DETECTION OF MISREPORTING NODES IN DISRUPTION TOLERANT NETWORK

Anu M
M.E,Computer Science Engineering ,Fourth Semester,Anna University
anuk9140@gmail.com
Maharaja Engineering College
Avinashi, Coimbatore(D.T)
mec_principal@yahoo.co.in

Abstract- Disruption Tolerant Networks (DTNs) exploit the intermittent connectivity between nodes to transfer data. It follows a store-carry-forward mechanism to transfer data. A node misbehave by dropping packets and acts selfish as they are unwilling to spend resources such as power and buffer on forwarding packets of other nodes. In such nodes routing misbehavior reduces the packet delivery ratio and wastes system resources such as power and bandwidth. Methods to mitigate routing misbehavior in mobile networks cannot be applied to DTN because of its intermittent connectivity. Existing systems are designed to identify selfish node or malicious node on DTNs.When it finds misbehaving or packet dropping node then it sends information to server. Server will then stop the data transfer and choose alternate route for communication.It detects misbehaving node and is selected dynamically to avoid it being compromised. When a misbehaving node misreports, it is converted to blacklist node and avoid this node from the network.

1. INTRODUCTION

Disruption Tolerant Networks (DTNs) consist of mobile nodes carried by human beings vehicles etc. DTNs enable data transfer when mobile nodes are only intermittently connected, making them appropriate for applications where no communication infrastructure is available such as military scenarios and rural areas. Due to lack of consistent connectivity, two nodes can only exchange data when they move into the transmission range of each other (which is called a contact between them). DTNs employ such contact opportunity for data forwarding with store carry-and-forward.When a node receives some packets, it stores these packets in its buffer carries them around until it contacts another node, and then forwards them. Since the contacts between nodes are opportunistic and the duration of a contact may be short because of mobility the usable bandwidth which is only available during the opportunistic contacts is a limited resource.

Also, mobile nodes may have limited buffer space. Due to the limitation in bandwidth and buffer space, DTNs are vulnerable to flood attacks. In flood attacks, maliciously or selfishly motivated attackers inject as many packets as possible into the network, or instead of injecting different packets the attackers forward replicas of the same packet to as many nodes as possible. For convenience, call the two types of attack packet flood attack and replica flood attack respectively. Flooded packets and replicas can waste the precious bandwidth and buffer resources prevent benign packets from being forwarded and thus degrade the network service provided to good nodes. Moreover, mobile nodes spend much energy on transmitting/receiving flooded packets and replicas which may shorten their battery life.

Therefore, it is urgent to secure DTNs against flood attacks. Although many schemes have been proposed to defend against flood attacks on the Internet and in wireless sensor networks they assume persistent connectivity and cannot be directly applied to DTNs that have intermittent connectivity. In DTNs, little work has been done on flood attacks, despite the many works on routing data dissemination black hole attack wormhole attack and selfish dropping behavior. Here noted that the packets flooded by outsider attackers packets and replicas with valid signatures. Thus, it is still an open problem is to address flood attacks in DTNs.

2 .MOTIVATION

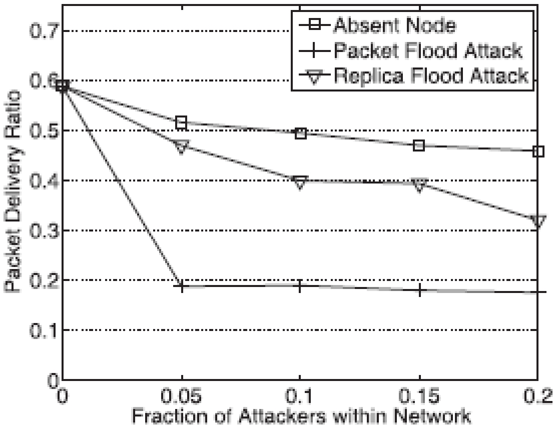
2.1 The Potential Prevalence of Flood Attack

Many nodes may launch flood attacks for malicious orselfish purposes. Malicious nodes, which can be the nodesdeliberately deployed by the adversary or subverted by theadversary via mobile phone worms [16], launch attacks tocongest the network and waste the resources of other nodes.

