Aliased Detection Mode for Detecting Clone Attacks in Wireless Sensor

M. Nagarani, M. Shirisha
1Dept. of CSE, Gnyana Saraswati College of Engineering and Technology, Warangal, India
2Dept. of CSE, Vaagdevi Engineering College Nizamabad, Warangal, India

Abstract
A central problem in sensor network security is that sensors are susceptible to physical capture attacks. Once a sensor is compromised, the adversary can easily launch clone attacks by replicating the compromised node, distributing the clones throughout the network, and starting a variety of insider attacks. In this paper, we consider fingerprint generation. The fingerprint verification is conducted at both the base station and the neighboring sensors, which ensures a high detection probability. In this paper we are extending this approach by introducing new enhancement called Alias Detection Mode (ADM), our system extends the fingerprint generation, which maintains the node behavior which maintains the node sensitivity by specifying automatic range values determined by the network modulation at the node creation. The system expects the range of the node while node entering into the sensor network. Range values incorporate the fingerprint of each sensor which leads to detect the clone attack in the sensor network. The security and performance analysis indicate that our extended fingerprint generation algorithm can identify clone attacks with a high detection probability at the cost of a low computation/communication/storage overhead.

Keywords
Wireless Sensor Network, Cloning Attack, Adversary Node

I. Introduction
In sensor networks, adversaries may easily capture and compromise sensors and deploy unlimited number of clones of the compromised nodes. Since these clones have legitimate access to the network (legitimate IDs, keys, other security credentials, etc.), they can participate in the network operations in the same way as a legitimate node, and thus launch a large variety of insider attacks [1, 8, 15], or even take over the network. If these clones are left undetected, the network is unshielded to attackers and thus extremely vulnerable. Therefore, clone detectors are severely destructive, and effective and efficient solutions for clone attack detection are needed to limit their damage. Clones may collude to cheat the network administrator into believing that they are legitimate. Note that an adversary may distribute clone nodes anywhere in the network. Thus localized detection schemes do not work effectively. Most existing research efforts in sensor networks against clone attacks focus on preventive technologies rather than reactive techniques, e.g., key schemes to prevent sensors from being compromised. Unfortunately, most of these preventive technologies (i.e., key schemes) may easily lose their power against clone attacks [3]. Therefore it is imperative to provide effective/efficient clone attack detection. Actually clone detection in sensor networks is a relatively overlooked area. In this paper we are proposing a new enhanced method for clone detection over the internet, in which node behavior are calculated according to predefined range value to the sensor in an active network modulation. Range values are specified at sensor creation time which is incorporates the fingerprint verification, this enhanced method called Alias Detection Mode (ADM), our scheme is the first to provide real-time detection of clone attacks in an effective and efficient way.

Compared with the existent work against clone attacks in sensor networks [3, 12], our algorithm has the following characteristics and advantages:
1. Our scheme explores the superimposed s-disjunct code for a timely clone attack detection. A fingerprint can be easily encoded with a very short bit stream, which results in small message overhead.
2. Our scheme can identify cloned sensors with a high detection accuracy at the expense of a very low communication/computation/storage overhead.
3. Our scheme is robust against collaborative clone attackers, and has no limitation on the number of compromised/ cloned sensors, which is a significant improvement compared to the existent works [3, 12].
4. Our scheme conducts fingerprint verification locally (via neighboring nodes) and globally (via the base station) for each message broadcasted by any node, therefore clone attackers can be detected in real-time.

II. Related Work
A straightforward solution to defend against clone attacks is to let the base station collect the neighborhood information (e.g. location, neighbor list, etc.) from each sensor and monitor the network in a centralized way. This approach suffers from high communication overhead by requesting redundant information from the network. For a sensor u, the neighbors should register u’s ID and location at multiple witness nodes. The witness nodes can be either randomly selected throughout the network, or simply picked up along a routing path. Any witness node having received conflicting reports about the same sensor, should initiate a revoke message. For a high detection probability, this witness-based scheme exploits flooding for information exchange and thus results in a high communication overhead. Also, the scheme relies on public key cryptography, which is expensive for most mote-like sensors. The storage overhead is also high since each sensor needs to store enough public keys of the others.

A. Network Model
Static homogeneous sensor network, in which a base station (BS) intermittently collects data from multi hops away. There exist N resource constrained sensors in the network, whose positions can be determined after deployment via a self-positioning mechanism such as those proposed in [2, 9, 13]. Let N(u) denote an open neighborhood of u that contains n nearest neighbors. Note that N(u) could be the one-hop neighborhood or any neighboring area containing n closest nodes. We denote N(N(u)) the cumulative neighborhood of N(u). For any two arbitrary neighboring nodes u and v, we can infer that N(u) and N(v) are different (N(u) not equal to N(v))
B. Security Model
Sensors are not tamper-resistant. The compromise or capture of a sensor releases all its security information to the attacker. Thereafter, the adversary can start replicating the node, and distribute the clones throughout the network. Note that the cloned nodes own all the legitimate information of the compromised node (e.g. ID, keys, code, etc.). Thus the replicas can easily participate in the network operation in the same way as the legitimate. The cloned nodes are under the control of the adversary, and therefore can launch various internal attacks afterward. For example, a cloned node can easily fabricate a false event report to mislead the decision makers, or keep injecting bogus data to cause network outage. We assume that the base station is sufficiently powerful to defend itself against security threats, while the low-cost sensors can be compromised or physically captured. We also assume that a secure routing protocol is available, such that the message being forwarded can be protected from being altered and can be authenticated.

