On the Optimal Allocation of Adversarial Resources

Stylianos Gisdakis
Royal Institute of Technology
Osquldas v. 10
gisdakis@kth.se

Panos Papadimitratos
Royal Institute of Technology
Osquldas v. 10
papadim@kth.se

ABSTRACT

Security is important for mission-critical wireless sensor networks (WSNs). This is especially so because powerful adversaries could compromise and control a significant fraction of the network nodes. A plethora of schemes has been developed to secure wireless sensor networks and resilience to sophisticated attacks has been analyzed. However, the question of how the adversary could deploy her resources to maximally affect the attacked system has remained largely unaddressed. This is the problem this paper is concerned with: Given a number of compromised entities (nodes) and cryptographic keys, how can the adversary devise a close-to-optimal attack tactic? To the best of our knowledge, this is the first investigation of its kind: while the basic adversarial behavior is well-known, the problem of how the adversary can optimally deploy her resources to maximize the attack impact has not been considered for WSNs. We consider an abstract model of the mission-critical WSN and the adversary, and we find that the determination of an optimal attack is computationally hard, thus, we devise an efficient heuristic approach. An intelligent adversarial resource allocation indeed yields disproportional gains for the attacker. Our analysis is the first necessary step to comprehend how to best address vulnerabilities.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: Security and protection

Keywords

Adversary modeling, security analysis, resource allocation

1. INTRODUCTION

Wireless sensor networks cover a broad range of mission-critical applications, notably gathering information of various sorts: from tactical surveillance data and facility monitoring to emergency response support. The nature of these applications, often operating in hostile and adverse environments, makes security indispensable. Resilience to attacks and operation even in the presence of adversaries are highly desired.

There has been a wide gamut of security schemes for wireless sensor networks, for example, managing cryptographic keys [4], securing communication [16], detecting faulty data aggregation [1], protecting data confidentiality [14], detecting sybil attacks [15].

Without dwelling on any specific application or security scheme, the most precarious situations occur when the adversary compromises multiple sensor nodes; that is, controls their operation, and extracts their private/secret cryptographic material. Then, such an adversary can replicate such cryptographic keys and insert her own misbehaving nodes at will. The high-stake mission-critical applications can indeed face such powerful adversaries.

Clearly, the more numerous the cryptographic keys and the nodes under the control of the adversary are, the higher her strength in general is and the more debilitating her attacks are. Worse even, such *internal* adversaries can choose to be *stealthy*, that is, perpetrate their attacks while remaining undetected. Moreover, they could even be *input controlling*, i.e., affects sensing even without taking over a node. Furthermore, the adversaries can choose not only how but also where to hit, i.e., elect the *part* of the sensor network or the *area* where to attack.

Consider a WSN designed to best serve its mission-critical purposes subject to its deployment constraints. In addition, assume that security mechanisms are put in place. But also consider the above-mentioned strong adversary, taking over a significant fraction of the system nodes. *How could an adversary "deploy her forces" in order to have the highest possible impact?* In other words, where and how should the adversary hit, or equivalently how should the adversary allocate her resources in order to distort the most of the data collected by the victim network?

The problem would be trivial if the adversary targeted only a single sensor or a part of the sensor network: she could strive to compromise just enough nodes, that is, all the nodes in the part in question and thus successfully mount the attack. Similarly, the problem would be trivial if the attacker could single-handedly take over the entire or an overwhelming part of the network. On the contrary, the problem is not trivial when the entire or a large part of the network is of interest for an adversary that does not have overwhelming power.

MiSeNet'12, August 26, 2012, Istanbul, Turkey.

In this paper, we are concerned exactly with this question. Our motivation is notably to approach this in an application- and security protocol- agnostic manner. We model the mission critical network as a set of parts where the adversary can attack. Then, we model the interplay between security and adversarial strength through a measure of what is necessary for an attack objective to be achieved in each of those parts. The better the choice of attack points, that is, network parts where the attack is mounted, the higher the impact and thus the "gain" of the adversary. The challenge lies exactly on how to optimize the attack gain, given substantial yet not abundant resources at the hands of the adversary.

