ISSN 2229-5518

Secure Data Intrusion Resilience in Mobile Unattended WSNs

Soumya Surendran

Abstract – Wireless Sensor Networks (WSNs) are causes a wide range of attacks due to their distributed nature, limited sensor resources, and lack of tamper resistance. A sensor is corrupted, the adversary learns all secrets. Most security measures become ineffective. Recovering secrecy after compromise requires help from a trusted third party In Unattended Wireless Sensor Networks (UWSNs), where the sink visits the network periodically. Unattended Wireless Sensor Networks (UWSNs), where sensors move according to some mobility models. That such a mobility feature could be independent from security (e.g., sensors move to improve area coverage). Here define novel security metrics to evaluate intrusion resilience protocols for sensor networks. And also propose a cooperative protocol that by leveraging sensor mobility allows compromised sensors to recover secure state after compromise

1. INTRODUCTION

Security in WSNs presents several well-known challenges stemming from all kinds of resource constraints of individual sensors. However, the main limitation that complicates sensor security techniques is lack of ubiquitous (inexpensive) tamper-resistant hardware. Lack of secure storage forces sensors to store cryptographic material, such as keys and seeds, in regular memory. Some recent work [1] showed that commodity sensors can be easily compromised, even without physical access [2]. With compromise, the adversary can read the sensor program memory and storage. As a result, no matter which security techniques are in use, sensor compromise reveals all of its secrets to an adversary. From that moment on, any cryptographic protocol ceases to be effective e collected in a large U.S. campus network. Based on the time of corruption, the security state of a given sensor can be partitioned in three epochs: 1) time before corruption; 2) time during corruption; and 3) time following corruption. Nothing can be done about security in epoch 2 as the adversary controls the sensor, while enforcing security in epochs 1 and 3 requires forward and backward secrecy, respectively. Informally, a cryptographic protocol is forward secure if exposure of secret material at a given time does not lead to compromise of secrets for any time preceding compromise. Whereas, a cryptographic protocol is backward secure if compromise of secret material at a given time does not lead to compromise of any secret to be used in future. It is well known that forward secrecy can be easily obtained by periodically evolving a secret (e.g., a key), using a one-way function. Backward secrecy is much more challenging, because knowledge of Kr allows the adversary to compute secrets for future rounds. It would be trivial to obtain backward secrecy if each sensor had a True Random Number Generator (TRNG). Because a TRNG yields information-theoretically independent values, even if the adversary learns many (but not all) TRNG outputs, it cannot compute the missing values, whether they correspond to the past or to the future. In other words, when the adversary compromises the sensor, it cannot learn past secrets; once the adversary leaves the sensor it will not be able to compute future sensor secrets.

In this paper, we investigate collaborative intrusion resilience in Mobile UWSNs (_UWSNs), where unattended sensors migrate within a fixed deployment area and gather environmental data waiting for the sink to approach the network and to collect them. Our ultimate goal is to design techniques that enable sensors to recover secrecy of their cryptographic material (e.g., BLS Signature) after compromise. In particular, we study the impact on collaborative intrusion resilience of sensor mobility models and number of regions controlled by the adversary.To reaches this goal, we first introduce general metrics to assess the effectiveness of intrusion-resilient protocols for _UWSNs and later propose a collaborative distributed protocol that leverages sensor cooperation and locomotion to achieve probabilistic key insulation. Sensors take advantage of mobility and collaboration with peers to regain secrecy after having been compromised by inadvertently wandering into the area under adversarial control. Using both analytical and simulation results, we show that the proposed protocol provides probabilistic key insulation without any trusted third parties or secure hardware and with minimal overhead. Note that the assurance on the probability to regain key secrecy is a system parameter that can be expressed as a tradeoffs between security objectives and sustained overhead.