III. Proposed Model
Aliased Detection Mode for Clone Attack Detection in Sensor
In this section, we present ADM method for detecting clone attacks. The detection scheme consists of two phases: computing a fingerprint for each sensor based on its social network, and then detecting clone attacks afterwards.

A. Fingerprints Generation using Range Values
Before deployment, a superimposed s-disjunct code X is pre-computed and assign range values during the node creation offline. Right after deployment, sensor u broadcasts a message containing u’s codeword to the neighborhood N(N(u)) and listens for the messages sent in the neighborhood N(N(u)). In our consideration, the neighborhood N(u) should satisfy n ≥ s, where n is the number of sensors in N(u), s is the strength of the superimposed code X. After the information collection, sensor u computes the fingerprint for each node v , N(u) and stores the range value.

The proposed method is based on the fingerprint generation. In this paper we are proposing a new enhanced method for clone detection over the internet in which node behavior are calculated according to predefined range value to the sensor. Range values are specified at sensor creation time which incorporates the fingerprint verification. This enhanced method called Alias Detection Mode(ADM), the system specifies the range value in between 0 and 1 which calculates the system behavior based on this value and the fingerprint code which makes the system to detect clone attack. our scheme is the first to provide real-time detection of clone attacks in an effective and efficient way.

Sensor u uses Algorithm to compute the fingerprint for each node v ∈ N(u) and behavior. Algorithm takes the codeword set X(v) as input sensor u has received from its neighborhood. The algorithm starts from an s-subset of X(v) that contains the codewords of the s closest neighbors of sensor v, and expands the subset until any further increment will cause the resulting boolean sum to contain no zero. For the subset resulting from the last increment, compute the boolean sum and select one of the zero elements, the position of which will be the fingerprint of sensor v.

B. Detection of Clone Attacks
For sensor u, its fingerprint is computed from the codewords collected from its neighborhood N(u), sensors are stationary after deployment. A legitimate sensor u belongs to a “fixed” neighborhood, whose social characteristics can be encoded into u’s fingerprint. Therefore, each sensor is required to “sign” with its fingerprint FPu whenever it generates a new message to send to the base station. The message transmission should be u → BS : {IDu, FPu, content}.

Assume X is the superimposed s-disjunct code to generate the social codeword for each sensor, which can be represented by an M × N matrix. Even with M = 100, 000, a fingerprint takes no more than 2 bytes to be included in a message. Hence, our detection algorithm imposes a very slight message overhead for protecting a sensor network against clone attacks.

In our consideration, a cloned node may use an arbitrary fingerprint (e.g. the fingerprint of the original sensor), or compute a new fingerprint that is consistent with its new residency. Hence, detecting clone attacks should be conducted in two aspects which incorporates the sensor behavior with the range values which could not be cloned to the adversary.

C. Detection Probability
we investigate the probability Pdetected that a clone node escapes from being detected successfully. Assume the adversary compromises a sensor u, clones t copies of u (denoted as u1, u2,..., ut), and distribute the clones into the network. To avoid being detected, these t clones must fulfill the following two requirements simultaneously:

Condition I: All the clones (denoted as u1’, u2’... ut’), must use the same fingerprint as the sensor u. Otherwise, the base station will identify the difference among the fingerprints used by these nodes that share the same IDs.

Condition II: Each of the clones denoted as u1, u2,..., ut), must use a fingerprint that is consistent with its current neighborhood. Otherwise, the cloned node will be identified by their neighbors. Thereafter, only when the neighbors of the t clones contribute to the same fingerprint as that of sensor u, our detection algorithm fails to identify[a].

IV. Performance Analysis
In the initialization step of our scheme, each node needs to collect the codewords from its local neighborhood, and computes its fingerprint. Therefore, the fingerprint generation poses O(N) local message transmissions in the network overall. Let numm denote the total number of regular data messages generated in the network during network lifetime, the total message transmission cost in the entire network is O(numm · √N). Unlike other detection schemes against distributed clone attack that have each node send a separate message rather than regular data message, our scheme attaches each regular message with a corresponding fingerprint. According to Algorithm 1[a], the fingerprint is pretty tiny. Its size can be bounded by log2M, where M is the number of rows in the superimposed s-disjunct code used in the network. Let ratio denote the ratio of the fingerprint size versus the regular packet size, and Lpacket denote the bit-length of a regular message. Then, the message overhead of our scheme in the entire network after initialization is O(numm · √N) · Lpacket · ratio. Let d denote the average size of the neighbor set in the network. Since each node u stores the fingerprints of its neighbors, the average memory cost is O(d) + max(M,ω · log2M), where max(M,ω · log2M) denotes the memory usage for u to store its social codeword, ω represents the column weight in the superimposed s-disjunct code.

Note that only simple binary operations are involved in local fingerprint computation and the sensor behavior range value therefore our scheme has extremely low computation overhead.
V. Conclusions and Future Work

In this paper, we present a novel real-time detection scheme against clone attacks. Our ADM is superior in that a high detection accuracy as well as resiliency can be achieved at the cost of a low communication/computation/storage overhead. In addition, real-time clone detection is conducted whenever messages flow in the network, and cloned attackers can be identified in a very efficient and effective way. To our best knowledge, scheme is the first to provide real-time detection against clone attacks. For future work, we are going to explore other kinds of social fingerprints and extend our scheme to other classes of networks.

References


