Impact of Jellyfish attack on routing protocols in TCP-based MANETs

Khaled Ahmed Abood Omer
Computer Science and Engineering Department, Faculty of Engineering, University of Aden
E-mail address: k_abood@hotmail.com

DOI: https://doi.org/10.47372/uajnas.2023.n1.a09

Abstract

Mobile ad-hoc networks (MANETs) are self-organized infrastructure-less network of mobile wireless devices that could be deployed for communication. Due to the insecure wireless communication medium, multi-hop routing communication process, and dynamic behavior of the nodes in MANETs, routing protocols are vulnerable to various security attacks, such as Jellyfish attacks. A Jellyfish node targets TCP-based MANET and exploits its working mechanism to degrade the communication performance. This attack is hard to detect since it is a TCP protocol compliant methodology.

In this paper, we evaluate the performance of the Ad hoc on-demand vector (AODV), Dynamic source routing (DSR), Temporally ordered routing algorithm (TORA), Geographic routing protocol (GRP), and optimized link state routing (OLSR) routing protocols under the Jellyfish delay variance attacks for TCP-based MANETs. Further, the TAHOE, RENO, and SACK variants of TCP protocol are considered for comparison. These routing protocols are simulated using the OPNET simulator to compare their performance, using specific performance metrics on the network. The experimental results show that the AODV protocol performs better than the DSR, TORA, OLSR, and GRP protocols under the jellyfish delay variance attack. Further, the SACK TCP variant performs better than the other TCP variants under the Jellyfish delay variance attack.

Keywords: Routing protocols, MANET, TCP, Jellyfish attacks.

1. Introduction

Mobile ad-hoc networks (MANETs) consist of wireless mobile nodes that communicate with each other in the absence of permanent infrastructure. In MANET, each node works as a router and a host for data forwarding. MANET is based on the cooperation among participating nodes such that every node forwards packets to ensure that packets are sent from source to destination in a multi-hop route, using intermediate nodes for data forwarding. These intermediate nodes are independent and expected candidates to become attacker nodes.

MANETs can be used in different fields such as earthquake and flooding, automated battlefields, agriculture fields, security and vigilance, search and rescue, crowd control, indoor and outdoor conferences, and robot networks [1, 2].

However, MANETs are exposed to security vulnerabilities due to their dynamic topology changes, and no centralized network management. Due to these reasons, MANETs are exposed to security risks, such as jellyfish, information disclosure, intrusion, denial of service, flooding and impersonation attacks, selfish node misbehaving, etc. Therefore, the security requirements in MANETs are much higher than those in wired networks [3, 9]. As a result, providing security in MANETs has become a major concern for researchers.

However, applications that require reliable in-order delivery and end-to-end services, such as file transfer protocol (FTP), and secure hypertext transfer protocol (HTTP), must rely on Transmission Control Protocol (TCP) for their communication. In MANETs, TCP performance degrades with an increase in network mobility. This is because TCP has no separate mechanism to identify whether a packet has been dropped due to wireless mobile network characteristics or network congestion. TCP's flow and congestion control mechanism treats every packet loss as a sign of congestion and decreases its transmission rate leading to a decrease in the network resource utilization and the network throughput.

This paper aims at evaluating the performance of routing protocols in TCP-based MANETs under Jellyfish delay variance attack. This attack is hard to detect since it is a protocol-compliant methodology. A Jellyfish node targets a TCP protocol and exploits its working mechanism to degrade the communication performance. We have evaluated the effects of the delay variance Jellyfish attack over the routing protocols in TCP-based MANETs , using Tahoe, Reno, and SACK TCP variants.

This paper is organized as follows: Section 2 describes the related work about routing protocols, TCP variants, and Jellyfish attacks. Section 3 illustrates a simulation environment of this work. The obtained experimental results and discussions are described in section 4. Finally, section 5 concludes the findings of this work.