2. RELATED WORK

Recently, mobile WSNs have begun to attract attention because of the advantages that mobility brings to sensing applications. If sensors move, the network can guarantee optimal area coverage, even if precise sensor deployment is infeasible (e.g., because of hostile or inaccessible conditions of the deployment area) .Also, mobility helps to solve network connectivity problems caused by sensor failures and allows

- *Soumya Surendran is currently pursuing masters degree program in computer science engineering in Anna University, Tamil Nadu, India, PH-+919447708299. E-mail: soumya291@gmail.com*

ISSN 2229-5518

sensors to adapt their sampling power to respond to precise events .Moreover, mobile sensors can extend sensor lifetimes bringing energy to sensors with depleted batteries. Finally, mobility is currently being investigated as a means to detect sensor capture attacks In the last few years, UWSNs have become subject of some attention. The initial work introduced the UWSN scenario, defined the mobile adversary and investigated simple techniques to counter attacks focused on erasing specific data. This was later extended to include the case, where the adversary's goal is to indiscriminately erase all sensor data. Another recent result introduced simple cryptographic techniques to prevent the adversary from recognizing data that it aims to erase. Sensor cooperation to achieve self-healing in static UWSNs. Self-healing in our scenario has been studied in and in presence of a centralized, static adversary, and a mobile adversary, respectively. This paper extends previous results assessing the impact of a distributed adversary on self-healing of mobile UWSNs.

3. SYSTEM ASSUMPTIONS

Deployment area. A spherical surface provides uniform coverage of the deployment area with random mobility models [23]. However, we stress that the shape of the deployment area is not the focus of our work. Our techniques can be applied to _UWSN deployed on any fixed-area surface: Uniform coverage only helps our analysis.

Time. Time is divided in rounds and all sensors' clocks are loosely synchronized, e.g., via [24]. Round length can be arbitrary; we assume that it reflects a single acquisition of data from the environment, i.e., sensors obtain measurements once per round, that is, at round r sensor sj obtains data drj.

Initialization. Before deployment, each sj is initialized with: 1) the sink public key PK; 2) a common crypto-graphic hash function Hð_Þ used as a pseudo-random number generator (PRNG); and 3) a unique secret seed to bootstrap its PRNG. The PRNG is invoked for all random choices made by the sensor and its status is updated at each invocation— status at round r for sensor sj is denoted with Krj.

Sink Visits and Re-initialization. The sink is an itinerant trusted party that visits the network with a certain frequency. Upon each visit, the sink obtains collected measurements from every sensor, erases sensor memory, provides a fresh initial secret seed for the PRNG, and resets the round counter to 1.

Security. Sensor secrets are fundamental to the provision-ing of several security services, such as data confidentiality and authentication. The protocol introduced in this paper allows sensors to regain secret status after compromise and is not concerned with usage of sensor secrets. However, to ease exposition we will focus on a concrete example. That is, we assume that secrets are used to generate padding values to achieve public-key randomized encryption. Although in the recent past public key encryption was shunned by the sensor security community because of its high cost, novel developments make public key encryption feasible on commodity sensors. Further, the reason why we are using public key encryption when symmetric encryption is cheaper in all respects is that using public key allows the sink to seamlessly decrypt anything that sensors encrypt (for it) in any round. Indeed, as discussed below, the security is based on the use of secret padding or randomizers and not on the mere use of public key encryption. In contrast, if we were to use symmetric encryption, it would be quite hard (and in some specific cases even impossible) for the sink to decrypt data.

3.1 ADVERSARIAL MODEL

The UWSN model considered in prior work assumes a mobile adversary that migrates among different subsets of compromised sensors. In our _UWSN setting, sensors are mobile, while the adversary is static. This latter operating hypothesis, other than being worth investigating on its own, is also motivated by the fact that the adversary might not have enough "resources" to move or there might just be no incentive for it to be mobile, i.e., it might as well be stationary and wait for sensors to move to its controlled area. Previous work has shown that the adversarial mobility model has no or very little impact on the network performance in terms of resiliency, when sensor are mobile. Hence, in this paper we focus on the impact on self-healing of a distributed, static adversary .Further, the envisioned adversary differs from other adversarial models considered in most prior WSN security literature. The latter is static in terms of the set of sensors it corrupts, i.e., it compromises k out of n sensor throughout the network lifetime. Our adversary (ADV) is stationary with respect to the portion of the deployment area it controls; but, the set of compromised sensors changes as nodes move in and out of the adversary-controlled area. Another unique feature characterizing

