COMPUTERS & SECURITY 65 (2017) 269–282

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Defense against packet collusion attacks in opportunistic networks

Majeed Alajeely *, *Robin Doss*, *Asma'a Ahmad*, *Vicky Mak-Hau*

School of Information Technology, Deakin University, Geelong, Australia

ARTICLE INFO

Article history:
Received 11 September 2015
Received in revised form 15 November 2016
Accepted 2 December 2016
Available online 8 December 2016

Keywords:
Opportunistic Networks
Security
Packet dropping attacks
Denial-of-service
Malicious node detection

ABSTRACT

Security is a major challenge in Opportunistic Networks (OppNets) because of its characteristics such as an open medium, dynamic topology, no centralized management and absent clear lines of defense. A packet dropping attack is one of the major security threats in OppNets since neither source nor destination nodes have control over the behaviour of intermediate nodes in the network. Consequently, the knowledge of where or when packets are/will be dropped is difficult to gather. In this paper, we present a novel attack and traceback mechanism against a special type of packet dropping attacks – packet collusion attacks, where the malicious node(s) drops some or all packets and then injects new fake packets in their place to mask the packet dropping. Our novel detection and traceback mechanism is based on the concept of a Merkle (or hash) tree and simulation results show it to be highly effective and accurate in terms of detecting attack instances and tracing back to the malicious node(s) in the network that is the attack source.

1. Introduction

Opportunistic networks (OppNets) refers to a number of wireless nodes that opportunistically communicate with each other in the form of a "Store-Carry-Forward" when they come into contact with each other. OppNets exploit human behaviours and social relationships to establish connectivity between mobile users and are particularly attractive for applications such as recommender systems, mobile data offloading and opportunistic computing. The potential for mobile computing to exploit the growth in social networks has increased research interest in OppNets – particularly on the security and privacy challenges that have emerged.

A packet dropping attack is one of the major security threats in OppNets. It can be classified as a denial of service attacks (DoS) where the malicious node drops all or some of the packets. This attack is one of the most difficult DoS attacks to detect since neither source node nor the destination node has knowledge of if, where or when packet dropping has occurred. Malicious packet dropping degrades the performance of the network leading to an increase in the number of packet retransmissions, communication latency and network overhead and obstructs the propagation of sensitive data. Therefore, to fully exploit the benefits of OppNets, we need effective solutions to detect and traceback against different types of packet dropping attacks.

We note however, that packet dropping attacks is not unique to OppNets. They have been studied before in the context of ad hoc networks and wireless sensor networks (Lu and Wong, 2007; Obaidat et al., 2012). However, the existing packet dropping defense mechanisms, such as the multipath routing based mechanisms (Lee and Gerla, 2001; Lu and Wong, 2007; Obaidat et al., 2012), reputation based mechanism (Ke et al., 2010), data provenance based mechanisms (Sultana et al., 2011), are inefficient in OppNets as we do not have end to end connections.

* *Corresponding author.*
E-mail addresses: majeed.alajeely@deakin.edu.au (M. Alajeely), robin.doss@deakin.edu.au (R. Doss), anahmad@deakin.edu.au (A. Ahmad), vicky.mak@deakin.edu.au (V. Mak-Hau).
http://dx.doi.org/10.1016/j.cose.2016.12.001
0167-4048/

Network coding based mechanisms (Chuah and Yang, 2009) are also inefficient as the destination nodes are required to have a copy of all neighbors packets/messages in order to decode its message which is difficult to achieve in OppNets. Encryption techniques (Devi and Damodharan, 2013) are inefficient as well, as we required the use of a secret key which is difficult to manage in OppNets since we have no centralized management. To avoid detection by other mechanisms such as Watchdog and Pathrater mechanism (Marti et al., 2000; Nasser and Chen, 2007; Vasantha and Manimegalai, 2007) an attacker can drop one or more packets and inject fake packets to defeat mechanisms that rely on packet count. We refer to this type of packet dropping attack as a packet collusion attack (i.e., packet dropping and packet injecting) and to the best of our knowledge this is the first work that identifies this type of packet dropping attacks in OppNets. The main contributions of this work are:

1. The identification of a new type of packet dropping attacks – packet collusion attacks in OppNets, where malicious nodes drop some or all packets and then inject fake packets instead to avoid detection by schemes such as Lu and Wong (2007) and Obaidat et al. (2012)
2. An efficient countermeasure against packet collusion attacks, where legitimate nodes detect an attack instance based on the received packets, and then traceback and identify the malicious nodes that triggered the attack.

