Identifying Flood Attackers in Disruption Tolerant Networks

P.Suresh, P.S.Nandhini

Department Of Information Technology,

Kongu Engineering College,

Perundurai, Erode.

{sureshme@gmail.com, nandhinips.12cce@kongu.edu}

ABSTRACT

Disruption Tolerant Network (DTN) is a networking architecture that is designed to provide communications in the most unstable and stressed environment. DTN network would normally be subjected to frequent disruptions. Due to the restriction in network resources such as contact opportunity and buffer space, DTNs are exposed to flood attacks. Flooding is a denial of service attack that is designed to bring a network or service down by flooding it with large amount of traffic. In this paper, limiting the rate is used to defend against flood attacks in DTNs, so that each node limits the number of packets that it can generate in each time interval and the number of replica that it can generate for each packet. It is a distributed scheme to detect the violation of rate limits. The detection scheme uses claim-carry-and-check mechanism. The structure of the claim adopts the pigeonhole principle to detect the attacker. The detection of the attacker and the efficiency of the proposed scheme are determined using event driven simulations.

Key words: DTN, Security, flood attack, detection

1 INTRODUCTION

Wired and wireless networks have enabled a wide range of devices to be interconnected over vast distance. For example, today it is possible to connect from a cell phone to a number of servers. Though these networks are successful, they still can't be reached everywhere, and for some applications the cost is prohibitive. The reason for this limitation is that current networking technologies relies on the set of fundamental assumptions that are not true in all environments. The first important assumption is that an end-to-end connection exists from the source to the destination, via multiple intermediaries. This assumption can be easily broken due to mobility, power saving or undependable networks.

An intermittently connected mobile network (ICMN) is an attempt to extend the research of networks, which include deep space networks, sensor networks, mobile adhoc networks and low-cost networks. The core idea is that communication can be enabled between these networks if protocols are designed to accommodate disconnection. DTN enable access to information when stable end-to-end paths don't exists and network infrastructure access can't be assumed. To overcome disruptions in connectivity, due to the availability of opportunistic mobility DTN technology utilizes the persistence within the network nodes. Disruption may occur because of the restrictions of wireless radio range, scattered mobile nodes, energy resources, attack and noise. DTN makes use of "store-carry-and-forward", i.e., if a node receives some packets, it stores these packets, carries them around until it contacts another node, and forwards them. Because of mobility, the bandwidth which is available during the opportunistic contacts is a limited resource. The mobile nodes also have limited buffer space.

Due to the limitation in network resources, DTNs are exposed to flood attacks. There are two types of flood attack: packet flood attack- in which attackers injects different packets into the network, replica flood attack- in which attackers inject replica of the same packet into the network. Flooded packets degrade the network service by wasting precious bandwidth and buffer resource.

In DTNs, many works are done on routing, data dissemination, black hole attack and wormhole attack but little work on flood attacks. The packet flood attack can be

cleared with authentication technique but it fails when there prevail insider attackers.

In this paper, rate limiting is used to defend against flood attacks in DTNs. To prevent packet flood attack, each node has a limit over the number of packets that it generates and sends to the network. To prevent replica flood attack, each node has a limit over the number of replica that it generates for each packet. If a node violates its rate limits, the attack will be detected. The basic idea of detection is claim-carry-and-check. Each node counts the number of packets or replica that it has sent out, and claims the packet or replica count to other nodes; the receiving node carries the claim and moves around when two nodes contact they exchange some claims to detect the inconsistency. If an attacker tries to violate its limits, then the claim structure adopts the principle of pigeonhole. Thus, the attacker is detected due to inconsistencies in the claim.

2 OVERVIEW

2.1 PROBLEM DEFINITIONS

DTN consists of mobile nodes which are carried by human beings, vehicles, etc. DTN enables data transfer when mobile nodes are intermittently connected. Due to lack of consistent connectivity, two nodes can only transfer data when they move into the transmission range of each other. Since the contacts between nodes are opportunistic and duration of contact may be short because of mobility, the bandwidth which is only available during the opportunistic contacts is limited resource. Also, mobile nodes have limited buffer space. Due to limitation in bandwidth and buffer space, DTNs are vulnerable to flood attack. In flood attack, attackers inject different packets (packet flood attack) or replicas of same packet (replica flood attack) in to the network. Flooded packets and replicas can waste the precious bandwidth and buffer space. The battery life of mobile is also wasted for transmission of flooded packets. It prevents benign packets from being forwarded and thus degrades the network service. So, it is necessary to secure DTNs against flood attack.

2.2 MODELS AND ASSUMPTIONS

2.2.1 Network model

In DTNs, since the duration of contacts may be small, a data item is usually split into smaller packets (or fragments) to facilitate data transfer. For simplicity, all packets are assumed to have the predefined size. Though in DTNs the allowed delay of packet delivery is usually lengthy, it is still not practical to allow unlimited delays. Thus, each packet will have a lifespan. The packet has no significance after its lifespan expires and will be discarded. Every packet generated by node is different. This is done by adding the source node ID and a unique sequence number, which is allocated by the source for the packet, in the packet header.