2 Related work

In MANETs, Routing protocols are classified into three categories named as proactive, reactive, and hybrid routing protocols. In proactive or table-driven routing protocols, the routes to all the nodes are maintained in the routing table. Packets are sent over a predefined route specified in the routing table such as OLSR and GRP protocols. In a reactive or on-demand routing protocol, the routes are established on request for routing such as AODV, DSR, and TORA protocols. A source node initiates the route discovery phase to find a new route whenever there are packets to be sent to a destination [1, 2]. In this paper, we consider AODV, DSR, TORA, OLSR, and GRP routing protocols for performance evaluation under Jellyfish security attacks in a TCP based MANET network.

2.1 TCP variants:

Current TCP implementations contain several algorithms aimed at controlling network congestion while maintaining good throughput. Early TCP implementations followed a go-back model, using cumulative positive acknowledgment and requiring a retransmit timer expiration to re-send data lost during transport. These TCPs contributed less to reducing network congestion.

The Tahoe TCP variant added some new algorithms and refinements to earlier implementations of the TCP protocol. These algorithms include Slow-Start, Congestion Avoidance, and Fast Retransmit [Jac88]. The refinements include a modification to the round-trip time estimator used to set retransmission timeout values. All modifications have been described elsewhere [6, 13]. In TCP Tahoe, an RTO is an indication of the congestion and enters the congestion avoidance phase by setting the congestion window (cwnd) to 1 and the slow start threshold (ssthresh) to half of cwnd. The cwnd is increased additively till ssthresh is reached, then increased linearly until a packet loss is encountered. It does not have a fast recovery state and , during the congestion avoidance phase, Tahoe treats triple duplicate ACKs the same as a timeout.

The Reno TCP variant retained the improvements included in Tahoe but modified the Fast Retransmit operation to include Fast Recovery. This algorithm prevents the communication path from going empty after Fast Retransmit, thus avoiding the need to Slow-Start to re-fill it after a single packet loss. Fast Recovery operates by assuming each duplicate ACK received represents a single packet having left the pipe. Thus, during Fast Recovery, the TCP sender makes quick estimates of the amount of outstanding data. TCP Reno uses the logic of duplicate acknowledgements (dupacks) to trigger Fast Retransmit. After 3 dupacks, TCP Reno takes it as a sign of segment loss and retransmits the packet immediately and enters Fast Recovery. In Fast Recovery, ssthresh and cwnd are set to half the value of current cwnd. For each subsequent dupack, increase cwnd by one and transmit a new segment if the new value permits it.

The SACK TCP variant holds the properties of Tahoe and Reno TCP of being robust in the presence of out-of-order packets, and uses retransmit timeouts as the recovery method. The main difference between the SACK TCP and the Reno TCP is in the behavior when multiple packets are dropped from one window of data. The SACK TCP allows the receiver to acknowledge non-consecutive data, which only permit non-transmitted or the missing data to be retransmitted once again [4, 5, 7].

2.2 Jellyfish Attacks

TCP-based MANETs use protocol with congestion control techniques in the transport layer. These attacks maintain compliance with both the control and data protocols to make their detection and prevention difficult. In MANETs, an intermediate node can introduce a critical vulnerability for the TCP congestion control mechanism. The jellyfish attacker disrupts the TCP connection which is established for communication. Jellyfish attacker intrudes into forwarding nodes and delays data packets unnecessarily for some amount of time before forwarding them [3]. Due to Jellyfish attack, high end-to-end delay takes place in the network resulting in poor performance of the network. In this attack, a malicious node disrupts the whole functionality of the TCP protocol and may reorder, delay, and drop packets. This behavior complies with the TCP protocol making it difficult to detect. Many applications, such as web, and file transfer, require reliable, congestion-controlled delivery as provided by the TCP protocol. Jellyfish attack is further divided into three categories i.e. jellyfish reorder attack, jellyfish periodic dropping attack, and jellyfish delay variance attack [9]. Such a compromised node alters its forwarding behavior as described in the following jellyfish delay variance attacks.

In jellyfish delay variance attack, Round trip time (RTT) of data packets vary considerably due to congestion. These changes in RTT force TCP to increase retransmission timeout (RTO). In the Jellyfish delay variance attack, the packets are delayed as they are forwarded by the Jellyfish Attacker node in MANET. High delay variation can cause TCP to send traffic in bursts that increases collisions and loss of packets. High delay variation leads to high RTO value. Packets delayed by the jellyfish attacker have the potential to reduce throughput of network.

