Efficient Broadcast Authentication for Wireless Sensor Networks

Erman Ayday, Farshid Delgosha and Faramarz Fekri
School of Electrical and Computer Eng.
Georgia Institute of Technology
Atlanta, GA 30332-0250, USA
Email: {erman, fekri}@ece.gatech.edu
delgosha@ieee.org

***Abstract*—This work proposes a reliable and secure broadcast protocol for ad hoc wireless (sensor) networks. Since reliability and security compete for the same resources, we jointly solve for error control coding (to achieve packet reliability) and integrity for a broadcast scenario. We assume that the packets sent by the source node would travel in a hop-by-hop fashion to arrive at other nodes. Hence, it is very important that data packets are received by all nodes in the network using minimum number of transmissions and with a minimum latency because of the limited resources of the sensor nodes and the urgency of the information. We assume that the adversary can drop (or modify) legitimate packets and inject its own packets by compromising nodes. Thus, each receiver node in the network should make sure that packets they receive is indeed generated by the source node and are not modified or injected on the way by possible malicious nodes. It is critical that, any node receiving a malicious packet immediately filters it out and uses only the legitimate ones for forwarding to the next hop and decoding. This makes the packet authentication a very challenging problem for broadcast. We build our authentication scheme, on top of a reliable and energy-efficient broadcasting protocol called *Collaborative Rateless Broadcast* (CRBcast) to achieve efficiency, reliability and authenticity.**

I. Introduction

Reliably broadcasting messages is necessary in many applications in wireless sensor networks. For example, the sink may need to re-program the sensor or actuator nodes to change their behavior in order to adapt to new application requirements or new environmental conditions. Node-to-network multi-hop broadcasting can serve as an efficient solution for the sensors to share their local measurements among each other [1]. Another application is in distributed detection of node replication attacks in which each node in the network uses an authenticated broadcast message to flood the network with its location information. Each node stores the location information for its neighbors, and if it receives a conflicting claim, revokes the offending node [2].

Typical attacks that an adversary may launch to interfere in the normal operation of a broadcast protocol are:

- *Data Drop:* An insider node drops a legitimate report on the forwarding path toward the sink.
- *Bogus Packet Injection* and *Packet Modification:* The adversary injects bogus packets or modifies the contents of legitimate reports. Therefore, the sink may not be able to retrieve the original message or may get the message with a high latency.

Cryptographic services required to prevent these attacks are *data availability* and *data authenticity*, respectively.

In this work, based on the CRBcast broadcast protocol [3], we design a scheme that provides the aforementioned security services with moderate communication and computation overheads. CRBcast is inspired and evolved from a mechanism that is recently developed in the context of reliable information delivery by rateless codes [4]. Furthermore, we show that our scheme has a considerable low latency when compared to previously proposed schemes.

A. Outline of Our Scheme

In this work, we propose an efficient authentication scheme for node-to-network multi-hop broadcast. The proposed scheme, provides data authenticity and availability with low communication and computation overheads as well as with minimum latency. In contrast to previous schemes, we integrate the design of our authentication protocol with that of the CRBcast of [3]. The CRBcast consists of two phases: probabilistic routing and completing the missing packets. It employs rateless coding [4]–[6]. The proposed scheme starts with the source node generating the packets and the authentication information for both phases of the CRBcast. Then, it broadcasts, in a multi-hop fashion, the coded packets along with the authentication information. A node receiving this information verifies the authenticity of the packets and drops bogus packets. In the next phase, nodes with insufficient number of packets (to reconstruct the original message) request extra packets from those who have sufficient number of packets (complete nodes). We assume the existence of an underlying MAC (medium access control) protocol for the channel access. For this purpose, we used the sensor-MAC (S-MAC) in [7] for our simulations to analyze the latency and the impact of adversary. Hence, during both phase I and phase II, when a node gets the channel, its one and two hop neighbors remain silent until the transmission is over to avoid collusion.

The main contributions of our scheme are summarized in the following.