2.2.2 Adversary Model

There are many attackers in the network. An attacker can produce packets or replicas. When flooding packets, the attacker will act as a source node. It generates and introduces more packets into the network than its rate limit L. When flooding replicas, the attacker forwards its buffered packets more times than its limit l for them. The attackers may be insiders in the network.

2.3 BASIC IDEA: CLAIM-CARRY-AND-CHECK

2.3.1 Packet Flood Detection

To detect the attackers that violate their rate limit L, count the number of unique packets that each node as a source has generated and sent to the network in the present interval. However, since the node may send its packets to any node it contacts, no other node can monitor all of its sending activities. To address this challenge, the idea is to let the node itself count the number of unique packets that it, as a source, has generated, and claim the up-to-date packet count (together with information such as its ID and a timestamp) in each packet sent out. If an attacker is flooding more packets than its limit of the rate, it has to dishonestly claim a count smaller than the real value in the flooded packet, since the value is larger than its limit of the rate and thus there is a clear indication of attack.

The count which is claimed must have been used prior by the attacker in another claim, which is fulfilled by the pigeonhole principle, and the two claims are inconsistent. The nodes which have received the packets from the attacker carry the claims added in those packets when they move around. When two of the nodes contact, they check if there is any inconsistency between their collected claims. Thus, the attacker is detected when an inconsistency is found.

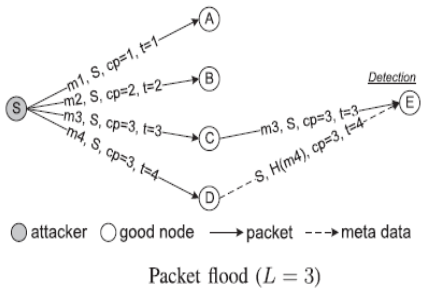


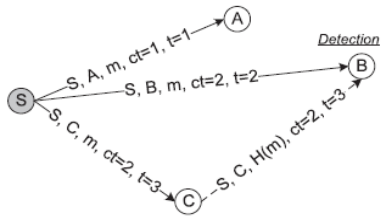
Fig.1. Packet Flood Attack Detection

In Fig.1, S is an attacker that successively generates out four packets to A, B, C, and D. Since L = 3,

if S claims the count 4 in the fourth packet m4, the packet will be discarded by D. Thus, S with no honesty claims the count to be 3, which has already been claimed in the prior packet m3. m3 (including the claim) is then forwarded to node E. When D and E contact, they exchange the count claims included in m3 and m4, and checks that S has used the same count value in different packets. Thus, they detect that S to be an attacker.

2.3.2 REPLICA FLOOD DETECTION

Claim-carry-and-check can also be used to detect the attacker that forwards a buffered packet more times than its limit l. In an exact way, when the source node of a packet or an intermediate node transmits the packet to its next hop, it claims a transmission count to the node it encounters which means the number of times it has transmitted that particular packet (including the current transmission). Depending on if the node is the source or an intermediate node and which routing protocol is used, the next hop node can know the limit of the node l for the packet, and ensure that the count which is claimed is within the correct range. Thus, if an attacker wants to transmit the packet more than l times, it must claim a count which has been used prior. Similarly the attacker will be detected, as in case of packet flood attack.



Replica flood by the source ($l = 2$)

Fig.2. Replica Flood Attack Detection

3 PROPOSED SCHEME

3.1 CLAIM CONSTRUCTION

Two pieces of metadata are added to each packet, P-claim and T-claim are used to detect packet flood and replica flood attacks, respectively. P-claim is added by the source and transmitted to later hop nodes along with the packet. T-claim is produced and processed hop-by-hop. Specifically, the source produces a T-claim and appends it to the packet. When the first hop gets this packet, it removes the T-claim; when it forwards the packet out, it appends a fresh T-claim to the packet. This process continues in future hops. Each hop keeps the P-claim of the source and the T-claim of its prior hop to detect attacks.

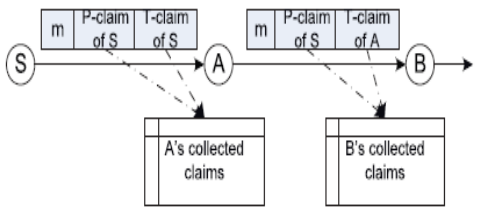


Fig. 3.3 Claim Construction

3.1.1 P-CLAIM

When a source node S sends a new packet m (which has been generated by S and not sent out before) to a communicated node, it generates a P-claim as follows:

$$S, C_P, t$$

Table 1 P-Claim Parameters

PARAMETERS	DESCRIPTION
S	Source Id
C_P	Packet count
T	Current time

The P-claim is attached to packet m as a header field, and will always be forwarded together with the packet to future hops. When the communicated node receives this packet, it checks the value of C_P. If C_P is greater than L, it discards this packet; otherwise, it stores the packet and P claim.

