International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.5, Sep-Oct. 2012 pp-3633-3636 ISSN: 2249-6645

A Novel Schema for Detecting Malicious Packet Losses

M. Kiran kumar[1], A. Sai Harish[2]
*(M.Tech, kiet engineering college, Andhra Pradesh, India)
** (Assistant professor, kiet engineering college, Andhra Pradesh, India)

ABSTRACT: *Detecting malicious dropping packets is a crucial issue in networks to minimize various security attacks such as blackhole, greyhole, and wormhole attacks. All networks drop the packets in the presence of collisions, channel errors and the network traffic exceeds its capacities. All the existing detection algorithms have addressed this issue by using user-defined threshold value. But these attacks could not able to solve it because too many dropped packets imply malicious intent. To address this problem in this paper we proposed a model to monitor a node for detecting malicious packet dropping. To evaluate the performance of the proposed algorithm we used NS2 simulator. Our experimental revels that the proposed algorithm performed very well to detect malicious packet drops due to the collisions, channel errors and heavy traffic.*

Keywords: *channel errors, energy drain attack, malicious dropping attack, collisions, blackhole*

I. INTRODUCTION

The Internet environment is not a safe place. Due to the unsecured nodes in the internet even well protected nodes may be face denial-of-service attacks, blackhole, greyhole, and wormhole attacks [1]. However, such attacks to a node are widely understood, it is less well appreciated that the network infrastructure itself is subject to constant attack as well. In this paper, we propose a method to find out where the loss occurred. If a hacker gains the control of a router, he may disturb the communication by dropping or manipulating the transferred packets. Network load can be disturbed by routers, refusing to serve their advertised routes, announcing nonexistent routes, or simply failing to withdraw failed routes, as a result of either malfunction or malice which is described in fig 1. The main idea of detecting malicious packet loss is finding where the packet loss has occurred in the network due to the presence of collisions, channel errors or heavy traffic. The attacker may disturb the packet forwarding by dropping packets routed to it by its neighbors. Mike Lynn's demonstrated how Cisco routers can be compromised via simple software vulnerabilities. Once a router has been compromised in such a fashion, an attacker may interpose on the traffic stream and manipulate it maliciously to attack others by selectively roping, modifying, or rerouting packets.

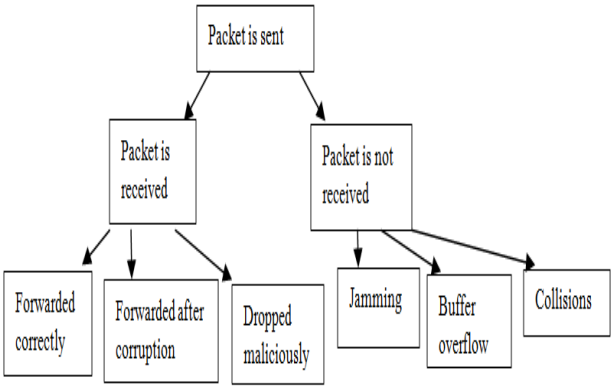

Fig 1: overview of packet loss

Several authors have proposed various protocols to detect packet manipulations, based on validating the traffic transmitted by one router is received unmodified by another [2], [3], [4]. All the proposed algorithms struggle in interpreting the absence of traffic. While a packet that has been modified in transit represents clear evidence of tampering, a missing packet is inherently ambiguous: it may have been explicitly blocked by a compromised router or it may have been dropped benignly due to network congestion. The modern routers drop the packets due to high network traffic and the widely used Transmission Control Protocol (TCP) is designed to cause such losses as part of its normal congestion control behavior [8]. All the existing traffic validation systems must inevitably produce false positives for benign events and/or produce false negatives by failing to report real malicious packet dropping [5]. To overcome this problem, in this paper we proposed a router detection protocol it dynamically infers the precise number of congestive packet losses. If the congestion is avoided, subsequent packet losses can be safely attributed to malicious actions. Our proposed protocol automatically predicts congestion in a systematic manner and takes necessary actions to avoid it. We evaluated the performance of the protocol using NS2 simulator and experimental results relived that the proposed protocol capable of accurately resolving extremely small and fine-grained attacks.

The rest of the paper is organized as section 2: discuss about the related work, section 3: presents the network model, section 4: discuss about the performance metrics, section 5: discuss about the proposed algorithm, section 6: discuss about the experimental setup and section 7: concludes the paper.

