ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementing Filtering Techniques to Prevent Packet Drop Attacks and Detecting Provenance Forgery

Vallavoju Krishna Priya
M.Tech (CSE)
Department of Computer Science and Engineering,
CMR Engineering College,
Kandlakoya, Medchal, Hyderabad, India.

Javvaji Venkatarao
Assistant Professor,
Department of Computer Science and Engineering,
CMR Engineering College,
Kandlakoya, Medchal, Hyderabad, India.

ABSTRACT

Large-scale sensor networks are used in numerous application domains, and the data they collect are used in decision making for critical infrastructures. Data are travelled from multiple sources through intermediate processing nodes that aggregate information. A malicious adversary may introduce additional nodes in the network or compromise existing ones. Therefore, assuring high data trustworthiness is crucial for correct decision-making. Data provenance represents a key factor in evaluating the trustworthiness of sensor data. Provenance management for sensor networks introduces several challenging requirements, such as low energy and bandwidth consumption, efficient storage and secure transmission. I implemented the filtering techniques to securely transmit provenance for sensor data. The proposed technique relies on in-packet Bloom filters to encode provenance. I introduce efficient mechanisms for provenance verification and reconstruction at the base station. In addition, we extend the secure provenance scheme with functionality to detect packet drop attacks staged by malicious data forwarding nodes. I evaluate the proposed technique both analytically and empirically, and the results prove the effectiveness and efficiency of the implementing filtering techniques in detecting packet forgery and loss attacks. As opposed to existing research that employs separate transmission channels for data and provenance. In contrast, only require a single transmission channel for both data and provenance. Furthermore, traditional provenance security solutions use intensively cryptography and digital signatures, and they employ append-based data structures to store provenance, leading to prohibitive costs. In contrast, I use only fast Message Authentication Code (MAC) schemes and Bloom filters (BF), which are fixed-size data structures that compactly represent provenance.

Introduction

Sensor networks are used in many application domains, such as cyber physical infrastructure systems, environmental monitoring, power grids, etc. Data are produced at a large number of sensor node sources and processed in-network at intermediate hops on their way to a Base Station (BS) that performs decision-making.

The diversity of data sources creates the need to assure the trustworthiness of data, such that only trustworthy information is considered in the decision process. Data provenance is an effective method to assess data trustworthiness, since it summarizes the history of ownership and the actions performed on the data.

Recent research [12] highlighted the key contribution of provenance in systems where the use of untrustworthy data may lead to catastrophic failures (e.g., SCADA systems). Although provenance modeling, collection, and querying have been studied extensively for workflows and curated databases [13], [14], provenance in sensor networks has not been properly addressed. I investigate the problem of secure and efficient provenance transmission and processing for sensor networks, and i use provenance to detect packet loss attacks staged by malicious sensor nodes.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

In a multi-hop sensor network, data provenance allows the Base Station (BS) to trace the source and forwarding path of an individual data packet. Provenance must be recorded for each packet, but important challenges arise due to the tight storage, energy and bandwidth constraints of sensor nodes [14]. Therefore, it is necessary to devise a light-weight provenance solution with low overhead. Furthermore, sensors often operate in an untrusted environment, where they may be subject to attacks.

Hence, it is necessary to address security requirements such as confidentiality, integrity and freshness of provenance. My goal is to design a provenance encoding and decoding mechanism that satisfies such security and performance needs.

A provenance encoding strategy whereby each node on the path of a data packet securely embeds provenance information within a Bloom filter that is transmitted along with the data.

Upon receiving the packet, the BS extracts and verifies the provenance information. I also devise an extension of the provenance encoding scheme that allows the Base Station (BS) to detect if a packet drop attack was staged by a malicious node.

As opposed to existing research that employs separate transmission channels for data and provenance [15], and only require a single channel for both. Furthermore, traditional provenance security solutions use intensively cryptography and digital signatures [16], and they employ append-based data structures to store provenance, leading to prohibitive costs. In contrast, I use only fast Message Authentication Code (MAC) schemes and Bloom filters (BF), which are fixed-size data structures that compactly represent provenance.