3.1.2 T-CLAIM

When node A transmits a packet m to node B, it appends a T-claim to m. The T-claim includes A's current transmission count C_t for m (i.e., the number of times it has transmitted m out) and the current time t. The T-claim is as follows:

$$A, B, C_t, t$$

Table 2 T-Claim Parameters

PARAMETERS	DESCRIPTION
A	Source Id
B	Receiver Id
C_t	A's Transmission count
T	Current time

3.2 DETECTION OF THE ATTACKER

Suppose two nodes contact and they have a number of packets to forward to each other, the nodes will exchange their collected P-Claims and T-Claims to detect the flood attacks. If all the claims are communicated, the communication cost will be too high. So, inter contact sampling technique is used. A node redirects the received

VARIABLES	DESCRIPTIONS
P_d	Probability that S is detected.
Ps	Probability that event e_A =TRUE (or e_B =TRUE, resp).
e_i	A Boolean event which means if node i (i € {A, B}) has sampled at least one flooded packet(e_i =TRUE) or not (e_i =FALSE).
P_{ovp}	The conditional probability that if $e_A = e_B$ =TRUE then $\hat{e}_{AB}$= TRUE
$\hat{e}_{AB}$	A Boolean event which means if node A and B have sampled at least one pair of inconsistent packets. ($\hat{e}_{AB}$ =TRUE) or not.
Z	Samples of P-claims and T-claims
N	The number of nodes in the network.
M	The number of attackers in the network.
r	The proportion of good nodes, i.e., r $= \frac{N-M}{N}$.
K	The number of nodes that a claim is exchanged to. K<<N
a	The number of flooded packets sent to A and B.
n	The total number of packets sent to A and B.
y	The proportion of flooded packets sent to A and B, i.e., y = $\frac{a}{n}$.

claims to next K (a system parameter) nodes it will contact, and this contacted node will exchange (but not redirect again) these redirected claim in the subsequent contacts.

If a node has packets, to send then for each packet it will generate a P-claim and T-Claim. If a node receives a packet, it verifies the P-Claim and T-Claim against the locally collected Claims. If inconsistency is detected then tag the signer of the Claims as an attacker and disseminate an alarm against the attacker to the network.

TABLE 1 VARIABLE USED IN ANALYSIS

4 PROBABILITY OF DETECTING THE ATTACKERS

The probability of detecting the attacker is illustrated as follows,

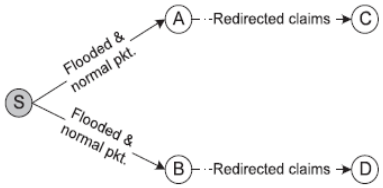


Fig.4. Detection of Attackers

In Fig.4, the attacker S floods two sets of inconsistent packets to two good nodes A and B, respectively. Each flooded packet received by A is inconsistent with one of the flooded packets received by B. In the contacts with A and B, S also forwards some normal, not flooded, packets to make the attack harder to detect.

Let y denotes the proportion of flooded packets among those sent by S. For simplicity, y is assumed to be same in both the contacts. Suppose A and B redirect the claims sampled in the contact with S to C and D, respectively. Then the probability of detecting the attacker is given by,

$$Pd = Ps\ [1 - \left(\frac{K}{N-1}\right).\left(1 - r\frac{K}{N-2}\right).(1 - r\frac{K}{N-2}P_s\ P_{ovp}).(1 - r2\frac{K}{N-2}P_s\ P_{ovp})]$$

The expected number of flooded packets that A or B can sample is yZ. Since Z is small while a is not that small (which is assumed to be realistic), Povp is negligible. Considering that K << N, Pd is approximated as follows:

$$Pd \cong Ps\ [1 - \left(\frac{K}{N-1}\right).\left(1 - r\frac{K}{N-2}\right)] \cong Ps\frac{1+r}{N}$$

5 PERFORMANCE EVALUATIONS

5.1 EXPERIMENTAL SETUP

Number of Nodes	40
Simulation Time	200 sec
Packet Size	10000 Bytes
Number of Attackers	20
Propagation Model	Two Ray Ground Model
Mobility Model	Random waypoint
Antenna Type	Omni Directional
Node Deployment	Random

5.2 ANALYSIS VERIFICATION

The performance of the proposed scheme is analyzed by taking two parameters into account. From the obtained result it is inferred that the performance of the detection scheme increases with increasing the number of intermediate nodes. The following are the parameters which are taken into account for the evaluation of performance.