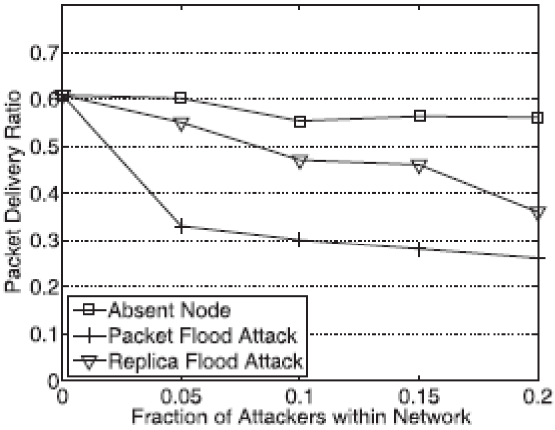
Selfish nodes may also exploit flood attacks to increasetheir communication throughput. In DTNs, a single packetusually can only be delivered to the destination with aprobability smaller than 1 due to the opportunistic connectivityIf a selfish node floods many replicas of its ownpacket, it can increase the likelihood of its packet beindelivered, since the delivery of any replica means successfuldelivery of the packet. With packet flood attacks, selfishnodes can also increase their throughput, albeit in a subtlermanner.

2.2 The Effect of Flood Attacks

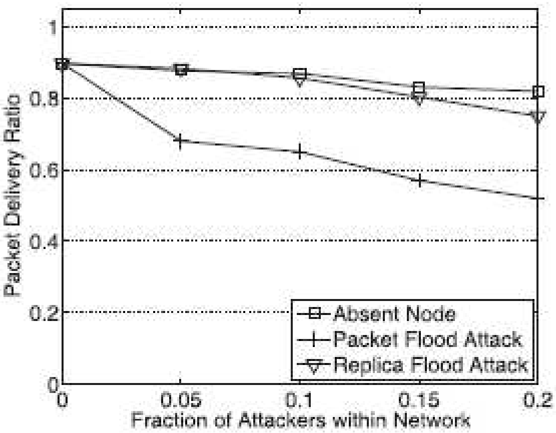
To study the effect of flood attacks on DTN routing andmotivate our work, we run simulations on the MIT Realitytrace [17] (see more details about this trace in Section 7).We consider three general routing strategies in DTNs.1) Single-copy routing (e.g., [18], [8]): after forwarding a packetout, a node deletes its own copy of the packet. Thus, eachpacket only has one copy in the network. 2) Multicopy routing(e.g., [19]): the source node of a packet sprays a certainnumber of copies of the packet to other nodes and each copyis individually routed using the single-copy strategy. Themaximum number of copies that each packet can have isfixed. 3) Propagation routing (e.g., [17], [20], [21]): when anode finds it appropriate (according to the routing algorithm)to forward a packet to another encountered node, itreplicates that packet to the encountered node and keeps itsown copy. There is no preset limit over the number of copiesa packet can have. In our simulations, SimBet [8], Spray-and-Focus [19] (three copies allowed for each packet) andPropagation are used as representatives of the three routingstrategies, respectively. In Propagation, a node replicates apacket to another encountered node if the latter has morefrequent contacts with the destination of the packet.



