A Novel Metric for Detection of Jellyfish Reorder Attack on Ad Hoc Network

B. B. Jayasingh[1] and **B. Swathi**[2]

Abstract - Ad Hoc networks are susceptible to many attacks due to its unique characteristics such as open network architecture, stringent resource constraints, shared wireless medium and highly dynamic topology. The attacks can be of different types out of which denial of service is one of the most difficult attacks to detect and defend. Jellyfish is a new denial of service attack that exploits the end to end congestion control mechanism of TCP (Transmission Control Protocol) which has a very devastating effect on the throughput. The architecture for detection of such attack should be both distributed and cooperative to suit the needs of wireless ad-hoc networks that is every node in the wireless ad-hoc network should participate in the intrusion detection. We intend to develop an algorithm that detects the jellyfish attack at a single node and that can be effectively deployed at all other nodes in the ad hoc network. We propose the novel metric that detects the Jellyfish reorder attack based on the Reorder Density which is a basis for developing a metric. The comparison table shows the effectiveness of novel metric, it also helps protocol designers to develop the counter strategies for the attack.

Index Terms - jellyfish attack, Percentage of Late Packets (P_L), Mean Displacement of Packets (M_D), Mean displacement of late packets (M_L), Reorder Entropy(E_R).

1. INTRODUCTION

Dynamic topology, distributed operation, and resource constraints are some of the unique characteristics that exist in the ad hoc networks, which inevitably increase the vulnerability of such network. Many characteristics might be used to classify attacks in the ad hoc networks. MANET not only inherits all the security threats faced in both wired and wireless networks, but it also introduces security attacks unique to itself [1]. As people will be encouraged to use a secured network, it is important to provide MANET with reliable security mechanisms if we want to see this exciting technology become widely used in a next few years. Before the development of any security measure to secure mobile ad hoc networks, it is important to study the variety of attacks that might be related to such networks. With the knowledge of some common attack issues, researchers might have a better understanding of how mobile ad hoc networks could be threatened by the attackers, and thus might lead to the development of more reliable security measures in protecting them.

Routing is one of the most vital mechanisms in the ad hoc networks. Improper and insecure routing mechanisms will not only degrade the performance of the ad hoc networks, but will also render such networks vulnerable to many security attacks. One of the basic elements in the routing mechanism is the routing message, which is used to establish and maintain relationships between nodes in the networks. The importance of the routing message has made it a main target by the attackers to launch attacks against the ad hoc networks [2, 3, 16]. In designing security mechanisms for mobile ad hoc networks, one must consider the attacks variations as well as the characteristics of the attacks that could be launched against the ad hoc networks.

The first JF attack is the packet reordering attack. TCP has a well-known vulnerability to reordered packets due to factors such as route changes or the use of multi-path routing, and a number of TCP modifications have been proposed to improve resistance to misordering including TCP Stack [5] and reorder robust TCP [6].However, no TCP variant is strong enough to resist such malicious and persistent reordering as employed by the JF misordering attack. The mechanism that the jellyfish node uses for attack consists of delivering all received packets, but in scrambled order by placing them in a reordering buffer instead of the canonical FIFO order i.e.JF nodes maliciously re-order packets. Consequently, such persistent re-ordering of packets will result in near zero goodput, despite having all transmitted packets delivered.

We intend to develop a detection algorithm that can detect the jellyfish Reorder attack at a single node. The attack can be effectively detected by deploying the same detection mechanism at all nodes in the ad hoc network. We are assuming that there is no packet loss and duplication. The algorithm detects the persistent reordering employed by the Jellyfish node [17].The algorithm takes sequence number, acknowledgment number and Receive index as inputs. The sequence number and Receive index are used to calculate a value called Reorder density which is the basis for developing a metric that can detect the reordering of the packets that is done by the attacker. The algorithm also checks whether the acknowledgment number, that are generated when the packets are received, are reordered. Thus the new metric and the detection of reordered acknowledgment numbers can detect the Jellyfish Reorder attack effectively.