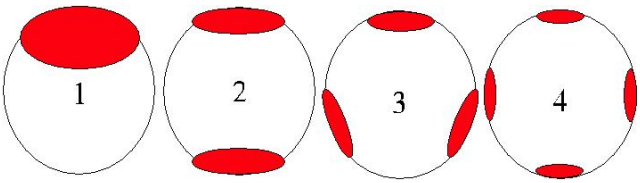

Fig. 1. Adversary layouts. Centralized adversary: (1) ADV ^{1}, and Distributed adversaries: (2) ADV ^{2}, (3) ADV ^{3}, and (4) ADV ^{4}.

Adversarial Degree. ADV is either centralized or distributed. In any case, it has an overall compromising area SADV that is partitioned in one or more equally sized, non-overlapping compromising regions. ADV A denotes an adversary with degree A, that is, distributed on A compromising regions. Each compromising region is a spherical cap with center apa, surface Sa, and range _a, for 1 _ a _ 4.

Fig. 1 shows four considered layouts.3 The centralized

ISSN 2229-5518

adversary (ADV 1) is placed at the north pole of the sphere, whereas, in other cases (A ¼ 2; 3, and 4), the adversary is distributed and placed according to Fig. 1. This paper does not investigate the impact of ADV's position: We placed the compromising regions such that the distance among them is maximal, in order to minimize their mutual influence.

Compromising Power. ADV A compromises all sensors within its range, i.e., sj is compromised at round r if Doðcprj; apaÞ _ _a, for any a _ A. For each compromised sj, the adversary reads all sj's storage/memory and eaves-drops on all incoming and outgoing communications. A compromised sensor is released as soon as it moves away from all the compromising regions, i.e., Doðcprj; apaÞ > _a, for all a _ A.

We assume that the adversary is not a global eaves-dropper and can only eavesdrop on its compromising regions. We stress that ADV does not interfere with sensors' behavior, and can be described as a read-only adversary. A number of techniques allow to discover sensor compromise when the adversary modifies the sensor code [29], [30], [31]. Hence, if the adversary is limited to "read-only" attacks and keeps the sensor code unchanged, there is no way to tell whether that sensor has ever been compromised. This allows ADV to stay undetected and benefit from repeated attacks to the network. Finally, we assume that ADV is aware of the network defense strategy while neither the sensors nor the sink know ADV 's location
.

4. MODEL AND METRICS FOR KEY INSULATION

Based on sensor compromise and the adversary knowledge of its secrets, the set of sensors can be partitioned into three distinct groups at any round:

- Red sensors (R r). A sensor sj is red if it is currently compromised (i.e., cprj 2 Sa) and its secrets are exposed to the adversary.
- Yellow sensors (Yr). A sensor sj is yellow if it is not currently compromised (i.e., cprj 26 Sa), but ADV still knows its secrets for the current round.
- Green sensors Gr. A sensor is green if its current secrets are unknown to ADV . This is because either it has never been compromised or because it has recovered secrecy via the key-insulated protocol.

When it becomes clear from the context, we will use the same notation to denote a set (i.e., Rr; Yr; Gr) and its size. In the following, we refer to green sensors as healthy and to red or yellow sensors as sick. The knowledge of the sensor's secrets allows ADV to perform several attacks, ranging from sensor impersonation to compromising confidentiality of sensed data. Main goals of the adversary are: Either to minimize the number of green sensors, or to keep a specific sensor compromised for as long as possible.

