Singaporean Journal of Scientific Research

International Journal of Security (IJS)
Singaporean Journal of Scientific Research(SJSR)
Vol.6. No.3 2014 Pp.172-178
Available at:www.iaaet.org/sjsr
Paper Received:08-02-2014
Paper Accepted:14-03-2014
Paper Reviewed by:1Prof. Dr.Ponnambalam2. Chai Cheng Yue
Editor: Dr. Binod Kumar

DENY FLOOD ATTACK DETECTION IN WIRELESS NETWORK

T.AngelRani[1] R.Shanthi[2]
Dept. Of CSE,
PRIST University.,
Thanjavur-613 403.
Email: ***angelrani72@gmail.com***shanthirajen26@gmail.com

ABSTRACT

Disruption Tolerant Networks (DTNs) utilize the mobility of nodes and the opportunistic contacts among nodes for data communications. Due to the limitation in network resources such as contact opportunity and buffer space, DTNs are vulnerable to flood attacks in which attackers send as many packets or packet replicas as possible to the network, in order to deplete or overuse the limited network resources. In this paper, we employ rate limiting to defend against flood attacks in DTNs, such that each node has a limit over the number of packets that it can generate in each time interval and a limit over the number of replicas that it can generate for each packet. We propose a distributed scheme to detect if a node has violated its rate limits. To address the challenge that it is difficult to count all the packets or replicas sent by a node due to lack of communication infrastructure, our detection adopts claim-carry-and check: each node itself counts the number of packets or replicas that it has sent and claims the count to other nodes; the receiving nodes carry the claims when they move, and cross-check if their carried claims are inconsistent when they contact. The claim structure uses the pigeonhole principle to guarantee that an attacker will make inconsistent claims which may lead to detection. To provide rigorous analysis on the probability of detection, and evaluate the effectiveness and efficiency of our scheme with extensive trace driven simulations.

***Keywords*—** DTN, security, flood attack, detection

1. INTRODUCTION

DISRUPTION Tolerant Networks (DTNs) consist of mobile nodes carried by human beings vehicles. 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”; i.e., 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 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 The packets flooded by outsider attackers can be easily filtered with authentication techniques However, authentication alone does not work when insider attackers flood packets and replicas with valid signatures. Thus, it is still an open problem is to address flood attacks in DTNs. Employs rate limiting to defend against flood attacks in DTNs. In our approach, each node has a limit over the number of packets that it, as a source node, can send to the network in each time interval. Each node also has a limit over the number of replicas that it can generate for each packet. The two limits are used to mitigate packet flood and replica flood attacks, respectively. If a node violates its rate limits, it will be detected and its data traffic will be filtered. In this way, the amount of flooded traffic can be controlled.

The technique to detect if a node has violated its rate limits. Although it is easy to detect the violation of rate limit on the Internet and in telecommunication networks where the egress router and base station can account each user's traffic, it is challenging in DTNs due to lack of communication infrastructure and consistent connectivity. Since a node moves around and may send data to any contacted node, it is very difficult to count the number of packets or replicas sent out by this node. Our basic idea of detection is claim-carry-and-check. Each node itself counts the number of packets or replicas that it has sent out, and claims the count to other nodes; the receiving nodes carry the claims around when they move, exchange some claims when they contact, and cross-check if these claims are inconsistent.

An attacker floods more packets or replicas than its limit, it has to use the same count in more than one claim according to the pigeonhole principle, and this inconsistency may lead to detection. Based on this idea, Different cryptographic constructions to detect packet flood and replica flood attacks. Because the contacts in DTNs are opportunistic in nature, our approach provides probabilistic detection. The detection probability can be flexibly adjusted by system parameters that control the amount of claims exchanged in a contact. A lower and upper bound of detection probability and investigate the problem of parameter selection to maximize detection probability under a certain amount of exchanged claims.

2. RELATED WORK

Disruption Tolerant Network (DTN) 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. Lack of consistent connectivity, two nodes can only exchange data when they move into the transmission range of each other. Disruption Tolerant Network employs such contact opportunity for data forwarding with "Store-Carry-and-Forward" method. First receive same files and it store in buffer and then carries another node and forward. Opportunistic contact is a limited resource and mobile nodes may have limited buffer space and limitation bandwidth. In DTN work has been done on flood attacks despite many works on routing data dissemination, blackhole attack, and wormhole attack selfish dropping behavior. Some disadvantages are large amount of traffic, a packet initiating incomplete connection requests, it can no longer process genuine connection requests, communication could not properly.