A well-understood answer can be highly significant: it can reveal vulnerabilities, in fact, it can reveal the worst case for the system designer, and consequently guide the deployment of defense mechanisms. In this paper, we do not venture to answer this question. Rather, we analyze the tactics of the adversary, exactly to shed light on how vulnerable a mission-critical sensor network can be as a function of the adversarial strength. We see that the problem of identifying an optimal attack, is *computationally hard*. Thus, we develop an efficient heuristic approach to determine a close-to-optimal attack tactic. To the best of our knowledge, this is the first work to consider explicitly the allocation of adversarial resources. We consider two characteristic cases and we find that such an intelligently guided attack can yield disproportional advantages for the adversary. Moreover, we find that stealthy behavior, with the adversary remaining "below the radar", can be also highly enhanced by an intelligent allocation of adversarial resources.

The rest of the paper is organized as follows: In Sec. 2 we present our system and adversary models and we define the problem we address. In Sec. 3 we describe our approach to optimize the adversarial resource utilization and in Sec. 4 we evaluate it through simulations. We then provide brief discussions on related future work in Sec. 6 before we conclude.

2. SYSTEM AND ADVERSARIAL MODEL

2.1 System Model

We model a wireless sensor network (WSN) as a set, S, of n clusters $S = \{C_1, C_2, ..., C_N\}$. Each cluster, C is essentially a subset of the WSN. We do not dwell on the cluster formation, e.g., the communication topology formation; rather we assume that the clusters are formed according to the requirements of the supported application and based on the monitored area and phenomena. The number of benign nodes within a cluster C_i is defined by a function $F_{ben}(C_i) : S \rightarrow \mathbb{N}$. The valuations of all the clusters of the network are encoded as a vector $V = \{V_1, V_2, ..., V_N\}$. These values of are either proportional to the number of benign nodes within the specific cluster or context specific. As an exemplification think of a disaster relief sensor network deployed prior to some disaster. In case an incident occurs, the utility of controlling a cluster with 3 nodes really close to the point of interest is much greater than the utility of a cluster containing 100 nodes that are further away. We term U_{total} to be the utility gained by the adversary by controlling the whole network so that $\sum_{i=1}^{N} V_i = U_{total}$.

We consider primarily data collection as the central operation, and related security mechanisms, such as robust aggregation or other related protocols. Nodes are equipped with a set of cryptographic keys used to ensure the confidentiality, the integrity and the authenticity of the communications among nodes and the sink. Rather than considering individual types of security protocols, we model the resilience through the ability of the adversary to deviate without being detected.

2.2 Adversarial Model

We assume that adversarial resources fall into two categories, physical devices and cryptographic keys. R_{Phy} is the number of sensor nodes the adversary controls, either by having introduced them to the system or by having compromised formerly deployed benign nodes. We assume that $R_{Phy} << n$, not a large fraction of the total number of sensor nodes (recall that we do not want an overwhelming strong adversary). The cryptographic keys the attacker possesses result from benign node compromise; that is, the adversary cannot in principle break otherwise the employed crypto-system. We term this kind of resource as R_{Crypto} where $R_{Phy} \leq R_{Crypto}$ (so that a compromised node can have more than one cryptographic key). In addition we require that one single key cannot be used by more than one node simultaneously. The reason is that we consider intelligent types of attackers that would not trigger sybil detection schemes.

The effectiveness of an adversary depends on her knowledge of the system. In our model, the adversary is aware of the allocation of benign nodes within each cluster (i.e knows $F_{ben}(C_i)$). Function $F_{val} : [0, n] \rightarrow \mathbb{R}^+$ maps cluster C_i to its value. The attacker is aware of F, and as a result can quantify the utility gained by controlling each cluster. It is possible that the WSN user and the adversary assume different values for the same clusters. Nevertheless, for simplicity of presentation we assume that they are the same. Variations of this are points of future work.