1) The proposed scheme is designed based on an existing broadcast protocol with the same nature. Hence, in terms of computation and communication overheads, it is more efficient than other schemes that all use flooding.
2) Rateless coding intrinsically provides data availability by allowing data processing at every node. This feature is lacking in previously proposed schemes.
3) Nodes individually authenticate each received packet. Therefore, the receivers can immediately filter out bogus packets and save energy.
4) Authentication information transmitted by the source can be used to detect malicious nodes in the network. The legitimate nodes may choose to prevent the detected malicious nodes from getting the channel.

5) The proposed scheme has a considerably low latency (even in the presence of the malicious nodes) when compared to previously proposed schemes that are using flooding technique.

B. Notations

In order to facilitate future references, frequently used notations are listed below with their meanings.

N	Total number of nodes in the network
p	Probability of forwarding in phase I
t	Number of data packets to be sent from the source
ℓ	Number of partitions in the second phase
P_i	The i-th encoded packet during phase I
Q_i	The i-th encoded packet during phase II
G_i	The i-th partition during phase II
M	Output of the Bloom filter
A	Authentication information

II. Description of the Scheme

In this section, we explain the two phases of the proposed scheme in detail.

A. Phase I

Phase I consists of three steps: (1) generating the report and encoding the data packets at the source, (2) generating authentication information, and (3) verifying the authenticity of the received packets by the nodes. In the following, we provide details of these steps.

Upon obtaining information critical to the entire network, a source node generates the t packets $w_1, \ldots, w_t$. Using ***RatelessI***, the source node generates the encoded packets $P_1, \ldots, P_T$, where $T = t\gamma$ and $\gamma > 1$. In ***RatelessI***, the linear coefficients are randomly driven from an optimized distribution [4].

The source generates authentication information partially using a Bloom filter. The Bloom filter takes the encoded packets $P_1, \ldots, P_T$ as inputs and employs k independent hash functions $H_1, \ldots, H_k$. The output of the Bloom filter, an array M of bit length m, forms a piece of the authentication information.

Another piece of the authentication information belongs to the phase II. In phase II, the encoded packets are generated from the original data known to the source. Moreover, all complete nodes generate the same set of encoded packets using ***RatelessII***. The linear coefficients employed in ***RatelessII*** are generated using a pseudorandom function based on an optimized distribution that is known to all nodes. We assume that all nodes have access to the same pseudorandom function and employ the same seed [1] to generate random coefficients. Hence, using ***RatelessII***, all nodes generate the same set of coefficients.

Therefore, the source generates authentication information for phase II as well. Let $Q_1, \ldots, Q_T$ be the encoded packets generated using ***RatelessII***. These packets are partitioned into ℓ groups $G_1, \ldots, G_\ell$. Assuming that $j = T/\ell$ is an integer, the ℓ groups are related to the encoded packets as follows.

$$G_i = \left[Q_{1+(i-1)j}, \ldots, Q_{ij} \right], \qquad \forall i = 1, \ldots, \ell \tag{1}$$

Eventually, the source compiles the authentication information required for both phases as:

$$A = ID \| M \| H\left(G_1\right) \| \cdots \| H\left(G_\ell\right), \tag{2}$$

[1]The seed is updated after every broadcast session.

where $H(\cdot)$ is a cryptographically secure hash function and ID is the ID of the source node. To prevent an adversary from modifying the authentication information, the source node signs A using an efficient signature scheme $Sign(\cdot)$ enhanced for use in wireless sensor networks [8]. Eventually, the source broadcasts the authentication information $(A, Sign(A), Ver)$ in a multi-hop fashion. Here, *Ver* is the description of the signature verification algorithm. We note that the source node initiates flooding the network with the authentication information. Other nodes in the network broadcast this information to their neighbors. Every node receiving the authentication information, first verifies its integrity using $Sign(A)$. If it is verified, the node broadcasts it with definite probability 1.

For simplicity, we assume that every node has access to the algorithm for verifying the integrity of the authentication information. Hence, we propose to use ID-based signature schemes [9]. In such schemes, the verification algorithm is obtained from the ID of the source node generating the signature.