(a) Single-copy Routing



(b) Multi-copy Routing



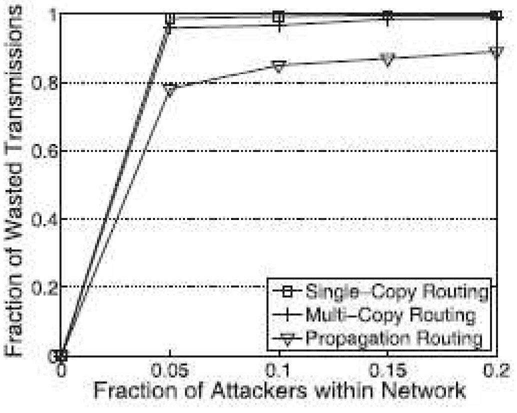
(c) Propagation Routing

Two metrics are used, The first metric is packet deliveryratio, which is defined as the fraction of packets delivered totheir destinations out of all the unique packets generated.The second metric is the fraction of wasted transmissions(i.e., the transmissions made by good nodes for floodedpackets). The higher fraction of wasted transmissions, themore network resources are wasted. We noticed that theeffect of packet flood attacks on packet delivery ratio hasbeen studied by Burgess et al. [22] using a different trace [4].Their simulations show that packet flood attacks significantlyreduce the packet delivery ratio of single-copyrouting but do not affect propagation routing much.However, they do not study replica flood attacks and theeffect of packet flood attacks on wasted transmissions.In our simulations, a packet flood attacker floods packetsdestined to random good nodes in each contact until thecontact ends or the contacted node's buffer is full. A replicaflood attacker replicates the packets it has generated toevery encountered node that does not have a copy. Eachgood node generates thirty packets on the 121st day of theReality trace, and each attacker does the same in replicaflood attacks. Each packet expires in 60 days. The buffer sizeof each node is 5 MB, bandwidth is 2 Mbps and packetsize is 10 KB.

3. PROBLEM DEFINITION

3.1 Defense against Packet Flood Attacks

We consider a scenario where each node has a rate limit Lon the number of unique packets that it as a source cangenerate and send into the network within each timeinterval T. The time intervals start from time 0, T, 2T, etc.The packets generated within the rate limit are deemedlegitimate, but the packets generated beyond the limit aredeemed flooded by this node. To defend against packetflood attacks, our goal is to detect if a node as a source hasgenerated and sent more unique packets into the networkthan its rate limit L per time interval.A node's rate limit L does not depend on any specific routing protocol, but it can be determined by a servicecontract between the node and the network operator asdiscussed in Section 3.1.3. Different nodes can have different rate limits and their rate limits can be dynamically adjustedThe length of time interval should be set appropriately. Ifthe interval is too long, rate limiting may not be veryeffective against packet flood attacks. If the interval is tooshort, the number of contacts that each node has during oneinterval may be too nondeterministic and thus it is difficultto set an appropriate rate limit. Generally speaking, theinterval should be short under the condition that mostnodes can have a significant number of contacts with othernodes within one interval, but the appropriate lengthdepends on the contact patterns between nodes in thespecific deployment scenario.



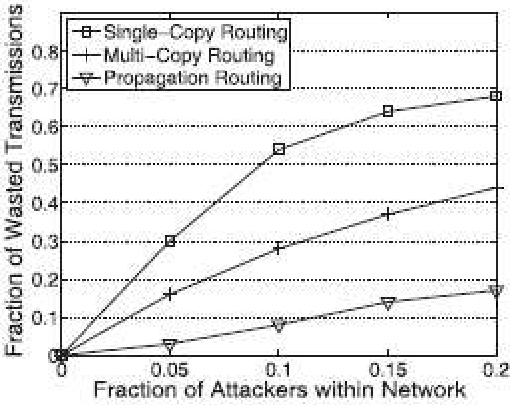
(a) Packet Flood Attack

3.2 Defense against Replica Flood Attacks

As motivated in Section 2, the defense against replica floodconsiders single-copy and multicopy routing protocols.These protocols require that, for each packet that a nodebuffers no matter if this packet has been generated by thenode or forwarded to it, there is a limit l on the number oftimes that the node can forward this packet to other nodes.The values of l may be different for different bufferedpackets. Our goal is to detect if a node has violated therouting protocol and forwarded a packet more times thanits limit l for the packet.