The remainder of this paper is organized as follows. In Section 2, we present related work. In Section 3, we present the novel attack and our efficient defense. In Section 4, we present our mathematical model. In Section 5, we present our simulation results and in Section 6, we present our conclusion and future work.

2. Related work

Defense mechanisms for packet dropping attacks can use multipath routing based mechanisms where packets are divided into a number of groups and then sent to a destination in more than one path (Lee and Gerla, 2001; Lu and Wong, 2007; Obaidat et al., 2012).

In E-HSAM (Obaidat et al., 2012), a security improvement mechanism is proposed where the packets that go through a path with a malicious node are redirected to an alternative path. However, in OppNets this variety is not always available since there is no end to end connection and no alternative path available all the time. This technique results in network overhead and difficulty in identifying malicious nodes. Moreover, this technique might be vulnerable to route discovery attacks. Lee and Gerla (2001) proposed an on-demand routing protocol by establishing and using multiple routes. This protocol uses a per-packet allocation scheme to spread data packets into multiple paths. This will utilize available network resources and prevent nodes from being traffic congested. Lu and Wong (2007) proposed a distributed, scalable and localized multipath search protocol for discovering multiple node-disjoint paths between the sink and source nodes. The authors also proposed a load balancing mechanism to spread the traffic over the discovered paths.

Acknowledgement based mechanisms can also be used for detecting a packet dropping attack (Baadache and Belmehdi, 2012; Carbunar et al., 2004). This is based on authenticated acknowledgment from the intermediate nodes and the destination within a specific time. The source or destination can detect a malicious node. Baadache and Belmehdi (2012) proposed a mechanism for detecting a packet dropping attack where the intermediate node acknowledges the reception of the packets. A source node used this acknowledgment to construct a Merkle tree, and then compared the value of the tree root with precalculated value. If these values are equal then no packets were dropped in that path, otherwise there was packet dropping. However, this technique can detect a path with a malicious node but is unable to detect the malicious node, therefore it looks for an alternative path for retransmission, thus resulting in network overhead.

Network coding based mechanisms can be used for detection and defense as in Chuah and Yang (2009), where a mitigation scheme to evaluate the impact of the packet selective dropping attack in Delay Tolerant Networks (DTN) is proposed by using network coding. In this scheme the destination node should measure the delivery ratio and sends it back to the sender. The sender then begins adjusting the redundancy factor dynamically to mitigate against the degradation in the delivery ratio caused by the attack. Theoretical analysis and experimental simulations also disclosed some characteristics of the impact of packet dropping on the routing performance, such as delivery ratio, delivery cost and delivery latency. These are degraded if the major nodes behave as packets dropping or behave selfishly. In addition, the impact of the non-cooperative action like selfishness or non-forwarding and dropping of messages in the routing performance where behavior of non-forwarding of messages reduces the delivery cost, while the behavior of dropping messages increases the delivery cost.

Data provenance based mechanisms (Sultana et al., 2011) can be used to identify malicious nodes where the characteristics of the watermarking based secure provenance transmission mechanism and the inter-packet timing characteristics are exploited to achieve this goal. There are three stages to this technique. The first detects lost packets using the distribution of the inter-packet delay. The second identifies the presence of the attack by comparing the empirical average packet loss rate with the natural packet loss rate of the data flow path, and finally the technique identifies a malicious node or link then isolates it by transmitting more provenance information along with the sensor data. However, this technique is not very accurate because it does not detect the exact malicious node in the entire path or link. The impact of TCP packet dropping attacks and detection methods is explored in Zhang et al. (2000). Three dropping mechanisms are investigated. These are periodic packet dropping (PerPD), Retransmission packet dropping (RetPD) and Random packet dropping (RanPD). Statistical based analyses (TDSAM) used for detection of these kinds of attacks are based on the NIDESETAT algorithm running on the ftp client side. However, only one detection technique is proposed in this work without any defense mechanism.