In Jellyfish attack, the attacker uses a suitable attack to insert malicious software on a number of nodes distributed all over the network. This malicious software does not cause any harm to these nodes. Next , all these nodes are coordinated and triggered to launch the attack in the network.

Dulaimi et. al. analyzed the effect of jellyfish attacks in an AODV network, using an OMNET++ simulator. The simulation was done, using UDP packets for varying number of nodes in the network. They found that the jellyfish attack affects throughput of the network because of congestion caused due to retransmissions, and PDR decreases due to increased retransmission timeout led by delay introduced by attacker. It increases end-to-end delay of the network. The effect is more devastating as the number of jellyfish node increases.

Kaur et. al. presented the impact of jellyfish attack on MANET, using TORA protocol and the proposed Selective Node Participation Approach [8]. The proposed approach reduces the impact of jellyfish attack in MANET by deactivating the jellyfish nodes to participate in the DAG of TORA protocol, but still maintains the overall integrity of the DAG. They concluded that the performance of network has been improved by selective node participation in terms of end-to-end delay, Packet Delivery Ratio and Throughput of the network.

Laxmi et. al. presented the performance evaluation of the AODV routing protocol under jellyfish attack in TCP-based MANETs [9]. Based on the simulation results generated over various MANET scenarios with a varying numbers of attackers, intermediate hops, and attack parameters, it has been observed that jellyfish attack degraded the network performance in terms of network throughput, end-to-end delay, and control overhead.

Mishra et. al. used trust based parameters and perceptron logic in order to avoid such maliciously behaving nodes , using network simulator NS-2 [10]. They studied and analyzed the malicious behavior of the jellyfish mobile nodes in MANET , using the AODV protocol. They suggested a technique to avoiding dropping packets and delay variance types of jellyfish attacks. Trust counter is used to avoid both types of attacks. Improvement in the AODV was observed by using their proposed scheme.

Sachdeva et. al. implemented a jellyfish delay variance attack on AODV and proposed a Jellyfish delay variance detection algorithm that analyzed packet delaying misbehavior of nodes and detected multiple jellyfish delay variance attacker nodes [11]. The results reduced average end-to-end delay and increased throughput by re-routing data packets through alternate routes consisting of non-malicious nodes.

Sajjad et. al. analyzed the performance of the Dynamic Source Routing (DSR) routing protocol in the presence of a jellyfish attack [12]. They created different scenarios having a various numbers of jellyfish attacks in MANETs , using the OPNET Modeler 14.5 simulator. From the simulation result, it has been observed that jellyfish attack significantly degraded the performance of the DSR protocol in terms of end-to-end delay, throughput, and packet delivery ratio. Moreover, it has also been observed that ,when the number of jellyfish attacks increases in the network the performance is further degraded.

Wazid et.al. analyzed the effect of jellyfish delay variance attack on MANET, using the AODV routing protocol. The performance analysis is done concerning some network parameters, like throughput, end-to-end delay, etc., using the OPNET modeler 14.5 simulator. It was observed that MANET is resilient to up to 10% of jellyfish attackers which did not make any hard impact on the performance of the network. For attackers above 10% and below 20% ,performance was affected with an average rate but for 20% or above 20% performance of the network became worse.

3. Simulation Environment

In this paper, we evaluate the performance of MANET routing protocols under Jellyfish delay variance attack, using the OPNET modeler 14.5 simulator. In the Jellyfish delay variance attack, the packets are delayed as they are forwarded by the Jellyfish Attacker node in MANET. In the attack model used in this work, the attackers compromise four mobile nodes by installing malicious code into them , using worms. However, the attacker could be an internal or an external node. The compromised mobile nodes delay the forwarded packets in the network. The Simulation parameters used in our experiments are shown in Table 1 below.