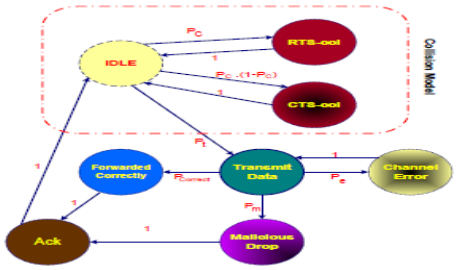

Fig 2: state diagram of packet loss

II. RELATED WORK

H. Ma [9] classified the types of interference which impacts the packet loss in networks. In Type-1 interference, the interference signal arrives before the desired signal. While in Type-2 interference, the interference signal arrives after the desired signal, and in the case of collisions both signals arrive at the same time. Statistical methods have been by used various authors to determine the packet loss rate at each node based on interference type. Pang [2] presented a method to distinguish between packet loss due to collisions and link errors. The main idea is that shorter RTS/CTS and MAC headers in 802.11 are less vulnerable to errors than data. Thus, during the RTS/CTS access procedure, errors are assumed to be due to collisions. If the node receives the CTS frame but not ACK frame then the transmission has more likely failed due to a channel error. However, if an RTS/CTS frame is not received, then the transmission more likely failed due to a collision. If a packet with a corrupted header is received then the receiver will not send anything and the sender assumes a collision as a timeout is occurred. If the data is corrupted, the receiver sends a NAK frame to the sender. But the sender assumes the packet has lost due to channel errors. J. Kim [3] proposed collision aware rate adaptation scheme based on RTS probing to differentiate collisions from channel errors. Malone [5] presented an algorithm to estimate packet losses caused by collisions and by channel errors. This algorithm needs some statics knowledge such as the number of successful transmissions out of the total transmissions over some period of time and the number of slots in which the station does not transmit. S. Marti [10] proposed a watchdog scheme for detecting malicious packet dropping attacks to distinguish between types of packet losses. M. Just [6] has used probes disguised as normal packets to detect malicious nodes and F. Anjum [7] used a centralized authority that receives reports on statistics of various IP flows. But these techniques could not able to distinguish between causes for packet loss. Appenzeller [25] has explored the question of "How much buffering do routers need?" A widely applied rule-of-thumb suggests that routers must be able to buffer a full delay bandwidth product. Due to congestion control effects, the buffering is proportional to the square root of the total number of TCP flows. To achieve this, the author presented a model of buffer occupancy as a function of TCP behavior.

III. NETWORK MODEL

A state diagram which is shown in fig 2, in the idle state node will be waiting for a packet to send. When a packet arrives, if the medium is free then the node sends an RTS packet. The system may move to an RTS-collision state when two or more nodes that are within each other's range transmit an RTS at the same time with probability P_c. The system may move to a CTS-collision state when a hidden node transmits something that collides with the CTS sent by the receiving node and the CTS-collision will occur with probability $(1- P_c)P_c$. A node will transmit a packet only if it receives a CTS reply to its RTS. The probability that a data packet is transmitted Pt, is:

$$P_t = (1 - P_C)(1 - P_C.(1 - P_C)) = 1 - 2P_C + 2{P_C}^2 - {P_C}^3 \quad (1)$$

The size of the RTS/CTS packets is small. After a packet is transmitted it will be either forwarded with probability $P_{Correct}$, lost due to channel errors with probability P_e, or maliciously dropped or not ACK-ed with probability P_m. Packets that are maliciously dropped may or may not be acknowledged. In our proposed model we assume that dropped packets will be acknowledged. Node in the Malicious Drop state will move to the ACK state if an ACK message is sent or return to the Transmit state if no ACK is sent. Packets lost due to channel errors will also not be acknowledged by the receiver, and will be retransmitted after some timeout. That is, when the Channel Error state is reached, the node will return back to the Transmit state. If a packet is transmitted by the sender, not dropped due to any errors then the packet will be received or forwarded correctly. Thus the probability to forward a packet correctly $P_{Correct}$ can be computed as:

$$P_{Correct} = P_t(1 - P_e)(1 - P_m) \quad (2)$$

IV. PERFORMANCE METRICS

Collisions: various authors analyzed the nature of Collisions in 802.11. Bianchi [11] and H. Wu [12] have used Markov chain model to find number of collisions in network. In [13], linearization proposed a method to find an approximate value for P_C based on contention window W and the number of nodes n and is given by:

$$P_C = \frac{2W(n-1)}{(W+1)^2 + 2W(n-1)} \quad (3)$$

Later X. Wang [14] presented a approach based on probability of a node sends the packet. Pc = 1- $(1-T)^{n-1}$ where n is the average number of contending nodes and T is the average probability that a node sends a packet T is denoted as 1/w and the probability of collision can defined as :

$$P_C = 1 - (1 - 1/W)^{n-1} \quad (4)$$

Finally Using the number of RTS and CTS packets that were counted during a time window w the probability that a packet was lost due to collision can defined as :

$$P_C = (\#RTS - \#CTS)/\#RTS \quad (5)$$

Channel errors: TN. Gupta [15] assumed a wireless channel with Markov chain model to analyze the performance of 802.11. The duration of wireless channel good and bad state is defined as $\lambda_g^{-1}$and λ_b^{-1}, respectively. J. N. ArauzIn [16] performed several experiments and modeled 802.11 links to find the values of λ_g^{-1} and λ_b^{-1} for several PHY layer bit rates and three SNR levels (high, medium, and low).

Energy Drain Attack: In this attack, malicious node intends to drain the sender's battery by not sending ACKs and making the sender retransmit the packet several times before sending an ACK. When the malicious node responds with an ACK to a data packet, the sender node will assume that the packet has been received and forwarded correctly. In this case, the sender node estimates $P'_{Correct}$ as:

$$P'_{Correct} = \#ACK/\#RTS \qquad (6)$$

Malicious node may drop the ACK-ed packet and not relay it to the next hop, because the attack is directed towards draining the battery, the ultimate fate of the ACK-ed packet is not relevant.

Malicious Dropping Attack: In this attack, malicious receiving node may send an ACK message upon receiving a packet to be relayed and not forward the packet to the next hop. There are two possible ways to know if the acknowledged packet was forwarded or not, either by monitoring the node using overhearing capability or by having feedback from intermediate nodes which include communication overhead. Hence, in the proposed model we prefer to monitor the receiving node.

V. PROPOSED ALGORITHM

The proposed algorithm used to detect if the neighbor node maliciously dropping packets.

Step 1: Node A will count the RTS messages it sent to node B during some time window w and also the CTS messages received from node B during the same time.

Step 2: Node A will use the model previously described for the value of Pe based on the link SNR. We assume symmetric links, and thus the SNR is expected to be equal at the sending and receiving sides.

Step 3: If the goal of node A is only to prevent energy drain attacks then compute $P'_{Correct.}$

Step 4: If the goal of node A to detect malicious packet dropping then it will use monitoring through overhearing to get an estimate of $P'_{Correct}$.

Step 5: Node A calculate the percentage of packets being maliciously dropped. If Pm is greater than some threshold value then the node is marked as being malicious and node A will inform other neighbors, remove it from routes, etc..

VI. EXPERIMENTAL SETUP

We have implemented our proposed algorithm in NS2, which has been highly validated by the networking research community. The simulation parameters where listed in table 1.

Table 1: NS2 parameters

Parameters	Value
MAC Layer	IEEE 802.11
Number of nodes	20
Data rate	11Mbps
Packet Size	512 B
Simulation Duration	200 sec
Traffic Flow	TCP

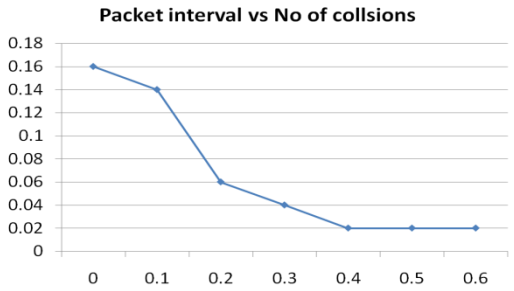

Fig 3: packet interval vs no of collisions

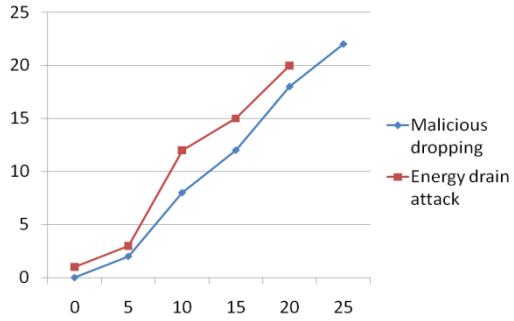

Fig 4: shows the probability of collisions for different traffic loads.