The rest of the paper focuses on the existing metrics for the packet reordering calculation in the section 2 and the calculations we do by using mathematical formulae's in section 3. We summarized the calculation in the section 4 and the lessons learned able to find a new metric that is proposed in section 5. The comparisons made between the existing metrics

[1]Dept. of IT, CVR College of Engineering, Vastunagar, Mangalpalli (V), Ibrahimpatan (M),
RR District – 501510, Hyderbad (AP), India.
[2]B. Tech (IV-IT), CVR College of Engineering, Vastunagar, Mangalpalli (V), Ibrahimpatan (M),
RR District – 501510, Hyderbad (AP), India.
E-Mail : [1]bbjayasingh9@rediffmail.com
[2]andbswathi.reddy25@gmail.com

calculation and the proposed metric is discussed in section 6, following with the conclusion in section 7.

2. EXISTING METRICS FOR REORDERING

Many attackers disobey protocol rules, whereas jellyfish obeys the protocol rules and hence is difficult to detect until after the sting. Jellyfish target closed-loop flows. One example of such a closed loop flow is the TCP flow .Just like any IP service, Jellyfish node can drop packets, Reorder packets, Delay / jitter packets but it is done in a malicious way. Since the Jellyfish Attack maintains compliance with all control plane and data plane protocols, it is difficult to distinguish from congestion and packet losses that occur naturally in a network, and therefore detection and diagnosis is hard, costly ,resource-consuming and time consuming [4].

There are several simple, derived metrics to monitor packet reordering in a network or an end to end connection. In this section, we discus the existing metrics for determining the reordering such as Percentage of Late Packets, Mean Displacement of Packets and the Reorder entropy [8, 9, 10].

We assume that there is no packet loss and duplication for the calculation of these metrics**.** However the calculation of these metrics is based on RD value which is obtained by applying the algorithm in [11, 12, 13, 14,]. Reorder Density (RD) is a discrete density function that is used to detect and capture the nature of reordering in a packet stream.

2.1 Percentage of Late Packets (P_L)

To capture the lateness of packets from their original positions the percentage of late packets is defined as the percentage of packets that exhibit lateness with respect to their expected position, as given by the receive index.

$$P_L = \sum_{i=+1}^{i=D_r} RD[i]$$

P_L =0 corresponds to the case where all the packets are in order. For a sequence with packet reordering, P_L >0.

2.2 Mean Displacement of Packets (M_D)

Packet reordering is associated with two types of events, lateness events and earliness events. In a lateness event, the corresponding displacement is always positive. And a negative displacement is mapped with the earliness event. When calculating the mean displacement of packets, if both late and early packets are included, from equation 1, the mean displacement is zero for all cases.

Equation 1:

$$\sum_{i} (i \times RD[i] =0$$

Therefore, the mean displacement, when all packets are taken together, is not useful. On the other hand, one can consider the magnitude of displacement of packets, and divide it by the total number of packets to define a mean displacement M_D:

Mean Displacement (M_D)

$$M_D = \left| \sum_{i=-D_r}^{i=+D_r} (|i| \times RD[i]) \right| / \left| \sum_{i=-D_r}^{i=+D_r} RD[i] \right|$$

Mean displacement of late packets (M_L)

RD[*i*] refers to the probability that a packet arrives *i* packets away from its expected position. Thus considering only the late packets, the mean displacement of late packets is given as:

$$M_L = [\sum_{i=1}^{i=+D_r} (i \times RD[i])] /[\sum_{i=1}^{i=D_r} RD[i]]$$

Similarly, the **Mean displacement for earliness** is:

$$M_E = \left|[\sum_{i=-1}^{i=-D_r} (i \times RD[i])] /[\sum_{i=-1}^{i=-D_r} RD[i]]\right|$$

Note here that we divide the total positive (negative) displacement by the total number of late (early) packets. Both M_L and M_E are always none negative values, and $M_L=M_D/2P_L$, $M_E= M_D/2P_E$.