Bloom filters make efficient usage of bandwidth, and they yield low error rates in practice. My specific contributions are:

LITERATURE SURVEY

Wireless networks are vulnerable to spoofing attacks, which allows for many other forms of attacks on the networks. Although the identity of a node can be verified through cryptographic authentication, authentication is not always possible because it requires key management and additional infrastructural overhead [1]. In this project, a method for both detecting spoofing attacks, as well as locating the positions of adversaries performing the attacks. First an attack detector for wireless spoofing that utilizes K-means cluster analysis. Next, describes how integrated our attack detector into a real time indoor localization system, which is also capable of localizing the positions of the attackers. Then show that the positions of the attackers can be localized using either area-based or point-based localization algorithms with the same relative errors as in the normal case. Then evaluated the methods through experimentation using both an 802.11 (WiFi) network as well as an 802.15.4 (ZigBee) network. My results show that it is possible to detect wireless spoofing with both a high detection rate and a low false positive rate, thereby providing strong evidence of the effectiveness of the K-means spoofing detector as well as the attack localizer.

I describe possible denial of service attacks to infrastructure wireless 802.11 networks. To carry out such attacks only commodity hardware and software components are required [2]. The results show that serious vulnerabilities exist in different access points and that a single malicious station can easily hinder any legitimate communication within a basic service set.

Wireless networks are vulnerable to many identity-based attacks in which a malicious device uses forged MAC addresses to masquerade as a specific client or to create multiple illegitimate identities [3]. For example, several link-layer services in IEEE 802.11 networks have been shown to be vulnerable to such attacks even when 802.11i/1X and other security mechanisms are deployed.

In this project, a transmitting device can be robustly identified by its signal print, a tuple of signal strength

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

values reported by access points acting as sensors. A different from MAC addresses or other packet contents, attackers do not have as much control regarding the *signalprints* they produce .Moreover, using measurements in a testbed network, and demonstrates that signalprints are strongly correlated with the physical location of clients, with similar values found mostly in close proximity. By tagging suspicious packets with their corresponding signalprints, the network is able to robustly identify each transmitter independently of packet contents, allowing detection of a large class of identity-based attacks with high probability.

Existing System

- Recent research highlighted the key contribution of provenance in systems where the use of untrustworthy data may lead to catastrophic failures (e.g., SCADA systems).
- Although provenance modeling, collection, and querying have been studied extensively for workflows and curated databases, provenance in sensor networks has not been properly addressed.

Disadvantages of Existing System

- Traditional provenance security solutions use intensively cryptography and digital signatures, and they employ append-based data structures to store provenance, leading to prohibitive costs.
- Existing research employs separate transmission channels for data and provenance.
- In sensor networks, provenance has not been properly addressed.
- System accessed by the attackers due to security violence.

Proposed System

- I investigate the problem of secure and efficient provenance transmission and processing for sensor networks, and I use provenance to detect packet loss attacks staged by malicious sensor nodes.
- My goal is to design a provenance encoding and decoding mechanism that satisfies such security and performance needs. I propose a provenance encoding strategy whereby each node on the path of a data packet securely embeds provenance information within a Bloom filter (BF) that is transmitted along with the data. Upon receiving the packet, the BS extracts and verifies the provenance information. I also devise an extension of the provenance encoding scheme that allows the BS to detect if a packet drop attack was staged by a malicious node [8].

Advantages of Proposed System

- I use only fast message authentication code (MAC) schemes and Bloom filters, which are fixed-size data structures that compactly represent provenance. Bloom filters make efficient usage of bandwidth, and they yield low error rates in practice.
- I formulate the problem of secure provenance transmission in sensor networks, and identify the challenges specific to this context.
- I propose an in-packet Bloom filter (iBF) provenance-encoding scheme [19].
- I design efficient techniques for provenance decoding and verification at the base station.
- I extend the secure provenance encoding scheme and devise a mechanism that detects packet drop attacks staged by malicious forwarding sensor nodes.
- I perform a detailed security analysis and performance evaluation of the proposed provenance encoding scheme and packet loss detection mechanism.
- And only require a single channel for both transmission channels for data and provenance.