Table 1 Simulation Parameters

Network Parameters	Values
Number of Mobile Nodes	21
Simulation Time	900 seconds
Simulation Area	1000 m x 1000 m
Routing Protocols	AODV, DSR, TORA, OLSR, GRP
Mobility Model	Random waypoint (speed 0-10m/s)
PHY Characteristic	PHY 802.11
TCP variants	Tahoe, Reno, SACK

The attack simulation model is made of 21 nodes deployed in a 1000m x1000m network, as shown in Figure 1. There is only one HTTP server node, 16 legitimate nodes, and 4 Jellyfish attacker nodes with infected software during the Jellyfish attack period. In the four jellyfish nodes, the datagram forwarding rate of the IP processor is reduced to 1000, 2000, 3000, and 4000 packets/sec to make the jellyfish nodes slower. Also, the memory size of the IP processor (queue) is reduced to 8 MB to see the packets get dropped after the queue is filled up.

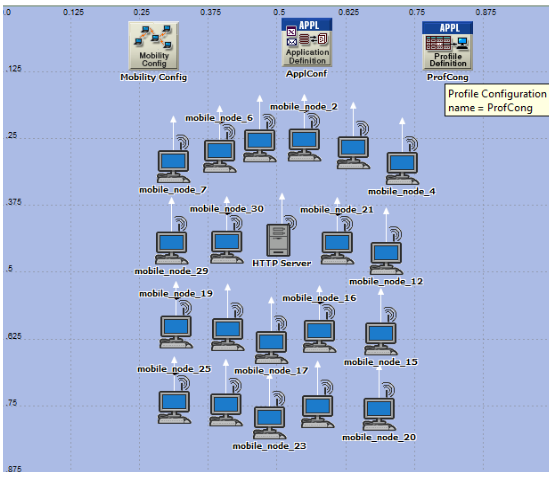


Figure 1 MANET network deployment

The simulation time for the attack simulation model is 900 seconds, and the start time of packet generation at 100 seconds. In the legitimate nodes, the datagram forwarding rate of the IP processor is 400000 packets/seconds to the end of the simulation (900 seconds).

3.1 Performance metrics:

In this paper, we consider the following performance metrics to evaluate the performance of the routing protocols under investigation:

Page response time (sec): It specifies the time required to retrieve the entire page with all the contained inline objects.

TCP Delay (in seconds): The delay of packets received by the TCP layers for all connections in the entire network. It is measured from the time an application data packet is sent by the source TCP layer to the time it is completely received by the TCP layer in the destination server.

Data dropped (buffer overflow bits/sec): The total size of higher layer data packets dropped by all the WLAN MACs in the network due to insufficient higher layer data buffer space. A lower data drop leads to better routing protocol performance.

Throughput (bits/s): Represents the total number of bits (in bits/sec) forwarded from wireless LAN layers to higher layers in all WLAN nodes of the network.

4. Results and Discussion:

In this section, we present and discuss the experimental results obtained by the simulation of the five routing protocols with the three TCP variants according to the attack model described above.

Figure 2a, Figure 2b, and Figure 2c show the page response time for the five routing protocols, using the three TCP variants respectively. The figures show that the AODV protocol has the minimum page response time for the considered TCP variants, and the TORA protocol has the maximum page response time.