To assess the effectiveness of a generic key-insulated protocol we define two new metrics: Health Ratio (HR) and Healthy Cycle (HC). The former represents the network healthiness as the number of the green sensors, while the latter represents the number of rounds a sensor is green over its lifetime. The natural goal of any intrusion-resilient protocol is to have both HR and HC as close as possible to 1. In particular, HR _ 1 means that secrets of almost all sensors are not exposed, while HC _ 1 means that each sensor is green for most part of its lifetime.

5. THE PROTOCOL

In our protocol, forward secrecy is (predictably) obtained with periodic secret evolution using PRNG Hð_Þ.To obtain backward secrecy, the main idea is for sensors to serve as a source of randomness for their peers. A sensor that resides outside the area controlled by ADV , but whose secrets are exposed (that is, a yellow sensor), can regain security and move to a new secure state (i.e., become green) if it obtains at least one contribution of secure randomness from a peer sensor whose secret state is not exposed (green sensor). As the adversary eavesdrops on red sensors, their received contributions are observable, so they cannot regain secrecy. Our protocol leverages mobility to bring computationally secure randomness to yellow sensors. Since ADV 's location is unknown and sensors cannot distinguish between compromised and noncompromised peers, the protocol is proactively run by all sensors.

At round r, each sj runs Algorithm 3: It moves according to the adopted mobility model (MoveðÞ), and, after reaching its new position, senses data from the environment (ReadðÞ). The latter is encrypted under the sink public key and stored locally. Function PadGenð_Þ uses the sensor's current secret state to generate an encryption padding. At that time, sj broadcasts a random value drawn from its secret state (PadGenð_Þ) and collects randomness sent by its neighbors. Secret state is updated with all received random contributions before moving to the next round.

Algorithm 3. Collaborative Intrusion-Resilient Protocol.

MoveðÞ;
drj ¼ ReadðÞ;
Kjr ¼ PadGenðKrjÞ;
StoreðEPK ðKjr; drj; r; sjÞÞ;
Rrj ¼ ½;&; c ¼ 0;
t ¼ RandGenðKrjÞ;

BroadcastðtÞ;

while ðroundTimerÞ do
Receive trp from sp;
Rrj½c& ¼ trp;

c ¼ c þ 1;

end

6. CONCLUSION

In this paper, we have provided several contributions to the UWSN field. First, we have introduced a new adversary

model that spreads over different areas of the deployment field. Second, we have introduced two novel metrics that, other than being interesting on their own, are of general help when assessing self-healing protocols in autonomous, distributed systems. Third, we have studied, for a wide range of system parameters, how the degree distribution of the adversary affects our self-healing protocol. In particular, the latter shows a great capability to recover from compromising for several deployment settings while incurring a negligible overhead—only local communications are required. Finally, thorough analysis and extensive simulation do support our findings.

.

REFERENCES

[1] C. Hartung, J. Balasalle, and R. Han, "Node Compromise in Sensor Networks: The Need for Secure Systems," Technical Report TR-CU-CS-990-05, Univ. of Colorado at Boulder, 2005.

[2] A. Francillon and C. Castelluccia, "Code Injection Attacks on Harvard-Architecture Devices," Proc. 15th ACM Conf. Computer and Comm. Security (CCS '08), pp. 15-26, 2008.

[3] Nat'l Inst. of Standards and Technology, "FIPS Pub 198: The Keyed-Hash Message Authentication Code," http://csrc.nist. gov/publications/fips/fips198/fips-198a.pdf, 2002.

[4] Y. Dodis, J. Katz, S. Xu, and M. Yung, "Key-Insulated Public Key Cryptosystems," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '02), pp. 65-82, 2002,

[5] M. Bellare and A. Palacio, "Protecting Against Key-Exposure: Strongly Key-Insulated Encryption with Optimal Threshold,"

[6] Applicable Algebra in Eng., Comm. and Computing, vol. 16, no. 6, 379-396, 2006.

[7] R. Di Pietro, L.V. Mancini, C. Soriente, A. Spognardi, and G. Tsudik, "Data Security in Unattended Wireless Sensor Networks," IEEE Trans. Computers, vol. 58, no. 11, pp. 1500-1511, Nov. 2009.