Proposed System Algorithm

Bloom Filters Algorithm

Bloom filters are compact data structures for probabilistic representation of a set in order to support membership queries (i.e. queries that ask: "Is element *X*

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

in set *Y*?"). This compact representation is the payoff for allowing a small rate of *false positives* in membership queries; that is, queries might incorrectly recognize an element as member of the set. I succinctly present Bloom filters use to date in the next section. In Section 3 we describe Bloom filters in detail, and in Section 4 we give a hopefully precise picture of space/computing time/error rate tradeoffs.

Constructing Bloom Filters

Consider a set $A = \{a_1, a_2, ..., a_n\}$ of n elements. Bloom filters describe membership information of *A* using a bit vector *V* of length *m*. For this, *k* hash functions, $h_1, h_2, ..., h_k$ with $h_i : X \to \{1..m\}$, are used as described below:

The following procedure builds an *m* bits Bloom filter, corresponding to a set A and using $h_1, h_2, ..., h_k$ hash functions:

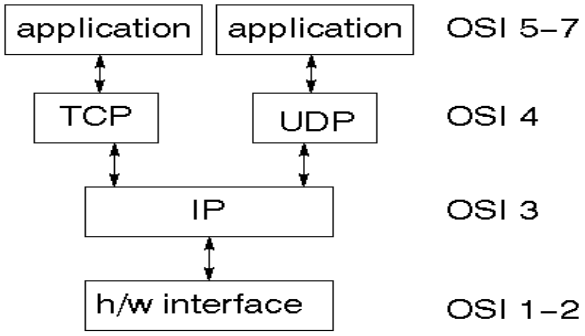
```
Procedure BloomFilter(set A, hash_functions, integer m)
                        returns filter
    filter = allocate m bits initialized to 0
    foreach a_i in A:
        foreach hash function h_j:
            filter[h_j(a_i)] = 1
    end foreach
    end foreach
    return filter
```

Therefore, if a_i is member of a set *A*, in the resulting Bloom filter *V* all bits obtained corresponding to the hashed values of a_i are set to 1. Testing for membership of an element *elm* is equivalent to testing that all corresponding bits of *V* are set:

Networking

TCP/IP stack

The TCP/IP stack is shorter than the OSI one:



TCP is a connection-oriented protocol; UDP is a connectionless protocol.

Figure 6.5: TCP/IP stack

IP datagram's

The IP layer provides a connectionless and unreliable delivery system. It considers each datagram independently of the others. Any association between datagram must be supplied by the higher layers. The IP layer supplies a checksum that includes its own header. The header includes the source and destination addresses. The IP layer handles routing through an Internet. It is also responsible for breaking up large datagram into smaller ones for transmission and reassembling them at the other end.

UDP

UDP is also connectionless and unreliable. What it adds to IP is a checksum for the contents of the datagram and port numbers. These are used to give a client/server model - see later.

TCP

TCP supplies logic to give a reliable connection-oriented protocol above IP. It provides a virtual circuit that two processes can use to communicate.

Internet addresses

In order to use a service, you must be able to find it. The Internet uses an address scheme for machines so that they can be located. The address is a 32 bit integer which gives the IP address. This encodes a network ID and more addressing. The network ID falls into various classes according to the size of the network address.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Network address

Class A uses 8 bits for the network address with 24 bits left over for other addressing. Class B uses 16 bit network addressing. Class C uses 24 bit network addressing and class D uses all 32.

Subnet address

Internally, the UNIX network is divided into sub networks. Building 11 is currently on one sub network and uses 10-bit addressing, allowing 1024 different hosts.

Host address

8 bits are finally used for host addresses within our subnet. This places a limit of 256 machines that can be on the subnet.

Total address

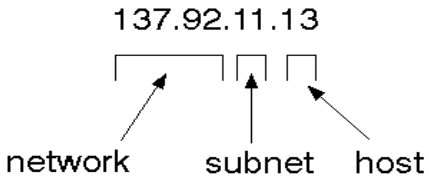


Figure 6.6: Total Address

The 32 bit address is usually written as 4 integers separated by dots.

Port addresses

A service exists on a host, and is identified by its port. This is a 16 bit number. To send a message to a server, you send it to the port for that service of the host that it is running on. This is not location transparency! Certain of these ports are "well known".