A node's limit l for a buffered packet is determined bythe routing protocol. In multicopy routing, l ¼ L0 (where L0is a parameter of routing) if the node is the source of thepacket, and l ¼ 1 if the node is an intermediate hop (i.e., itreceived the packet from another node). In single-copyrouting, l ¼ 1 no matter if the node is the source or anintermediate hop. Note that the two limits L and l do notdepend on each other.

We discuss how to defend against replica flood attacksfor quota-based routing [23], [19], [24] in Section 4.9.3.1.3 Setting the Rate Limit LOne possible method is to set L in a request-approve style.When a user joins the network, she requests for a rate limitfrom a trusted authority which acts as the network operator.

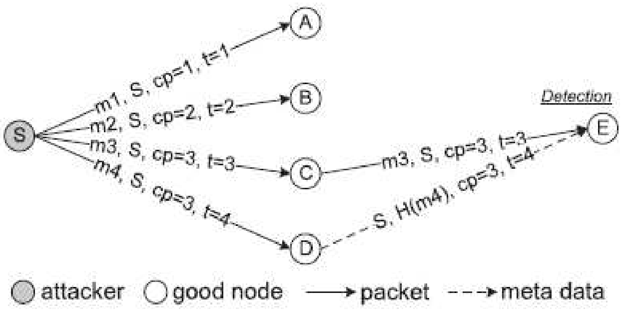


(b) Replica Flood Attack

4. BASIC IDEA: CLAIM-CARRY-AND-CHECK

4.1 Packet Flood Detection

To detect the attackers that violate their rate limit L, wemust count the number of unique packets that each node asa source has generated and sent to the network in thecurrent interval. However, since the node may send itspackets to any node it contacts at any time and place, noother node can monitor all of its sending activities. Toaddress this challenge, our idea is to let the node itself countthe number of unique packets that it, as a source, has sentout, and claim the up-to-date packet count (together with alittle auxiliary information such as its ID and a timestamp)in each packet sent out. The node's rate limit certificate isalso attached to the packet, such that other nodes receivingthe packet can learn its authorized rate limit L. If an attackeris flooding more packets than its rate limit, it has todishonestly claim a count smaller than the real value in theflooded packet, since the real value is larger than its ratelimit and thus a clear indicator of attack. The claimed countmust have been used before by the attacker in anotherclaim, which is guaranteed by the pigeonhole principle, andthese two claims are inconsistent. The nodes which havereceived packets from the attacker carry the claims includedin those packets when they move around. When two ofthem contact, they check if there is any inconsistencybetween their collected claims. The attacker is detectedwhen an inconsistency is found.Let us look at an example in Fig.



(a) Packet flood ($L = 3$)

a. S is an attacker thatsuccessively sends out four packets to A, B, C, and D,respectively. Since L ¼ 3, if S claims the true count 4 in thefourth packet m4, this packet will be discarded by D. Thus,S dishonestly claims the count to be 3, which has alreadybeen claimed in the third packet m3. m3 (including the claim) is further forwarded to node E.

5. ALGORITHM

The protocol run by each node in a contact

```
Metadata (P-claim and T-claim) exchange and attackdetection
if Have packets to send then
    For each new packet, generate a P-claim;
    For all packets, generate their T-claims and signthem with a hash tree;
    Send every packet with the P-claim and T-claimattached;
end if
if Receive a packet then
    if Signature verification fails or the count value in itsP-claim or T-claim is invalid then
        Discard this packet;
    end if
    Check the P-claim against those locally collected andgenerated in the same time interval to detectinconsistency;
    Check the T-claim against those locally collected forinconsistency;
    if Inconsistency is detected then
        Tag the signer of the P-claim (T-claim, respectively)as an attacker and add it into a blacklist;
        Disseminate an alarm against the attacker to thenetwork;
    else
        Store the new P-claim (T-claim, respectively);
    end if
end if
```

6. METADATA EXCHANGE

When two nodes contact they exchange their collectedP-claims and T-claims to detect flood attacks. If all claimsare exchanged, the communication cost will be too high.Thus, our scheme uses sampling techniques to keep thecommunication cost low. To increase the probability ofattack detection, one node also stores a small portion ofclaims exchanged from its contacted node, and exchangesthem to its own future contacts. This is called redirection.