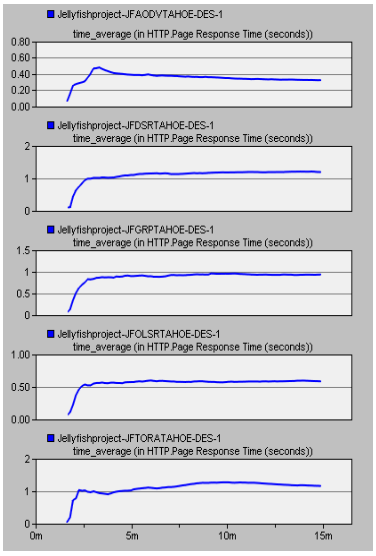


Figure 2a page response time, Tahoe

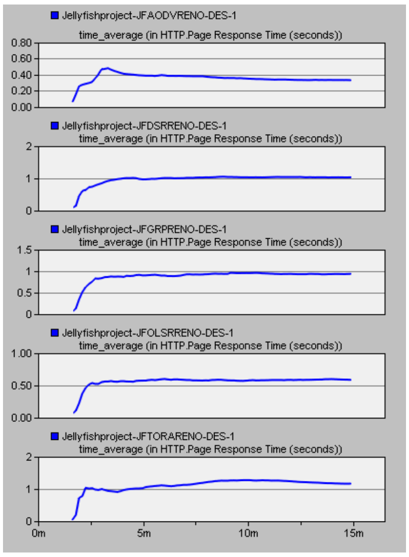


Figure 2b page response time, Reno

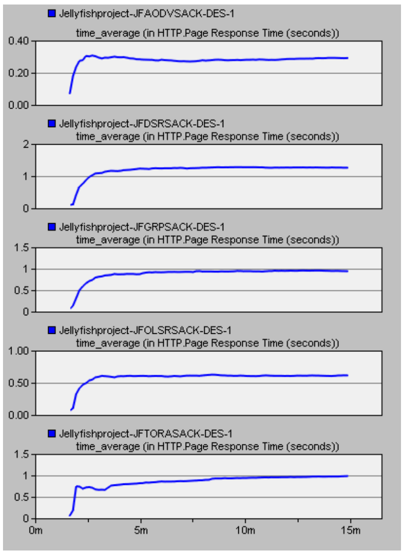
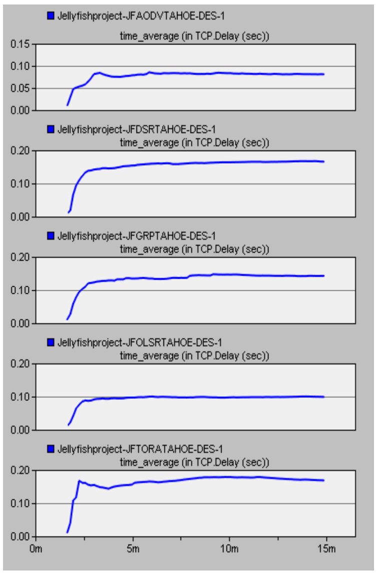
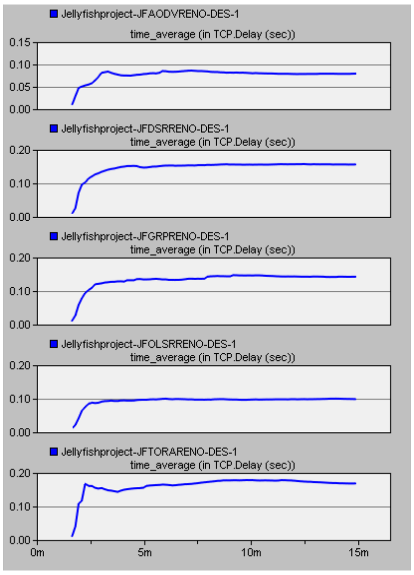


Figure 2c page response time, SACK

Figure 3a, Figure 3b, and Figure 3c show the TCP delay for the five routing protocols, using the three TCP variants respectively. The figures show that the AODV protocol has the minimum TCP delay for the considered TCP variants, and the DSR protocol has the maximum TCP delay.





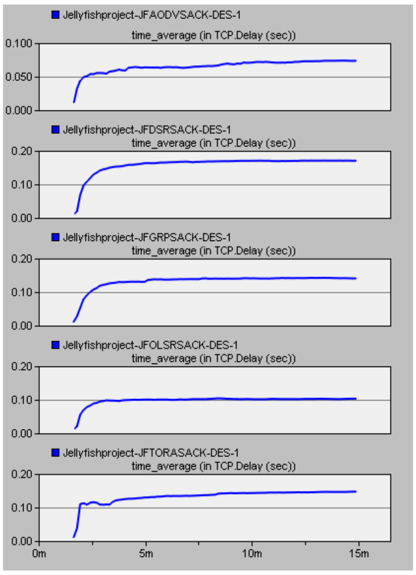
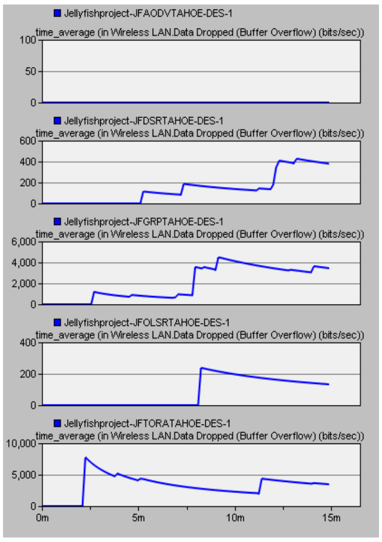
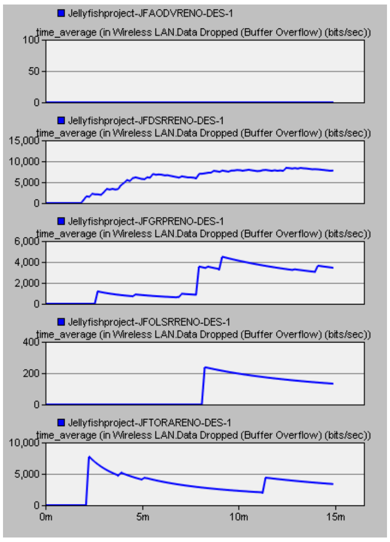