Fig 3 describes the less packet interval time, the more is the load so high probability of collision. Fig 4 shows the "computed" Pm percentage values at each node for the energy drain attack (ED) and malicious dropping (MD) as a function of simulation "specified" malicious packet (or ACK) dropping levels.

VII. CONCLUSION

Detecting malicious dropping packets is a crucial issue in networks to minimize various security attacks such as blackhole, greyhole, and wormhole attacks. All networks drop the packets in the presence of collisions, channel errors and the network traffic exceeds its capacities, which depend on the environment of the network. Hence, in this paper we present a method to determine the cause of packet drops by a node such as collisions, channel errors and heavy traffic conditions. If nodes can have reasonable estimates for collision probabilities and channel error probabilities, even fairly low levels of malicious packet drops can be detected significantly. To evaluate the performance of the proposed method, we simulated it using NS2. Experimental results relived that proposed method performances well.

REFERENCES

[1] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: a secure on-demand routing protocol for ad hoc networks," Wirel. Netw., vol. 11, no. 1-2, pp. 21–38, 2005.

[2] Q. Pang, S. Liew, and V. Leung, "Design of an effective lossdistinguishable mac protocol for 802.11 wlan," Communications Letters, IEEE, vol. 9, no. 9, pp. 781–783, Sep 2005.

[3] J. Kim, S. Kim, S. Choi, and D. Qiao, "Cara: Collision-aware rate adaptation for ieee 802.11 wlans," INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, pp. 1–11, April 2006.

[4] J.-H. Yun and S.-W. Seo, "Novel collision detection scheme and its applications for ieee 802.11 wireless lans," Computer Communications, vol. 30, no. 6, pp. 1350–1366, —2007.

[5] D. Malone, P. Clifford, and D. J. Leith, "Mac layer channel quality measurement in 802.11," Communications Letters, IEEE, vol. 11, no. 2, pp. 143–145, Feb. 2007.

[6] M. Just, E. Kranakis, and T. Wan, "Resisting malicious packet dropping in wireless ad hoc networks," in In Proc. of ADHOCNOW03. Springer Verlag, 2003, pp. 151–163.

[7] F. Anjum and R. Talpade, "Lipad: lightweight packet drop detection for ad hoc networks," Vehicular Technology Conference, 2004. VTC2004- Fall. 2004 IEEE 60th, vol. 2, pp. 1233–1237 Vol. 2, Sept. 2004.

[8] O. F. Gonzalez, M. P. Howarth, and G. Pavlou, "Detection of packet forwarding misbehavior in mobile ad-hoc networks." in WWIC,ser. Lecture Notes in Computer Science, F. Boavida, E. Monteiro,

[9] S. Mascolo, and Y. Koucheryavy, Eds., vol. 4517. Springer, 2007, pp. 302–314. [Online]. Available: http://dblp.uni-trier.de/db/conf/wwic/wwic2007.html#GonzalezHP07.

[10] H. Ma, J. Zhu, and S. Roy, "On loss differentiation for csma-based dense wireless network," Communications Letters, IEEE, vol. 11, no. 11, pp. 877–879, November 2007.

[11] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking. New York, NY, USA: ACM, 2000, pp. 255–265.

[12] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," Selected Areas in Communications, IEEE Journal on, vol. 18, no. 3, pp. 535–547, Mar 2000.

[13] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of reliable transport protocol over ieee 802.11 wireless lan: analysis and enhancement," INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, pp. 599–607 vol.2, 2002.

[14] M. Carvalho and J. Garcia-Luna-Aceves, "Delay analysis of ieee 802.11 in single-hop networks," Network Protocols, 2003. Proceedings. 11th IEEE International Conference on, pp. 146–155, Nov. 2003.

[15] J. Yin, X. Wang, and D. Agrawal, "Optimal packet size in error-prone channel for ieee 802.11 distributed coordination function," Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE, vol. 3, pp. 1654–1659 Vol.3, March 2004.

[16] N. Gupta and P. R. Kumar, "A performance analysis of the 802.11 wireless lan medium access control," Communications in Information and Systems, vol. 3, no. 4, pp. 279–304, 2004.

[17] J. N. Arauz, "802.11 markov channel modeling," Ph.D. dissertation, School of Information Sciences, University of Pittsburgh, 2004.

[18] Thaier Hayajneh, Prashant Krishnamurthy, David Tipper, and Taehoon Kim, "Detecting Malicious Packet Dropping in the Presence of Collisions and Channel Errors in Wireless Ad hoc Networks, IEEE Transaction,2009.