Sockets

A socket is a data structure maintained by the system to handle network connections. A socket is created using the call socket. It returns an integer that is like a file descriptor. In fact, under Windows, this handle can be used with Read File and Write File functions.

```
#include <sys/types.h>
#include <sys/socket.h>
int socket(int family, int type, int protocol);
```

Here "family" will be AF_INET for IP communications, protocol will be zero, and type will depend on whether TCP or UDP is used. Two processes wishing to communicate over a network create a socket each. These are similar to two ends of a pipe - but the actual pipe does not yet exist.

Result

IDS Manager (Router)

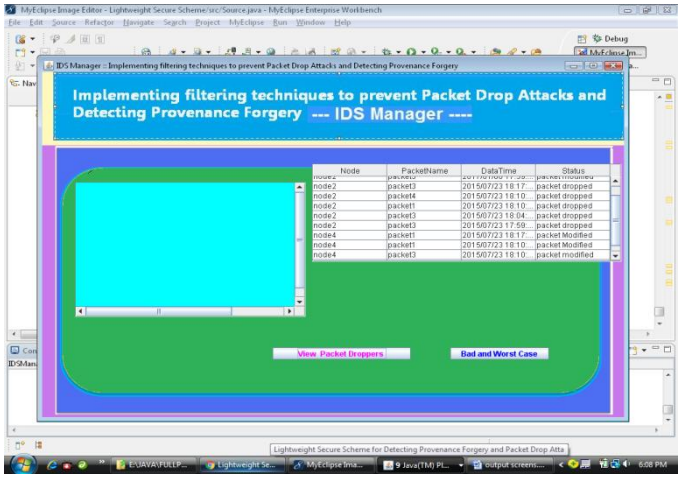


Figure 9.14: Screen Shot of IDS Manager

IDS Manager verifies all packets which are dropped or not. And it also verifies the packets which are modified or not and it can identify the modifiers in the process based on the bit identification and performs Ranking for each node based on the Category of nodes. Sink gives ranking like Good, Temporarily Good, Suspiciously Bad, Bad based on the node behavior in the process.

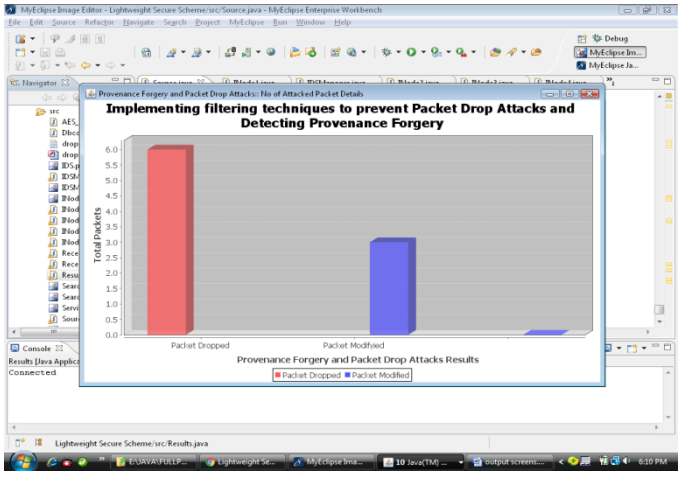


Figure 9.15: Screen Shot of Result

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

This figure shows the result of how many packets are dropped and how many packets are modified i.e., the number of attacked packets details.

Conclusion

I investigated the problem of securely transmitting provenance for sensor networks, and proposed a provenance encoding and decoding scheme based on Bloom filters. The scheme ensures confidentiality, integrity and freshness of provenance. I extended the scheme to incorporate data-provenance binding, and to include packet sequence information that supports detection of packet loss attacks. Experimental and analytical evaluation results show that the proposed scheme is effective, light-weight and scalable.

Future Scope

In future work, I plan to implement a real system prototype of our secure provenance scheme, and to improve the accuracy of packet loss detection, especially in the case of multiple consecutive malicious sensor nodes.

REFERENCES

[1] Y. Chen, W. Trappe, P. Martin, "Detecting and Localizing Wireless Spoofing Attacks," 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2007, pp. 193-202.