Our goal is to detect if a node has violated the routing protocol and forwarded a packet more times than its limit. In the Disruption Tolerant Network is vulnerable to flood attack. This attack causes more traffic in the network while receiving the data by the receiver node. There is no verification process done on the receiver side. So, a packet initiating incomplete connection requests and also communication could not be properly large amount of traffic occurs in packet or replicas flood attacks.

To defense against packet flood attacks, our goal is to detect if a node as a source has generated and sent more unique packets into the network that 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 service contract between the node and the network. To defense against replica flood attacks, our goal is to detect if a node has violated the routing protocol and forwarded a packet more times than its limit l for a packet. A node's limit l for a buffered packet is determined by the routing.

If the trusted authority approves this request, it issues a rate limit certificate to this user, which can be used by the user to prove to other nodes of the rate limit. To prevent users from requesting unreasonably large rate limits, a user pays an appropriate amount of money or virtual currency for her rate limit. The flexibility of rate limits leaves legitimate user's usage of the network.

Many nodes may launch flood attacks for malicious or selfish purposes.Selfish nodes may also exploit flood attacks to increase their communication throughput. The more traffic an attacker floods, the more likely it will be detected. We provide a lower and upper bound of detection probability.

To detect the attackers that violate their rate limit L, we must count the number of unique packets that each node as a source has generated and sent to the network in the current interval. Claim-carry-and-check can also be used to detect the attacker that forwards a buffered packet more times than its limit

l. It claims a transmission count which means the number of times it has transmitted this packet.

When a source node S sends a new packet m to a contacted node, it generates a P-claim. Checks the value. If c_p is larger than L, it discards this packet; otherwise, it stores this packet and the P-claim. When node A transmits a packet m to node B, it appends a T-claim to m. In single-copy and multicopy routing, after forwarding m for enough times, A deletes its own copy of m and will not forward m again. In a dishonest P-claim, an attacker uses a smaller packet count than the real value. This causes an inconsistency called count reuse, which means the use of the same count in two different P-claims generated by the same node.

3. PROPOSED DESIGN

To employ rate limiting to defend against flood attacks in Disruption Tolerant Network each node has a limit over the number of packets that it, as a source node, can send to the network in each time interval. Each node also has a limit over the number of replicas that it can generate for each file. The two limits are used to mitigate packet flood and replica flood attacks, respectively. It will be detected and its data traffic will be filtered. In this way, the amount of flooded traffic can be controlled. Although it is easy to detect the violation of rate limit on the Internet and in telecommunication networks. Since a node moves around and may send data to any contacted node, it is very difficult to count the number of files sent out by this node. It is mainly used in claim-carry-and-check method. Each node itself counts the number of packets or replicas that it has sent out, and claims the count to other nodes, the receiving nodes carry the claims around when they move, exchange some claims when they contact, and cross-check if these claims are inconsistent. In this approach to detect packet flood and replica flood attacks. In this approach provides probabilistic detection. The detection probability can be flexibly adjusted by system parameters that control the amount of claims exchanged in a contact. The effectiveness and efficiency of are evaluated with extensive trace-driven simulations. Some advantages are, the main advantage is a technique to detect if a node has violated its rate limits, the network resources that a node can use with the node's contributions to the network in terms of forwarding.

Based on the Principle generation of Dynamic Node with its IPAddress, Port Address, and also the Random Number for each node uniquely and each node information will be maintained by the database. The Claim Network is mainly used to fetch the content from the Source in other words used to claims the data from source node. This network also maintains the user information such as specific filename, file size, file type, sending time, forward node, and count for an individual user. The claim Network is mainly used to claims the signatures. It creates the source node dynamically for avoiding flood attacks.

If an attacker is flooding more packets than its rate limit, it has to dishonestly claim a count smaller than the real value in the flooded packet, since the real value is larger than its rate limit and thus a clear indicator of attack. The flood attack is takes place on receiver node and the attack on file packet. In order to avoid those attack verification process takes place before accessing the file content. If the user is an authorized person then they can access the files, otherwise deny accessing the file. The P-claim and T-claim is verified and the each node has P-claim information from the source node.

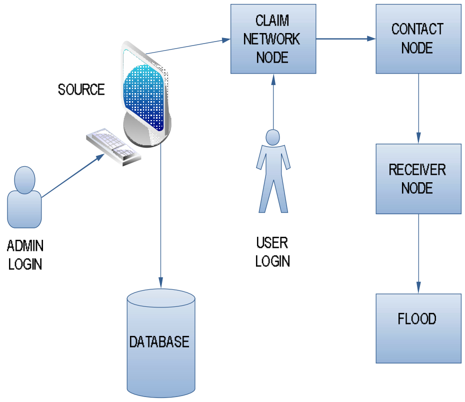


Figure.1.System Architectural Design

3.1. Pigeonhole Principle

Based on the Principle generation of Dynamic Node with its IPAddress, Port Address, and also the Random Number for each node uniquely and each node information will be maintained by the database. Multiple Nodes can be generated at a time and the communication could also be established between them.