[8] R. Di Pietro, L.V. Mancini, C. Soriente, A. Spognardi, and G. Tsudik, "Catch Me (If You Can): Data Survival in Unattended Sensor Networks," Proc. IEEE Sixth Ann. Int'l Conf. Pervasive Computing and Comm. (PerCom '08), pp. 185-194, 2008,

[9] D. Ma, C. Soriente, and G. Tsudik, "New Adversary and new Threats: Security in Unattended Sensor Networks," IEEE Network, vol. 23, no. 2, pp. 43-48, Mar. 2009.

[10] R. Di Pietro, D. Ma, C. Soriente, and G. Tsudik, "POSH: Proactive Co-Operative Self-Healing in Unattended Wireless Sensor Net-works," Proc. IEEE 27th Symp. Reliable Distributed Systems (SRDS '08), pp. 185-194, 2008,

[11] D. Ma and G. Tsudik, "Dish: Distributed Self-Healing," Proc. 10th Int'l Symp. Stabilization, Safety, and Security of Distributed Systems (SSS '08), pp. 47-62, 2008.

[12] R. Dutta, Y.D. Wu, and S. Mukhopadhyay, "Constant Storage Self-Healing Key Distribution with Revocation in Wireless Sensor Network," Proc. IEEE Int'l Conf. Comm. (ICC '07), pp. 1323-1328, 2007.

[13] V. Naik, A. Arora, S. Bapat, and M.G. Gouda, "Whisper: Local Secret Maintenance in Sensor Networks," IEEE Distributed Systems Online, vol. 4, no. 9, 2003.

[14] K. Dantu, M.H. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G.S. Sukhatme, "Robomote: Enabling Mobility in Sensor Networks,"

[15] Proc. Fourth Int'l Symp. Information Processing in Sensor Networks (IPSN '05), pp. 404-409, 2005.

[16] J. Corte´s, S. Marti´nez, T. Karatas, and F. Bullo, "Coverage Control for Mobile Sensing Networks," Proc. IEEE Int'l Conf. Robotics and Automation (ICRA '02), pp. 1327-1332, 2002.

[17] G. Wang, G. Cao, T.F. La Porta, and W. Zhang, "Sensor Relocation in Mobile Sensor Networks," Proc. IEEE INFOCOM, pp. 2302-2312, 2005.

[18] M.H. Rahimi, H. Shah, G.S. Sukhatme, J.S. Heidemann, and D. Estrin, "Studying the Feasibility of Energy Harvesting in a Mobile Sensor Network," Proc. IEEE Int'l Conf. Robotics and Automation (ICRA '03), pp. 19-24, 2003.

[19] M. Conti, R. Di Pietro, L.V. Mancini, and A. Mei, "Emergent Properties: Detection of the Node-Capture Attack in Mobile Wireless Sensor Networks," Proc. First ACM Conf. Wireless Network Security (WISEC '08), pp. 214-219, 2008.

[20] M. Conti, R. Di Pietro, A. Gabrielli, L.V. Mancini, and A. Mei, "The Quest for Mobility Models to Analyse Security in Mobile Ad Hoc Networks," Proc. Seventh Int'l Conf. Wired/Wireless Internet Comm. (WWIC '09), pp. 85-96, 2009.

[21] M. Conti, R. Di Pietro, L.V. Mancini, and A. Mei, "Mobility and Cooperation to Thwart Node Capture Attacks in MANETs,"

[22] EURASIP J. Wireless Comm. and Networking, vol. 2009, article 8, 2009.

[23] R. Di Pietro, L.V. Mancini, C. Soriente, A. Spognardi, and G. Tsudik, "Playing Hide-and-Seek with a Focused Mobile Adver-sary in Unattended Wireless Sensor Networks," Ad Hoc Networks, vol. 7, no. 8, pp. 1463-1475, 2009.