Figure 3a TCP Delay, Tahoe Figure 3b TCP Delay, Reno Figure 3c TCP Delay, SACK

Figure 4a, Figure 4b, and Figure 4c show the wireless LAN data dropped for the five routing protocols, using the three TCP variants respectively. The figures show that the AODV protocol has the minimum data dropped which is almost equal to zero for the considered TCP variants, and the GRP protocol has the maximum data dropped.





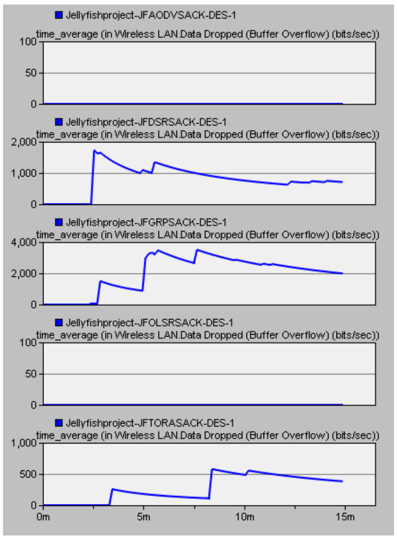
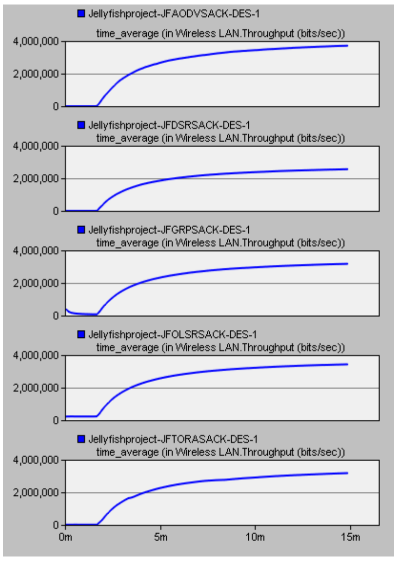
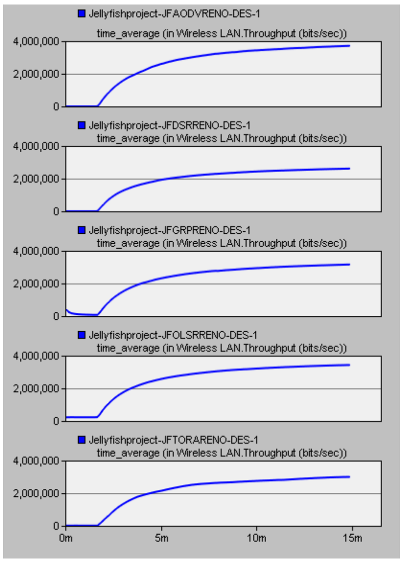


Figure 4a WLAN Data drop, Tahoe Figure 4b WLAN Data drop, Reno Figure 4c WLAN Data drop for SACK

Figure 5a, Figure 5b, and Figure 5c show the wireless LAN throughput for the five routing protocols, using the three TCP variants respectively. The figures show that the AODV protocol has the maximum throughput for the considered TCP variants, and the DSR protocol has the minimum throughput.