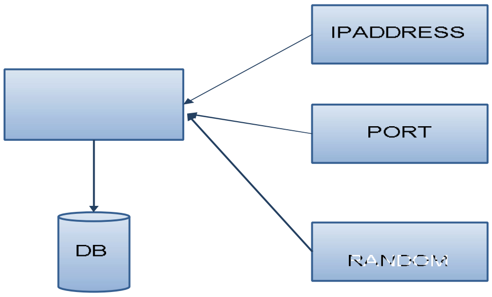


Figure.2. Dynamic Node Creation

The IP address and port address for networking to other node, and the random number comes from maximum number generation and it is confidential purpose. It creates the source node dynamically for avoiding flood attacks. The claim structure uses this principle to guarantee that an attacker will make inconsistent claims which may lead to detection.

3.2. Claim Network

The Claim Network is mainly used to fetch the content from the Source in other words used to claims the data from source node. This network also maintains the user information such as specific filename, file size, file type, sending time, forward node, and count for an individual user. The claim Network is mainly used to claims the signatures.It creates the source node dynamically for avoiding flood attacks.

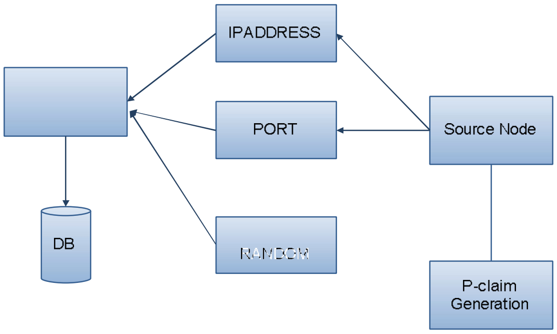


Figure.3. Claiming the Signatures

In P-Claim generation it claims all the details in encrypted form for avoiding flood attacks. It contains all signatures in encrypted form, it attaches into the source node. The source node contains IP address and port address for sending packet to other node. The dynamic node also maintains all information and it can be stored in database.

3.3 Claim Carry Check

If an attacker is flooding more packets than its rate limit, it has to dishonestly claim a count smaller than the real value in the flooded packet, since the real value is larger than its rate limit and thus a clear indicator of attack. The claimed count must have been used before by the attacker in another claim, which is guaranteed by the pigeonhole principle, and these two claims are inconsistent. The nodes which have received packets from the attacker carry the claims included in those packets when they move around. When two of them contact, they check if there is any inconsistency between their collected claims. The attacker is detected when an inconsistency is found.The nodes which have received packets from the attacker carry the claims included in those packets when they move around. When two of them contact, they check if there is any inconsistency between their collected claims. The attacker is detected when inconsistency is found.Claim Carry Check is, each node itself count the number of packets or replicas that it has sent and claims the count to other nodes, the receiving nodes carry the claims when they move, and cross check if their carried claims are inconsistent when they contact.

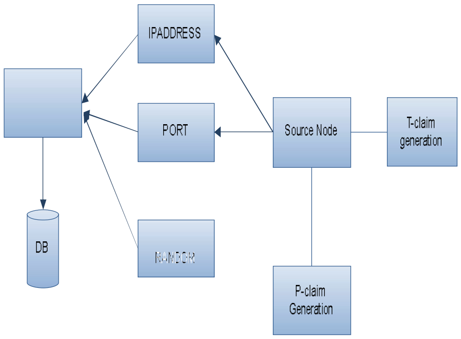


Figure.4. Counting the Packets

3.4. Deny Flood Attacks

The flood attack is takes place on receiver node and the attack on file packet. In order to avoid those attack verification process takes place before accessing the file content. If the user is an authorized person then they can access the files, otherwise deny accessing the file.

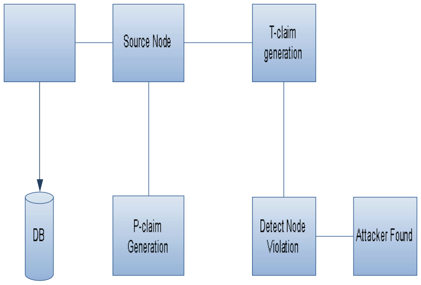


Figure.5. Deny Flood Attacks

The Node it violates its rate limits and they will be found as an attacker and an inconsistent node. The Node it violates its rate limits and they will be found as an attacker and an inconsistent node. It has timestamping, it is used for receiving the send and received time. The send and received time is not same, the timestamping is not valid and it is find the attackers by detect node violation.