After the broadcast of the authentication information, encoded packets are broadcast to the network in a hop-by-hop fashion. Every node receiving these packets forwards each one of them with a probability p. Since the information relay is probabilistic, none of the nodes can determine the packets it is going to receive beforehand. Therefore, every forwarding packet has to be authenticated individually by each node.

Every node receiving packets encoded with ***RatelessI***, first verifies the authenticity of each packet individually using the Bloom filter output M. The receiver node employs k independent hash operations to a packet and decides whether the result is consistent with M or not. Every packet authenticated by a receiving node is forwarded with probability p. Otherwise, it is dropped, and the receiving node will not accept any other packets from the transmitter of these packets. Equivalently, the receiver will not let the detected malicious node to get the channel again.

B. Phase II

In phase II, complete nodes advertise their completeness to their neighbors by broadcasting ADV messages. Incomplete nodes respond by sending a request message REQ that includes the number of required packets. Complete nodes send packet groups $G_1, \ldots, G_\ell$ instead of the encoded packets $Q_1, \ldots, Q_T$. Since incomplete nodes receive their requested packets from complete ones, they are not required to verify the authenticity of packets individually.

Let c be the maximum number of packets requested from a complete node. This node broadcasts $G_1, \ldots, G_s$, where $s = \lceil (c/j) \rceil$. Using the authentication information A, incomplete nodes verify the authenticity of the blocks instead of individual packets.

Failure to authenticate blocks implies that the complete node sending them is malicious. Hence, the receiver will no longer accept any packets from that specific complete node (will not let that node to get the channel again). In this case, the incomplete node waits for an ADV message from another complete node. We note that in our scheme every node is able to detect malicious nodes individually without using expensive and vulnerable voting systems.

It is worth noting that, there is no strict boundary between phase I and phase II. As soon as a node becomes complete, it starts phase II and sends ADV message to its neighbors. During

this time other nodes may keep on pursuing phase I at other parts of the network.

III. Analysis and Comparison

In this section, we analyze the security of our scheme in terms of data authenticity and data availability. Moreover, we compare our scheme with a recently proposed broadcast authentication scheme in [10]. We assume that the network is connected unless otherwise stated.

A. Data Authenticity

Assuming the legitimacy of the source node, the adversary cannot modify the report at its generation time. Moreover, adversary cannot deceive receivers by modifying the message since authentication information is provided and digitally signed by the source node. We note that, a receiver node do not accept any data packets before receiving the legitimate authentication information from the source. A bogus packet injected during phase I is filtered out with a high probability after one hop travel. The filtering strength of the protocol depends on the false positive probability of the Bloom filter. The network designer can arbitrarily decrease this probability to the expense of increasing communication overhead.

An adversary may attempt to inject bogus packets to the network in any one of the two phases of the protocol. However, an illegitimate packet received by a node is filtered out using the authentication information. After this event, the impersonating node is rejected by legitimate nodes.

B. Data Availability

As opposed to previous works, we define the data availability based on the latency. In other words, for 100% availability, all nodes in the network need to become complete in a definite time.

Using computer simulations, we have studied data availability in adversarial environments where malicious nodes either drop or modify legitimate packets. In our simulations, we have assumed $N = 1000$, $T = 100$, $r = 0.2$, and the size of the deployment field is 2×2. We consider both the energy consumption of the network and the latency versus the forwarding probability p. Hence we created the energy consumption-latency metric as

$$E - L = \frac{N_{tx}(i)}{min(N_{tx})} \times \frac{latency(i)}{min(latency)} \quad (3)$$

where N_{tx} is the number of transmissions per node for $p = i$ ($i = 0, 0.1, ..., 1$) and $min(N_{tx})$ is the minimum number of transmissions per node among all p values. It is worth noting that we assumed 1 *packet transmission* $=$ 1 *time-unit* upon calculating the latency. Hence we obtained an optimal value for p as in Figure 1. We also simulated the same network for [10] and observed $E - L = 6.0679$ which is more than 4 times larger than our optimal value. As a result, our scheme gives more efficient results both in terms of energy consumption and latency when compared to [10]. Hence, we can say that our scheme outperforms all previous schemes that are using flooding technique, including [10].