2.3 Reorder Entropy(E_R)

Entropy is a concept that is used to define the randomness or the disorder. As RD is a discrete probability distribution, that of packet displacement (a form of disorder), we define reorder entropy as:

$$E_R = (-1) \times \sum_{i=-D_r}^{i=+D_r} (RD[i] \times \mathrm{Log}_e RD[i])$$

RD [0] = 1

, when no packet ordering is present, the reorder entropy is equal to zero. On the other hand, the packet sequence has the most variance, when packets are displaced uniformly with equal probabilities [15].

3. REORDERING CALCULATION

This section deals with calculation of existing metrics for different number of packets, with and without reorder.

S denotes the Sequence Number of packets. Receive Index (RI) is a value assigned to a packet as it arrives at its destination, according to the order of arrival. Displacement (D) is the difference between RI and the sequence number of the packet. Displacement Frequency FD[k] is the number of arrived packets having a displacement of k. RD is defined as the distribution of the Displacement Frequencies FD[k], normalized with respect to N'. N' is equal to the sum (FD[k])[14].

3.1 Three Packets with Reorder

Consider the sequence of packets (2,3,1). The Tables 1 and 2 show the computational steps when the RD algorithm is applied to the above sequence.

S	2	3	1
RI	1	2	3
D	-1	-1	2
FD[D]	1	2	1

Table 1: showing the calculation of D and FD[D]

The last row (FD[D]) represents the current frequency of occurrence of the displacement D, e.g., column 2 indicates FD[-1] = 2 while column 3 indicates FD[2] = 1. The final sets of values for RD are shown in Table 2.

D	-1	2
FD[D]	2	1
RD[D]	0.66	0.33

Table 2: showing the calculation of RD[D]

P_L= RD[2]=0.33
M_D=(1*0.66+2*0.33)/(0.66+0.33)=1.32
M_L= (2*0.33)/(0.33)=2
M_E= |-1*0.66|/0.66=1
E_R= 0.63

3.2 Three Packets without Reorder

Consider the sequence of packets (1,2,3). The Tables 3 and 4 show the computational steps when the RD algorithm is applied to the above sequence.

S	1	2	3
RI	1	2	3
D	0	0	0
FD[D]	1	2	3

Table 3: showing the calculation of D and FD[D]

The last row (FD [D]) represents the current frequency of occurrence of the displacement D, e.g., column 2 indicates FD[0] = 2 while column 3 indicates FD[0] = 3. The final sets of values for RD are shown in Table 4.

D	0
FD[D]	3
RD[D]	1

Table 4: showing the calculation of RD[D]

P_L= 0, M_D= 0, M_L= 0, M_E= 0, E_R= 0

When the packets sent are in the sequential order i.e. when there is no reordering Percentage of late packets (P_L), Mean Displacement of Packets (M_D), Mean displacement of late packets (M_L), Mean displacement for earliness (M_E) and Reorder Entropy are zero. But when there is any reordering then these values will not be zero and will have some distinct value.

3.3 Five Packets with Reorder

Consider the sequence of packets (5,2,3,1,4). The Table 5 and 6 show the computational steps when the RD algorithm is applied to the above sequence.

S	5	2	3	1	4
RI	1	2	3	4	5
D	-4	0	0	3	1
FD[D]	1	1	2	1	1

Table 5: showing the calculation of D and FD[D]

The last row (FD[D]) represents the current frequency of occurrence of the displacement D, e.g., column 3 indicates FD[0] = 2 while column 4 indicates FD[3] = 1. The final sets of values for RD are shown in Table 6.

D	-4	0	1	3
FD[D]	1	2	1	1
RD[D]	0.2	0.4	0.2	0.2

Table 6: showing calculation of RD[D]

P_L= 0.2+0.2=0.4
M_D=(4*0.2+*0.2+3*0.2)/(0.2+0.4+0.2+0.2)=1.6
M_L= (1*0.2+3*0.2)/(0.2+0.2)=2
M_E= (4*0.2)/0.2=4
E_R= 1.33

3.4 Five Packets without Reorder

Consider the sequence of packets (1,2,3,4,5). The Tables 7 and 8 show the computational steps when the RD algorithm is applied to the above sequence.