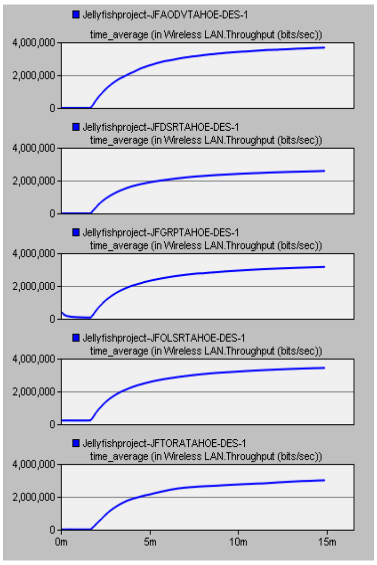


Figure 5a WLAN Throughput, Tahoe   Figure 5b WLAN Throughput, Reno   Figure 5c WLAN Throughput, SACK

Finally, Figure 6a shows the page response time and Figure 6b shows the TCP delay for the AODV routing protocol , using the three TCP variants respectively. The figures show that the SACK TCP variant outperforms Tahoe and Reno TCP variants since SACK has minimum page response time and minimum TCP delay.

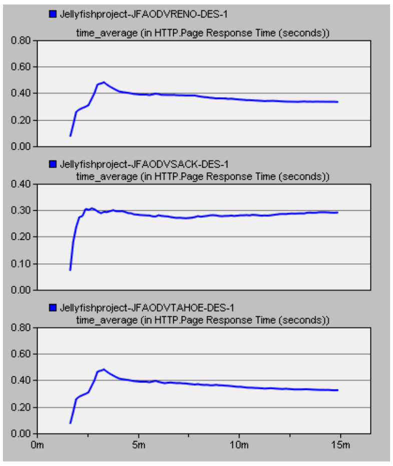


Figure 6a page response time, AODV

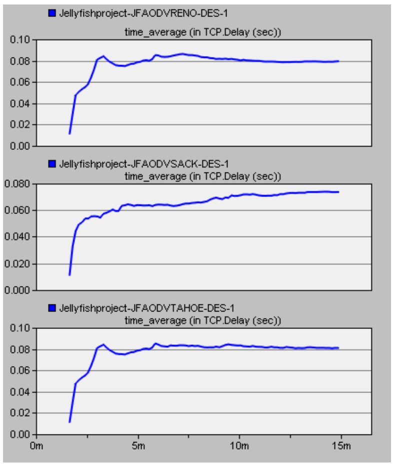


Figure 6b TCP delay, AODV

5. Conclusion

In this paper, we have compared the performance of AODV, DSR, TORA, OLSR, and GRP routing protocols under the attack of jellyfish delay variance , using the OPNET simulator. This comparison is obtained by reducing the flow rate of the attacker nodes in the MANET network according to the attack simulation model.

The experimental results show that the performance of routing protocols under investigation is degraded under jellyfish delay variance attacks in the MANET network. Further, the simulation results show that the AODV routing protocol outperforms the remaining routing protocols, and hence the AODV routing protocol is more resistant to the jellyfish delay variance attack. Also, the experimental results show that the SACK TCP variant performs better than Tahoe and Reno TCP variants in this work.