1. Packet Delivery Ratio

2. Detection Rate.

5.2.1 COMPARISON OF PACKET DELIVERY RATIO

Nodes Vs Packet delivery ratio

Packet delivery ratio is defined as the ratio of the number of delivered data packets to the destination. It is compared before and after deploying the flood attackers and how it is improved after applying the detection scheme. The detection scheme generates an alarm message

after identifying the attacker. So, the receiver drops all the duplicate packets from the attackers. Thus, the packet delivery ratio is increased. The packet delivery ratio variation is shown in Figure 5.1

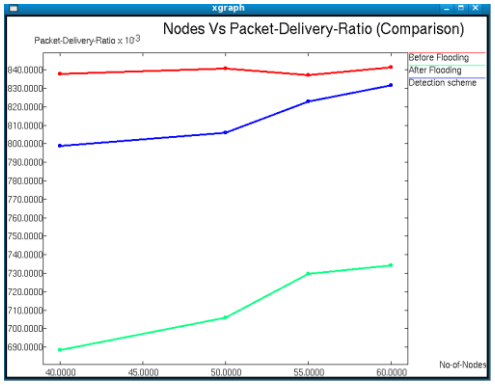


Figure 5.1 Nodes Vs Packet delivery ratios

Nodes Vs Detection rate

Detection Rate is defined as the number of attackers detected to the total number of attackers in the network. The detection rate increases by increasing the number of neighbor nodes. The neighbor nodes that come in contact with each other exchange the claims frequently and identify the attackers. The detection rate is shown in Figure 5.2

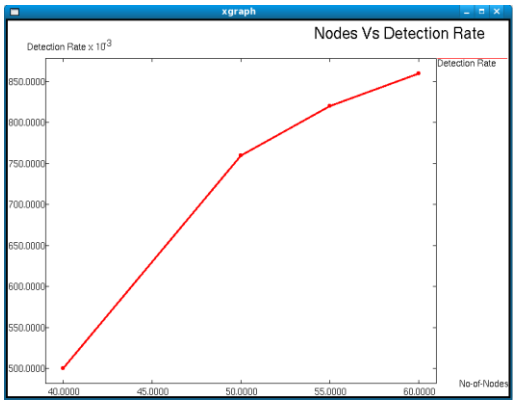


Figure 5.2 Nodes Vs Detection rate

6 CONCLUSIONS

In this paper, rate limiting is used to mitigate flood attacks in DTNs, and proposed a scheme which exploits claim-carry-and-check to probabilistically identify the violation of rate limit in DTN environments. Trace-driven simulations showed that the scheme is effective to detect flood attacks. The scheme works in a distributed manner, not relying on any infrastructure or online central authority, which well suits the environment of DTNs.

7 REFERENCES

1. Amin Vahdat and David Becker, 'Epidemic Routing for Partially Connected AdHoc Networks', Technical Report on CS-200006, Duke University, 2000.
2. Bryan Parno, Adrian Perrig, and Virgil Gligor, 'Distributed Detection of Node Replication Attacks in Sensor Networks' IEEE Symposium on Security and Privacy, 2005.
3. Feng Li, Avinash Srinivasan, and Jie Wu, 'Thwarting Blackhole Attacks in Distruption-Tolerant Networks Using Encounter Tickets,' Proceedings of IEEE INFOCOM, 2009.
4. Jelena Mirkovic, Seven Dietrich, David Dittrich and Peter Reiher, Internet Denial of Service: Attack and Defense Mechanisms. Prentice Hall, 2005.
5. Qinghua Li and Guohong Cao, 'Mitigating Routing Misbehavior in Disruption Tolerant Networks,' IEEE Transaction on Information Forensics and Security,vol. 7, No. 2, pp. 664-675, 2012.
6. Qinghua Li and Guohong Cao, 'Efficient and Privacy-Preserving Data Aggregation in Mobile Sensing,' IEEE International Conference on Network Protocols, 2012.
7. Qinghua Li, Wei Gao, Sencun Zhu and Guohong Cao, 'A Routing Protocol for Socially Selfish Delay Tolerant Networks,' Ad Hoc Networks, vol. 10, No. 8, 2011.
8. Qinghua Li, Sencun Zhu and Guohong Cao, 'Routing in Socially Selfish Delay Tolerant Networks,' Proceedings of IEEE INFOCOM, 2010.
9. Vivek Natarajan, Yi Yang and Sencun Zhu , 'Resource-Misuse Attack Detection in Delay-Tolerant N etworks', International Performance of Computing and Communication Conference (IPCCC), 2011.
10. Yanzhi Ren, Mooi Choo Chuah, Jie Yang and Yingying Chen, 'Detecting Wormhole Attacks in Delay Tolerant Networks', IEEE Magazine on Wireless Communication, vol. 17, no. 5, pp. 36-42, 2010.