S	1	2	3	4	5
RI	1	2	3	4	5
D	0	0	0	0	0
FD[D]	1	2	3	4	5

Table 7: showing the calculation of D and FD[D]

The last row (FD[D]) represents the current frequency of occurrence of the displacement D, e.g., column 3 indicates FD[0] = 3 while column 4 indicates FD[0] = 4. The final sets of values for RD are shown in Table 8.

D	0
FD[D]	5
RD[D]	1

Table 8: showing the calculation of RD[D]

P_L= 0, M_D= 0, M_L= 0, M_E= 0, E_R= 0

When the packets sent are in the sequential order i.e. when there is no reordering Percentage of late packets (P_L), Mean Displacement of Packets (M_D), Mean displacement of late packets (M_L), Mean displacement for earliness (M_E) and Reorder Entropy are zero. But when there is any reordering then these values will not be zero and will have some distinct value.

4. SUMMARY TABLES

In this section we show the table containing the D and RD(D) for different number of packets. D and RD(D) for 3 and 5 number of packets are obtained from the table2 and table6 respectively. Similarly D and RD(D) can be calculated for 8 and 10 packets as shown in Table 9 .

3pkts	D	-1	2				
	RD(D)	0.66	0.33				
5pkts	D	-4	1	0	3		
	RD(D)	0.2	0.2	0.4	0.2		
8pkts	D	-2	-1	0	1	2	
	RD(D)	0.125	0.125	0.5	0.125	0.125	
10pkts	D	-6	-4	0	2	5	7
	RD(D)	0.1	0.2	0.4	0.1	0.1	0.1

Table 9: showing the D and RD(D) for 3,5,8,10 packets

The RD values shown in Table 9 are used to calculate the existing metrics i.e. P_L, M_D, M_L, M_E, E_R for different number of packets as shown in Table 10. Mean displacement of packets is calculated for earliness (M_E), lateness (M_L) and combination of both (M_D).

		P_L	M_D	M_L	M_E	E_R
3 packets	With Reorder	0.33	1.32	2	1	0.63
	Without Reorder	0	0	0	0	0
5 packets	With Reorder	0.4	1.6	2	4	1.33
	Without Reorder	0	0	0	0	0
8 packets	With Reorder	0.25	0.75	1.5	1.5	1.386
	Without Reorder	0	0	0	0	0
10 packets	With Reorder	0.3	2.8	4.6	4.6	1.609
	Without Reorder	0	0	0	0	0

Table 10: showing the existing metric values

5. PROPOSED METRIC

Though many Intrusion Detection Systems are available, they are not suitable to detect the attack in ad hoc network because wireless ad-hoc networks don't have any fixed infrastructure and since almost all of current network based IDS sit on the network gateways and routers and analyze the network packets passing through them, these type of network based IDS are rendered ineffective for the wireless ad-hoc networks [7]. Anomaly Detection models of IDS cannot be used for wireless ad-hoc networks, since the separating line between normalcy and anomaly is obscure. A node that transmits erroneous routing information (fabrication) can be either a compromised or is currently out of sync due to volatile physical movement. Hence in wireless ad-hoc networks it is difficult to distinguish between false alarms and real intrusions. So, we have developed a novel metric for detection of the Jellyfish Reorder attack. The metric is calculated by multiplying frequency and reorder density for all the displacements and then taking their summation i.e. ∑ FD*RD. Let us analyze this metric for different number of packets in both cases i.e. with reorder and without reorder cases.

5.1 Three Packets with Reorder

With reference to Table 2 there are two Displacement frequencies i.e.2 and 1 and two Reorder density values i.e.0.66 and 0.33.The corresponding displacement frequencies and reorder density are multiplied and then summed which gives a value of 1.65 as shown below.

∑ FD*RD = 2*0.66+1*0.33=1.65

This value is between 1 and 3 i.e. number of packets.

5.2 Five packets with reorder

With reference to Table 6 there are four Displacement frequencies i.e.1, 1, 2 and 1 and four Reorder density values i.e0.2, 0.2, 0.4 and 0.2.The corresponding displacement frequencies and reorder density are multiplied and then summed which gives a value of 1.4 as shown below.

∑ FD*RD = 1*0.2+1*0.2+2*0.4+1*0.2=1.4

This value is between 1 and 5 i.e. number of packets.