It creates the source node dynamically for avoiding flood attacks. In P-Claim generation it claims all the details in encrypted form for avoiding flood attacks. It contains all signatures in encrypted form, it attaches into the source node. In T-Claim generation, it has the packets counting and claims the count to other node. So, we found the hackers by detect node violations and avoid the flood attacks.

When a node forwards a packet, it attaches a T-claim to the packet. Since many packets may be forwarded in a contact and it is expensive to sign each T-claim separately, an efficient signature construction. The node also attaches a P-claim to the packets that are generated by it and have not been sent to other nodes before (called new packet in line 3, Algorithm 1). When a node receives a packet, it gets the P-claim and T-claim included in the packet. It checks them against the claims that it has already collected to detect if there is any inconsistency. Only the P-claims generated in the same time interval (which can be determined by the time tag) are cross-checked. If no inconsistency is detected, this node stores the P-claim and T-claim locally.Suppose two nodes contact and they have a number of packets to forward to each other. Then our protocol is sketched in following Algorithm .

Algorithm . The protocol run by each node in a contact

```
1: Metadata (P-claim and T-claim) exchange and attack detection
2: if Have packets to send then
3: For each new packet, generate a P-claim;
4: For all packets, generate their T-claims and sign
   them with a hash tree;
5: Send every packet with the P-claim and T-claim
   attached;
6: end if
7: if Receive a packet then
8: if Signature verification fails or the count value in its
   P-claim or T-claim is invalid then
9: Discard this packet;
10: end if
11: Check the P-claim against those locally collected and generated in the same time interval to detect
      inconsistency;
12: Check the T-claim against those locally collected for
      inconsistency;
13: if Inconsistency is detected then
14: Tag the signer of the P-claim (T-claim, respectively)
     as an attacker and add it into a blacklist;
15: Disseminate an alarm against the attacker to the
     network;
16: else
17: Store the new P-claim (T-claim, respectively);
18: end if
19: end if
```

The node also attaches a P-claim to the packets that are generated by itself and have not been sent to other nodes before (called new packet in line 3, Algorithm). The packets are reached the receiver and the authorized user alone can access the packet. There are two kinds of packets one is from source and the other one is from individual user data they can accessing it only they can authenticating them. The Pigeonhole principle and claim carry check is more efficient because no need of additional storage.

4. RESULTS AND DISCUSSIONS

To evaluate the performance and cost of our scheme, we run simulations on a synthetic trace generated by the Random Waypoint, mobility model and on the MIT Reality trace collected from the real world. In the synthetic trace, 97 nodes move in a 500 *500 square area with the RWP model. The moving speed is randomly selected from [1, 1,6] to simulate the speed of walking, and the transmission range of each node is 10 to simulate that of Bluetooth. Each simulation lasts 5*10^5 time units.

The MIT Reality trace has been shown, to have social community structures. Ninety seven Smart phones are carried by students and staff at MIT over 10 months. These phones run Bluetooth device discovery every 5 minutes and log about 110 thousand contacts. Each logged contact includes the two contact parties, the start time and duration of the contact. In the simulations, 20 percent of nodes are deployed as attackers. They are randomly deployed or selectively deployed to high-connectivity nodes. The buffer size of each node is 5 MB, the Drop Tail policy is used when buffer overflows. The bandwidth is 2 Mbps. Each node generates packets of 10 KB with random destinations at a uniform rate. Parameter Z =1.

4.1Analysis Verification

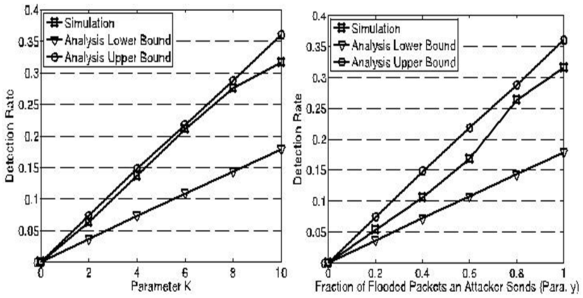
To use the synthetic trace for to verify our analysis results, since in this trace the contacts between node pairs are , which conforms the assumption for the analysis. To divide the trace into 10 segments, each with 5*10^4 time units, and run simulations on each of the third-seventh segments three times with different random seeds. Each data point is averaged over the individual runs. Spray-and-Wait is used as the routing protocol to consider the worst case of packet flood detection.
To evaluate the cost of our scheme in a steady state, no attackers are deployed in this group of simulations.

