes characteristics that are smart of the action of the presented applications on the contraptions and how the clients use such contraptions. Thusly, DDoS disclosure approaches considering individual device characteristics require re-learning or even redevelopment of the fundamental model for each new contraption that appears accessible. Such a procedure is unquestionably convoluted and deficiently nonexclusive in an obviously baffling and dynamic IoT climate.

REFERENCES

[1] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, M. Rajarajan, and R. Buyya, "Combating DDoS attacks in the cloud: Requirements, trends, and future directions," IEEE Cloud Computing, vol. 4, no. 1, pp. 22–32, 2017.

[2] Y. Chen, K. Hwang, and W.-S. Ku, "Collaborative detection of DDoS attacks over multiple network domains," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 12, pp. 1649–1662, 2007.

[3] J. Franc¸ois, I. Aib, and R. Boutaba, "Firecol: a collaborative protection network for the detection of flooding ddos attacks," IEEE/ACM Transactions on Networking, vol. 20, no. 6, pp. 1828–1841, 2012.

[4] M. Villamizar, O. Garc´es, L. Ochoa, H. Castro, L. Salamanca, M. Verano, R. Casallas, S. Gil, C. Valencia, A. Zambrano et al., "Cost comparison of running web applications in the cloud using monolithic, microservice, and aws lambda architectures," Service Oriented Computing and Applications, vol. 11, no. 2, pp. 233–247, 2017.

[5] J. Idziorek, M. F. Tannian, and D. Jacobson, "The insecurity of cloud utility models," IT Professional, vol. 15, no. 2, pp. 22–27, 2013.

[6] M. H. Sqalli, F. Al-Haidari, and K. Salah, "Edos-shield-a two-steps mitigation technique against edos attacks in cloud computing," in Proceedings of the 4th International Conference on Utility and Cloud Computing, 2011, pp. 49–56.

[7] A. Bakshi and Y. B. Dujodwala, "Securing cloud from ddos attacks using intrusion detection system in virtual machine," in Proceedings of the 2nd International Conference on Communication Software and Networks, 2010, pp. 260–264.

[8] W. Dou, Q. Chen, and J. Chen, "A confidence-based filtering method for ddos attack defense in cloud environment," Future Generation Computer Systems, vol. 29, no. 7, pp. 1838–1850, 2013.

[9] T. Karnwal, S. Thandapanii, and G. Aghila, "A Filter Tree Approach to Protect Cloud Computing against XML DDoS and HTTP DDoS Attack," in Proceedings of the 2012 International Symposium on Intelligent Informatics, 2012, pp. 459–469.

[10] M. N. Ismail, A. Aborujilah, S. Musa, and A. Shahzad, "Detecting flooding based DoS attack in cloud computing environment using covariance matrix approach," in Proceedings of the 7th international conference on ubiquitous information management and communication, 2013, pp. 36:1–36:6.

[11] Z. Chen, G. Xu, V. Mahalingam, L. Ge, J. H. Nguyen, W. Yu, and C. Lu, "A cloud computing based network monitoring and threat detection system for critical infrastructures," Big Data Research, vol. 3, pp. 10–23, 2016.

[12] M. Ficco and M. Rak, "Stealthy denial of service strategy in cloud computing," IEEE Transactions on Cloud Computing, vol. 3, no. 1, pp. 80–94, 2015.

[13] O. Olivo, I. Dillig, and C. Lin, "Detecting and exploiting second order denial-of-service vulnerabilities in web applications," in Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015, pp. 616–628.

[14] G. Somani, A. Johri, M. Taneja, U. Pyne, M. S. Gaur, and D. Sanghi, "DARAC: DDoS Mitigation Using DDoS Aware Resource Allocation in Cloud," in Proceedings of the 11th International Conference on Information Systems Security, 2015, pp. 263–282.

[15] Y. Gilad, A. Herzberg, M. Sudkovitch, and M. Goberman, "CDNon- Demand: An affordable DDoS Defense via Untrusted Clouds," in Proceedings of the 23rd Annual Network and Distributed System Security Symposium, 2016.

[16] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and M. Rajarajan, "DDoS victim service containment to minimize the internal collateral damages in cloud computing," Computers & Electrical Engineering, vol. 59, pp. 165–179, 2017.

[17] Y. Li, J. Zhang, C. Jiang, J. Wan, and Z. Ren, "PINE: optimizing performance isolation in container environments," IEEE Access, vol. 7, pp. 30 410–30 422, 2019.

[18] P. Du and A. Nakao, "DDoS defense as a network service," in Proceedings of the 10th Network Operations and Management Symposium, 2010, pp. 894–897.

[19] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "Service resizing for quick DDoS mitigation in cloud computing environment," Annales des T´el´ecommunications, vol. 72, no. 5-6, pp. 237–252, 2017.

[20] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and M. Rajarajan, "DDoS victim service containment to minimize the internal collateral damages in cloud computing," Computers & Electrical Engineering, vol. 59, pp. 165–179, 2017.

[21] Y. Xiang, K. Li, and W. Zhou, "Low-Rate DDoS Attacks Detection and Traceback by Using New Information Metrics," IEEE Transactions on Information Forensics and Security, vol. 6, no. 2, pp. 426–437, 2011.

[22] J. Luo, X. Yang, J. Wang, J. Xu, J. Sun, and K. Long, "On a mathematical model for low-rate shrew DDoS." IEEE Transactions on Information Forensics and Security, vol. 9, no. 7, pp. 1069–1083, 2014.

[23] J. Burnim, S. Juvekar, and K. Sen, "WISE: Automated test generation for worst-case complexity," in Proceedings of the 31st International Conference on Software Engineering, 2009, pp. 463–473.

[24] B. Tak, C. Isci, S. S. Duri, N. Bila, S. Nadgowda, and J. Doran, "Understanding security implications of using containers in the cloud," in Proceedings of the 2017 USENIX Annual Technical Conference, 2017, pp. 313–319.

[25] C. Raiciu, S. Barr´e, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, "Improving datacenter performance and robustness with multipath TCP," in Proceedings of the 2011 ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2011, pp. 266–277.