As a future work, we will provide the performance of our scheme in the presence of malicious nodes and will show that our scheme provides a considerable data availability with respect to the previous schemes. Moreover, we will find the most effective attack type for the adversary and show that our scheme can remain robust even against those intelligent attacks.

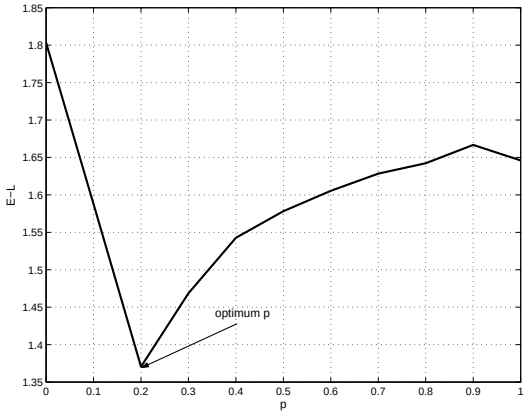

Fig. 1. Energy consumption-Latency metric versus p.

IV. Conclusions

This work investigates reliability and authentication of broadcast in ad hoc wireless (sensor) networks. We propose a node-to-network multi-hop broadcasting scheme by simultaneously considering these two features in our design. Our scheme provides reliability and authenticity to the broadcast with minimum number of transmissions and minimum amount of latency. Therefore, it is suitable for networks of sensors with limited resources. The network may operate in an adversarial environment where an adversary may physically capture nodes, drops or modifies packets, and injects bogus packets. For the future work, we will investigate the impact of adversarial nodes on our performance metric more deeply and compare the results with the existing schemes.

References

[1] Y.-W. Hong and A. Scaglione, "Energy-efficient broadcasting with cooperative transmissions in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 5, no. 10, pp. 2844–2855, Oct. 2006.

[2] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *05*. CA: IEEE Comput. Soc., May 2005, pp. 49–63.

[3] N. Rahnavard and F. Fekri, "CRBcast: A collaborative rateless scheme for reliable and energy-efficient broadcasting in wireless sensor networks," in *Proc. Int. Symp. Inform. Process. Sensor Networks - IPSN'06*. TN: ACM Press, Apr. 2006, pp. 276–283.

[4] M. Luby, "LT codes," in *Proc. IEEE Symp. Found. Comput. Science*. Vancouver: IEEE Comput. Soc., Nov. 2002, pp. 271–280.

[5] P. Maymounkov, "Online codes," New York University,, Tech. Rep. TR2002-833, Nov. 2002, available Online at: http://pdos.csail.mit.edu/~petar/papers/maymounkov-online.pdf.

[6] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[7] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proc. IEEE Conf. Comput. Commun. - INFOCOM'02*, 2002.

[8] G. Gaubatz, E. O. Jens-Peter Kaps, and B. Sunar, "State of the art in ultra-low power public key cryptography for wireless sensor networks," in *05*. HI: IEEE Comput. Soc., Mar. 2005, pp. 146–150.

[9] W. Lee and W. Sriborrirux, "Optimizing authentication mechanisms using ID-based cryptography in ad hoc wireless mobile networks," in *Int. Conf. Inform. Netw. - ICOIN'04*, ser. Lecture Notes in Computer Science, H.-K. Kahng and S. Goto, Eds., vol. 3090. Berlin: Springer-Verlag, Feb. 2004, pp. 925–934.

[10] P. K. Dutta, J. W. Hui, D. C. Chu, and D. E. Culler, "Securing the deluge network programming system," in *Proc. Int. Symp. Inform. Process. Sensor Networks - IPSN'06*. TN: ACM Press, Apr. 2006, pp. 326–333.