6.1 Sampling

Since P-claims and T-claims are sampled together (i.e., when a P-claim is sampled the T-claim of the same packet is alsosampled), in the following we only consider P-claims.A node may receive a number of packets (each with aP-claim) in a contact. It randomly samples Z (a systemparameter) of the received P-claims, and exchanges thesampled P-claims to the next K (a system parameter)different nodes it will contact, excluding the sources of theP-claims and the previous hop from which these P-claimsare received.However, a vulnerability to tailgating attack should beaddressed. In tailgating attack, one or more attackerstailgate a good node to create a large number (say, d) offrequent contacts with this node, and send Z packets (notnecessarily generated by the attackers) to this node in eachcreated contact. If this good node sends the Zd P-claims ofthese contacts to the next K good nodes it contacts, mucheffective bandwidth between these good nodes will bewasted, especially in a large network where K is not small.To address this attack, the node uses an inter-contactsampling technique to determine which P-claims sampledin historical contacts should be exchanged in the currentcontact. Let SK denote a set of contacts. This set includes theminimum number of most recent contacts between thisnode and at least K other different nodes. Within this set, allthe contacts with the same node are taken as one singlecontact and a total of Z P-claims are sampled out of thesecontacts. This technique is not vulnerable to the tailgatingattack since the number of claims exchanged in each contactis bounded by a constant.

6.2 Redirection

There is a stealthy attack to flood attack detection. Forreplica flood attacks, the condition of detection is that atleast two nodes carrying inconsistent T-claims can contact.However, suppose the attacker knows that two nodes A andB never contact. Then, it can send some packets to A, andinvalidly replicate these packets to B. In this scenario, thisattacker cannot be detected since A and B never contact.Similarly, the stealthy attack is also harmful for somerouting protocols like Spray-and-Wait [19] in which eachpacket is forwarded from the source to a relay and thendirectly delivered from the relay to the destination.To address the stealthy attack, our idea is to add onelevel of indirection. A node redirects the Z P-claims andT-claims sampled in the current contact to one of the next K nodes it will contact, and this contacted node willexchange (but not redirect again) these redirected claims inits own subsequent contacts. Look at the example in Fig. 6.Suppose attacker S sends mutually inconsistent packets totwo nodes A and B which will never contact. Suppose Aand B redirect their sampled P-claims to node C and D,respectively. Then so long as C and B or D and A or C andD can contact, the attack has a chance to be detected. Thus,the successful chance of stealthy attack is significantlyreduced

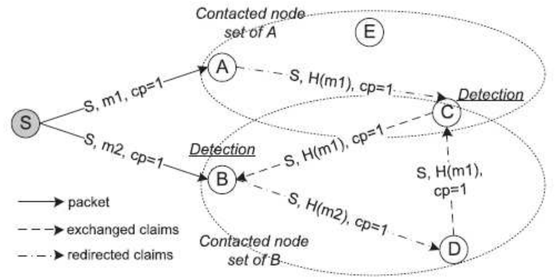


Fig. 6. The idea of redirection which is used to mitigate the stealthy attack.

6.3 The Exchange Process

Each node maintains two separate sets of P-claims (T-claims,respectively in the following) for metadata exchange, asampled set which includes the P-claims sampled from themost recent contacts with K different nodes (i.e., SK in section 5.1), and a redirected set which includes the P-claimsredirected from those contacts. Both sets include Z P-claimsobtained in each of those contacts.When two nodes A and B contact, they first select KZ Pclaimsfrom each set with the inter-contact samplingtechnique and then send these P-claims

to each other. When A receives a P-claim, it checks if thisP-claim is inconsistent with any of its collected P-claims. If the receivedP-claim is inconsistent with a locally collected one and thesignature of the received P-claim is valid, A detects that theissuer (or signer) of the received P-claim is an attacker.Out of all the P-claims received from B, A randomlyselects Z of the P-claims from the sampled set of B, andstores them to A's redirected set. All other P-claims receivedfrom B are discarded after inconsistency check.