References

1. Ablhasan M., T. Wysocki and E. Dutkiewicz, (2004), " A review of routing protocols for mobile ad hoc networks", Elsevier Journal of Ad Hoc Networks, Vol. 2, No. 1, pp. 1-22.
2. Conti M., Giordano S., (2007), " Multi-hop ad hoc networking: the theory", IEEE Communication Magazine, Vol. 45(4), pp. 78-86.
3. Dulaimi, L. A. K., Ahmad R. B., Yaakob N., Shamsuddin S. N. W., and Elshaikh M, (2019), " Behavioral and performance jellyfish attack", Indonesian Journal of Electrical Engineering and Computer Science Vol. 13, No. 2, pp. 683-688.
4. Fall K, Floyd S, (1996), "Simulation Based Comparison of Tahoe, Reno and SACK TCP" , computer communication review, 26 (3), pp. 5-22.
5. Floyd S., (1996), "Issues of TCP with SACK", Technical report, pp. 1-5.
6. Jacobson V., (1988), "Congestion Avoidance and Control", SIGCOMM Symposium on Communications Architectures and Protocols, pp. 314–329.
7. Jacobson. V., Floyd, S., (1993) " Random early detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking 1(4), pp. 397–413
8. Kaur A, Singh T. P., (2015), "Securing MANET from jellyfish attack using selective node participation approach, International Journal of Engineering and Technical Research (IJETR), Vol. 3, pp. 80-83.
9. Laxmi V, Lal C, Gaur M. S., Mehta D, (2015), "Jellyfish attack: Analysis, detection and countermeasure in TCP-based MANET", Journal of information security and applications, Vol. 22, pp. 99-112.
10. Mishra D., More A., Maru P., Dandekar A., Vaity N., (2017), "Analysis and Avoidance Of Jellyfish Attack", International Journal of Technical Research and Applications, Special Issue 43, pp. 86-89.
11. Sachdeva S. and Kaur P., (2016), " Detection and Analysis of Jellyfish Attack in MANETs", International Conference on Inventive Computation Technologies (ICICT), Vol. 2, pp. 1-5.
12. Sajjad M, Saeed K., Hussain T., Abbas A. W., Khalil I., Ali I., Gul N., (2019), " Impact of Jellyfish Attack on the Performance of DSR Routing Protocol in MANETs", Journal Of Mechanics Of Continua And Mathematical Sciences, Vol.14, No.4, pp. 132-140.
13. Stevens W. R., (1994), "TCP/IP Illustrated, Volume 1: The Protocols", Addison Wesley.
14. Wazid M., Sachan R. S., Goudar R. H., "Measuring the Impact of Jellyfish Attack on the Performance of Mobile Ad Hoc Networks using AODV Protocol", Proc. Int. Conf. on Computational Intelligence and Information Technology, CIIT, Elsevier, pp. 293-299.

تأثير هجوم Jellyfish على بروتوكولات التوجيه في الشبكات المتنقلة (MANET) القائمة على بروتوكول TCP

خالد أحمد عبود عمر
قسم علوم وهندسة الحاسب، كلية الهندسة، جامعة عدن
عنوان البريد الإلكتروني: k_abood@hotmail.com

DOI: https://doi.org/10.47372/uajnas.2023.n1.a09

الملخص

الشبكة المتنقلة (MANET) هي شبكة ذاتية التنظيم خالية من البنية التحتية وتتكون من الأجهزة اللاسلكية المحمولة التي يمكن نشرها من أجل الاتصال. نظرًا لوسط الاتصال اللاسلكي غير الآمن، وعملية الاتصال ذات التوجيه المتعدد، والسلوك الديناميكي للعقد في MANET، فإن بروتوكولات التوجيه معرضة لهجمات أمنية مختلفة، مثل هجمات Jellyfish. تستهدف عقدة Jellyfish في شبكة MANET القائمة على بروتوكول TCP وتستغل آلية عمل البروتوكول لتقليل أداء الاتصال في الشبكة. يصعب اكتشاف هذا الهجوم لأنه يقوم على منهجية متوافقة مع عمل البروتوكول TCP.

في هذا البحث، نقوم بتقييم أداء بروتوكولات التوجيه الآتية AODV، DSR، TORA، GRP، وOLSR في إطار هجمات تباين تأخير Jellyfish لشبكات MANET القائمة على بروتوكول TCP. علاوة على ذلك، يتم مقارنة TAHOE و RENO و SACK لبروتوكول TCP. تتم محاكاة بروتوكولات التوجيه هذه باستخدام محاكي OPNET لمقارنة أدائها باستخدام مقاييس أداء محددة على الشبكة. تظهر النتائج التجريبية أن بروتوكول AODV يعمل بشكل أفضل من بروتوكولات DSR و TORA و OLSR و GRP تحت هجوم تباين تأخير Jellyfish . علاوة على ذلك، يعمل SACK TCP بشكل أفضل من بقية اصدارات بروتوكول TCP الأخرى في ظل هجوم تباين تأخير Jellyfish .

الكلمات المفتاحية: بروتوكولات التوجيه، MANET، TCP، هجمات Jellyfish.