5.3 Three Packets without Reorder

With reference to Table 4 there is only one Displacement frequency value i.e.3 and one Reorder density value i.e.1.The corresponding displacement frequency and reorder density is multiplied and then summed which gives a value of 3 as shown below.

∑ FD*RD = 3*1=3

This value is equals to 3 i.e. number of packets

5.4 Five Packets without Reorder

With reference to Table 8 there is only one Displacement frequency value i.e.5 and one Reorder density value i.e. 1 .The corresponding displacement frequency and reorder density is multiplied and then summed which gives a value of 5 as shown below.

∑ FD*RD = 5*1=5

This value is equals to 5 i.e. number of packets.

After analyzing the value of the proposed metric under 2 cases for different number of packets it was found that the value of the metric (∑ FD*RD)

i) is greater than equal to one and less than the number of packets when there is reordering.

1<= ∑ FD*RD<Number of packets

ii) is always equals to number of packets when there is no reordering.

∑ FD*RD=Number of packets

6. COMPARISION OF METRICS

The values of the existing metrics are always zero when there is no reorder whereas the proposed metric value is equals to number of packets when there is no reordering. The calculation of the proposed metric is computationally simple because it involves less calculation i.e. it involves simple algebraic operation of addition and multiplication. So the complexity of the algorithm that calculates this proposed metric for determining reordering is comparatively less when compared to the previous metrics.

		P_L	M_D	M_L	M_E	E_R	New Metric $\sum$ **FD*RD**
3 packets	With Reorder	0.33	1.32	2	1	0.63	1.65
	Without Reorder	0	0	0	0	0	3
5 packets	With Reorder	0.4	1.6	2	4	1.33	1.4
	Without Reorder	0	0	0	0	0	5
8 packets	With Reorder	0.25	0.75	1.5	1.5	1.386	2.5
	Without Reorder	0	0	0	0	0	8
10 packets	With Reorder	0.3	2.8	4.6	4.6	1.609	2.4
	Without Reorder	0	0	0	0	0	10

Table 11: showing all the metrics for different number of packets.

7. CONCLUSION

Jellyfish attack is protocol complaint and passive. So it is difficult to detect this attack until after the sting. Though there are existing metrics which can detect the Jellyfish Reorder attack to some extent by incorporating them in the algorithm, but the complexity involved in calculating these metrics increases with the increase in number of packets. The metric that we have proposed to be used in algorithm to detect the Jellyfish Reorder attack is highly effective because the metric is quite simple, efficient and also less time consuming

REFERENCES

[1] T. Karygiannis and L. Owens, "Wireless Network Security, 802.11, Bluetooth and Handheld Devices," NIST Publication, p. 800(48), November 2002.

[2] H. Li, Z. Chen and X. Qin, "Secure Routing in Wired Networks and Wireless Ad Hoc Networks," Univ. of Kentucky, Department of Computer Science, Term-paper, 2003.

[3] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Networks," in Proc. of 2002 IEEE International Conference on Network Protocols (ICNP), pp. 778-89, Nov. 12-15, 2002.

[4] Jean-Pierre Hubaux , Edward W. Knightly,Impact of Denial of Service Attacks on Ad Hoc Networks Imad Aad, IEEE/ACM Transactions on Networking, Publication Date: Aug 2008 Volume: 16.

[5] K. Fall and S. Floyd, "Simulation-based comparison of Tahoe, Reno and SACK TCP," ACM Computer Communications Review, vol. 5, no. 3, pp. 5–21, July 1996.

[6] M. Zhang, B. Karp, S. Floyd, and L. Peterson, "RR-TCP: A reordering robust TCP with DSACK," in Proceedings of IEEE ICNP, 2003.

[7] Intrusion detection in wireless ad-hoc networks, Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, Massachusetts, United States, Year of Publication: 2000.

[8] B. Ye, A. P. Jayasumana and N. Piratla, "On Monitoring of End-to-End Packet Reordering over the Internet," Proc.Int. Conference on Networking and Services (ICNS'06), Santa Clara, CA, July 2006.