Metadata Deletion

A node stores the P-claims and T-claims collected fromreceived data packets for a certain time denoted by _ anddeletes them afterward. It deletes the claims redirectedfrom other nodes immediately after it has exchanged themto K different nodes.

7. CONCLUSION

The rate limiting to mitigate flood attacks in DTNs, and proposed a scheme which exploits claim-carry-and-check to probabilistically detect the violation of rate limit in DTN environments. Our scheme uses efficient constructions to keep the computation, communication and storage cost low. Also, analyzed the lower bound and upper bound of detection probability.Extensivetrace-driven simulations showed that our scheme is effective to detect flood attacks and it achieves such effectiveness in an efficient way. Our scheme works in a distributed manner, not relying on any online central authority or infrastructure,which well fits the environment of DTNs. Besides, it can tolerate a small number of attackers to collude.

REFERENCES

[1] K. Fall, "A Delay-Tolerant Network Architecture foChallengedInternets," Proc. ACM SIGCOMM, pp. 27-34, 2003.

[2] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, andC. Diot,"Pocket Switched Network and Human Mobility in ConferenceEnvironments," Proc. ACM SIGCOMM, 2005.

[3] M. Motani, V. Srinivasan, and P. Nuggehalli, "PeopleNet:Engineering a Wireless Virtual Social Network," Proc. obiCom,pp. 243-257, 2005.

[4] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Maxprop:Routing for Vehicle-Based Disruption-TolerantNetworks," Proc.IEEE INFOCOM, 2006.

[5] S.J.T.U.Grid Computing Center, "Shanghai Taxi Trace data, http://wirelesslab.sjtu.edu.cn/, 2012.

[6] J. Mirkovic, S. Dietrich, D. Dittrich, and P. Reiher, Internet Denial ofService: Attack and Defense Mechanisms. Prentice Hall, 2005.

[7] C. Karlof and D. Wagner, "Secure Routing in Wireless SensoNetworks: Attacks and Countermeasures," Proc. IEEE First Int'lWorkshop Sensor Network Protocols and Applications, 2003.

[8] E. Daly and M. Haahr, "Social Network Analysis for Routing inDisconnected Delay-Tolerant MANETs," Proc. MobiHoc, pp. 32-40,2007.

[9] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in DelayTolerant Networks: A Social Network Perspective," Proc. ACMMobiHoc, 2009.

[10] F. Li, A. Srinivasan, and J. Wu, "Thwarting Blackhole Attacks inDistruption-Tolerant Networks Using Encounter Tickets," Proc.IEEE INFOCOM, 2009.

[11] Y. Ren, M.C. Chuah, J. Yang, and Y. Chen, "Detecting WormholeAttacks in Delay Tolerant Networks," IEEE Wireless Comm.Magazine, vol. 17, no. 5, pp. 36-42, Oct. 2010.

[12] U. Shevade, H. Song, L. Qiu, and Y. Zhang, "Incentive-AwareRouting in DTNS," Proc. IEEE Int'l Conf. Network Protocols(ICNP '08), 2008.

[13] Q. Li and G. Cao, "Mitigating Routing Misbehavior in DisruptionTolerant Networks," IEEE Trans. Information Forensics and Security,vol. 7, no. 2, pp. 664-675, Apr. 2012.

The Storage (KB) Used for Claims and Data Packets

[14] H. Zhu, X. Lin, R. Lu, X.S. Shen, D. Xing, and Z. Cao, "AnOpportunistic Batch Bundle Authentication Scheme for EnergyConstrained DTNS," Proc. IEEE INFOCOM, 2010.

[15] B. Raghavan, K. Vishwanath, S. Ramabhadran, K. Yocum, and A.Snoeren, "Cloud Control with Distributed Rate Limiting," Proc.ACM SIGCOMM, 2007.