Table .1.The Storage (KB) Used for Claims and Data Packets

ךּ(days)	10	20	30	40	50
Claims Packets	67 3330	101 3301	125 3321	139 3336	145 3316
Packet Generation Rate(pkt/node/day)	0.1	0.5	1	2	5
Claims packets	65 334	93 1572	113 2596	125 3321	124 3716

Since the receiver does not buffer these packets, it does not store these claims or verify their signatures. When the packet generation rates crosses 1, the signature verification cost turns to decrease. When the packet generation rates crosses 1, the signature verification cost turns to decrease. This is because when the traffic load is high many received packets are dropped due to buffer overflow

Finally, we evaluate the storage cost of our scheme against two factors, the time a claim is stored and the packet generation rate.



(a) Effect of Parameter K (b) Strength of Attack (Para. y)

Figure. 6. Verification of analysis results on the synthetic trace. Spray-and-Wait is used as the routing protocol. Each attacker launches the basic attack once.

Here we only verify the detection probability for the basic attack, since the detection probability for the strong attack can be derived from it in a straightforward way. In this group of simulations, each attacker launches the basic attack once. It sends out two sets of packets to two good nodes with 10 packets in each set (i.e., n=10), and these two sets contain mutually inconsistent packets. We first fix parameter y=1:0 but change parameter K from 0 to 10, and then we fix parameter K=10 but change y from 0 to 1.0. The results are shown in Figs. 6a and 6b, respectively. It can be seen that the simulation results are between the analytical lower bound and upper bound, which verifies the correctness of our analysis.

4.2 Detection Delay

To evaluate the cost of our scheme in a steady state (i.e., all attackers have been detected), no attackers are deployed in this group of simulations. The Reality trace is used. Packets are generated between the 61st and 120th day of the trace, and statistics are collected from the 91th day. By default, each node generates two packets per day, parameter _ (i.e., the time a claim is stored) is 30 days and K is 10.

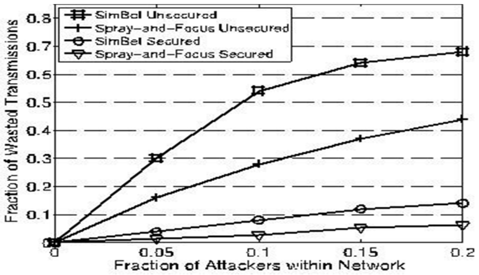


Figure. 7. The effect of undetected replicas on wasted transmissions when attackers collude to launch replica flood attacks.

In a contact, a node may receive some packets but then immediately drop them due to buffer overflow. In such cases, the transmission of the claims attached to these packets is counted into the communication overhead, and the signature generations for these claims are counted into the computation overhead. Since the receiver does not buffer these packets, it does not store these claims or verify their signatures.

5. CONCLUSION

In this paper, we employed 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, we analyzed the lower bound and upper bound of detection probability. Extensive trace-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. The large number of attacker can be handled to avoid providing the flood attacks in Disruption tolerant networks. To implement this on online central authority or infrastructure, this well fits the environment of DTNs. The efficient way of file verification process done on the Receiver side. The Merged File on Receiver will be compared with the original copy of data in the Source.

ACKNOWLEDGMENTS

The author would like to thank the management and staff members of the PRIST University for his help and his thoughtful comments.

REFERENCES

[1] S.J.T.U.Grid Computing Center, “Shanghai Taxi Trace Data,” http://wirelesslab.sjtu.edu.cn/, 2012.

[2] Q. Li, W. Gao, S. Zhu, and G. Cao, “A Routing Protocol for Socially Selfish Delay Tolerant Networks,” Ad Hoc Networks, vol. 10, no. 8, November 2012.

[3] V. Natarajan, Y. Yang, and S. Zhu, “Resource-Misuse Attack Detection in Delay-Tolerant Networks,” Proc. Int’l Performance Computing and Comm. Conf. (IPCCC), 2011.

[4] Z. Zhu and G. Cao, “Applaus: A Privacy-Preserving Location Proof Updating System for Location-Based Services,” IEEE INFOCOM, 2011.

[5] Q. Li, S. Zhu, and G. Cao, “Routing in Socially Selfish Delay Tolerant Networks,” Proc. IEEE INFOCOM, 2010.

[6] Chen and C. Choon, “Mobicent: A Credit-Based Incentive System for Disruption Tolerant Network,” Proc. IEEE INFOCOM, 2010

[7] W. Gao, Q. Li, B. Zhao, and G. Cao, “Multicasting in Delay Tolerant Networks: A Social Network Perspective,” Proc. ACM MobiHoc, 2009.