[9] Banka, T., Bare, A. and Jayasumana, A., "Metrics for Degree of Reordering in Packet Sequences," Proc. IEEE 27Local Computer Networks Conf, Nov. 2001, pp. 333-342.

[10] Bare, A. A., "Measurement and Analysis of Packet Reordering," Masters Thesis, Dep. Computer Science,Colorado State University, 2004.

[11] Jayasumana, A., Piratla, N. M., Bare, A. A., Banka, T.,Whitner R., and McCollom, J.,"Reorder Density Function -A Metric for Packet Reordering Measurement," IETF draft.

[12] Piratla, N. M., Jayasumana, A. P., and Bare, A. A., "RD:A Formal, Comprehensive Metric for Packet Reordering,"Proceedings Fourth IFIP-TC6 Networking Conference(Networking 2005), LNCS 3462, Ontario, May 2005, 78-89.

[13] Piratla, N. M. , Jayasumana A. P. ,and Bare A. A., "A Comparative Analysis of Packet Reordering Metrics," Proc. IEEE/ACM 1st Int. Conf. Communication System Softwareand Middleware (COMSWARE 2006), New Delhi, Jan. 2006.

[14] A. Jayasumana, N. Piratla, T.Banka, A. Bare, R. Whitner. "Improved Packet Reordering Metrics",Network Working Group, Colorado State University, June 2008.

[15] Shannon C. E., "A Mathematical Theory of Communication", Bell Sys. Tech. Journal, vol. 27, 1948.

[16] Kristoffer Karlsson IT3, Billy HoIT3,Ad hoc networks: Overview, applications and routing issues, Chalmers University of Technology.

[17] S. Muthukumar, C. Arunkumar, G. Ramesh, Enhancing The Transport Mechanisms In The Presence Of Packet Re-Ordering In Tcp-Pr Algorithm, Dept. of MCA, Adhiparasakthi Engineering College, Melmaruvathur, Tamilnadu

Continued from Page No. 173

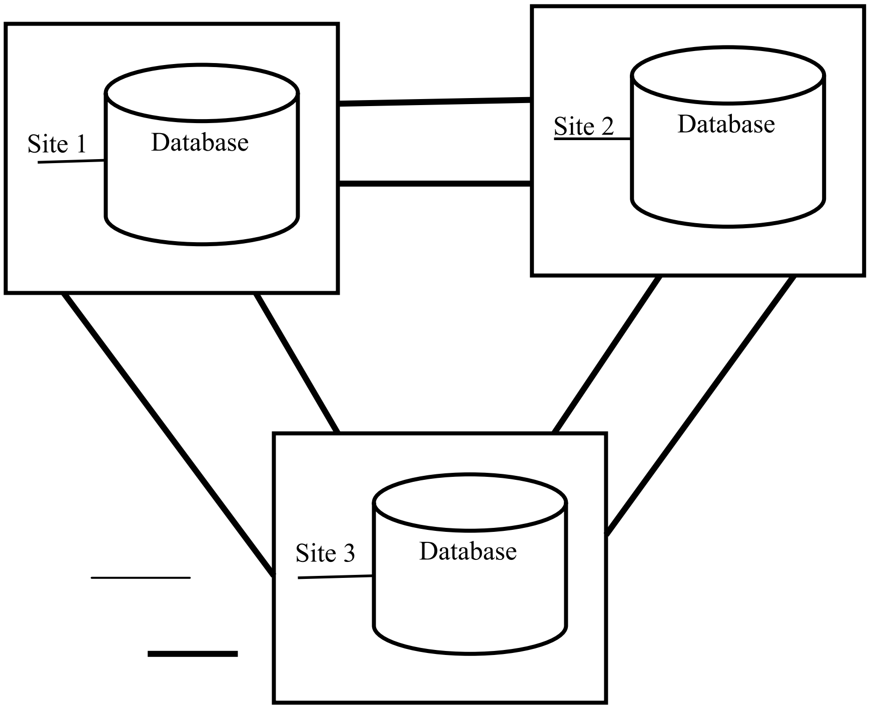

Figure 2: Working Scheme for Replication