[2] F. Ferreri, M. Bernaschi, L. Valcamonici, "Access points vulnerabilities to DoS attacks in 802.11 networks," IEEE Wireless Communications and Networking Conference(WCNC), Vol.1, 2004, pp. 634 – 638.

[3] Sudip Misra, Ashim Ghosh, A. P. Sagar, Mohammad S. Obaidat, "Detecting Identity Based Attacks in Wireless Networks Using Signal prints," IEEE/ACM International Conference on Green Computing and Communications (GreenCom) and Cyber, Physical and Social Computing (CPSCom), 2010, pp. 35 – 41.

[4] B. Wu, J. Wu, B. Fernandez, S. Magliveras, "Secure and Efficient Key Management in Mobile Ad Hoc Networks," in Proc. Of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2005, pp.1530-2075.

[5] L. Sang, A. Arora, "Spatial Signatures for Lightweight Security in Wireless Sensor Networks" IEEE INFOCOM - The 27th Conference on Computer Communications, 2008, pp. 176-180.

[6] Sultana. S, Ghinita. G, Bertino. E and Shehab. M, "A Lightweight Secure Scheme for Detecting Provenance Forgery and Packet Drop Attacks in Wireless Sensor Networks," IEEE Transactions On Dependable And Secure Computing, Vol. 12(3):256-269, May/June 2015.

[7] Tharani. M, Sivachandran. K and Saranya. S.N, "An Efficient Detection of Forgery And Packet Drop Attacks In Wireless Sensor Networks," International Journal of Advanced Information and Communication Technology (IJAICT), Vol. 2(7): 1055-1058, November,2015.

[8] Adhau. R, Ambekar. A, Drakshe. A and Thorave. R, "Detecting Attacks in Wireless Sensor Network through Bloom Filtering," International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 6(5): 4397-4399, 2015.

[9] Kadam. M, Dakhore. S, Chavan. K and Bandgar. A, "A Secure Model For Detecting Origin Forgery And Packet Drop Attacks In WSN," Multidisciplinary Journal of Research in Engineering and Technology, Vol. 2(4): 823-828, 2015

[10] Dinde. R, Jain. A, Thorkar. S, Patil. A "Provenance Forgery and Packet Drop Attacks Detection in Wireless Networks," International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), Vol. 4(2): 2560-2565, February 2016.

[11] Sultana. S, Ghinita. G, Bertino. E, Shehab. M, "A Lightweight Secure Provenance Scheme for Wireless

Sensor Networks," IEEE Computer Society-International Conference on Parallel and Distributed System, pp.101-108, 2012.

[12] H. Lim, Y. Moon, and E. Bertino, "Provenance-based trustworthiness assessment in Sensor Networks," in Proc. Of Data Management for Sensor Networks, 2010, pp.2–7.

[13] I. Foster, J. Vockler, M. Wilde, and Y. Zhao, "Chimera: A virtual data system for representing, querying, and automating derivation," in Proc. of Conf. on Scientific and Statistical Database Management, 2002, pp. 37–46.

[14] K. Muniswamy-Reddy, D. Holland, U. Braun, and M. Seltzer, "Provenance-aware Storage systems," in Proc. of the USENIX AnnualTechnical Conf., 2006, pp. 4–4.

[15] Y. Simmhan, B. Plale, and D. Gannon, "A survey of data provenance in e-science," SIGMOD Record, vol. 34, pp.31–36, 2005.

[16] R. Hasan, R. Sion, and M. Winslett, "The case of the fake picasso: Preventing history forgery with secure provenance," in Proc.Of FAST, 2009, pp.1–14.

[17] S. Sultana, E. Bertino, and M. Shehab, "A provenance based mechanism to identify Malicious packet dropping adversaries in sensor networks," in Proc. of ICDCS Workshops, 2011, pp. 332–338.

[18] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, "Summary cache: a scalable wide-area web cache sharing protocol," IEEE/ACM Trans. Netw., vol. 8, no. 3, pp. 281–293, Jun. 2000.

[19] C. Rothenberg, C. Macapuna, M. Magalhaes, F. V rdi, and A. Wiesmaier, "In-packet bloom filters: Design and networking applications," Computer Networks, vol. 55, no. 6, pp. 1364 – 1378, 2011.