We consider data manipulation, so that the view of the data the WSN user gets is not the actual one but the one the adversary wishes for it. This attack can be launched against each and any of the clusters. If the manipulation takes place, arbitrarily or within a level wished by the adversary, then we say the adversary won over the cluster. We term the utility of the adversary as U_{mal}. We consider two generic types of attack to control a cluster and to manipulate the produced measurements:

Local Majority Attack: The adversary controls the majority of the nodes in the cluster. We assume that independently of the attack form, such an adversary can impair or affect any data collection process and evade any misbehavior detection mechanism. Intuitively, however, controlling a majority is not the only way to affect operation. With the help of a smaller fraction of nodes an attacker can still affect data collection. Deviations can be products of false measurements injected by the malicious nodes.

Stealthy Data Attack: To remain undetected in case misbehavior detection mechanisms are in place, adversary controlled nodes report data that differ no more than δ from the measurements reported by benign cluster members.

2.3 Problem Statement

The adversary can choose which clusters to attack, and to what extent. This means, in our model, to choose which clusters she will deploy adversarial nodes (out of the avail-

able R_{Phy}). Then, for each of the nodes allocated, the adversary can choose how many keys to equip each of those nodes with (out of the R_{crypto}). As a result of a deployment, the adversary may manipulate/control the outcome within a cluster. Based on the system and adversarial modeling presented in the two previous paragraphs, a crucial question for the attacker is to identify the optimal allocation of resources that maximizes her U_{mal} given the deployment of the benign nodes of the WSN. This is the exact question that we address in Sec. 3 and 4.

3. ADVERSARIAL TACTICS

First, the adversary decides on the subset M of clusters, which when controlled will maximize U_{mal}. In order to attack a cluster she must allocate at least one physical device into each of the clusters of M. As a result, we have that $|M| \leq R_{Phy}$. Since each cluster has a value, the problem becomes the definition of the subset of clusters that will yield a U_{mal} as close to U_{total} as possible. This is equivalent to the *Subset Sum* $S(V, U_{total})$ that is an *NP-Complete* combinatorial optimization problem [12] which asks for a subset of a set S whose sum is as large as possible, but does not exceed a predefined value termed as *target*.

To calculate the U_{mal} for a given subset M the attacker should define the optimal distribution of R_{Crypto} among the clusters in the subset. If R_{Crypto} allows the attacker to control all of the clusters of a subset M, then the problem is trivial. If not, the problem is equivalent to the *0-1 Knapsack Problem* of combinatorial optimization. This binary version of the generic knapsack problem is an *NP-Hard* problem [7].

The non-polynomial nature of these two problems is an indication of the hardness of selecting an optimal resource allocation strategy. Nevertheless, there exist various heuristics and algorithms that can efficiently approximate the optimal solutions.

3.1 Cluster Selection

Various heuristics have been proposed to find efficient and accurate solutions for the subset sum problem. One popular and robust option are *Genetic Algorithms* (GAs) [11]. These are search heuristics whose main algorithmic structure is termed a *Chromosome*: a candidate solution to the optimization problem. The appropriateness of a chromosome is based on an evaluation function termed the *Fitness Function*. From an original population of chromosomes and with the use of evolutionary techniques, a GA converges towards better (according to the defined fitness function) solutions. The two basic operators of genetic algorithms are *Mutation* and *Cross-Over*. The mutation operator takes a chromosome that encodes a candidate solution and probabilistically mutates one or more of its genes (see Fig. 1). The cross-over operator takes as an input two or more candidate solutions in the form of chromosomes and produces another candidate solution (offspring chromosome) from their combination (see Fig. 2).

3.1.1 Genetic Algorithms for Cluster Selection

To apply a Genetic Algorithm as a heuristic for the Cluster Selection problem we have to specify the form of the chromosomes that will be used. Recall that the adversary is physically constrained to R_{Phy}. This restriction is encoded by chromosomes whose number of genes is equal to R_{Phy}.

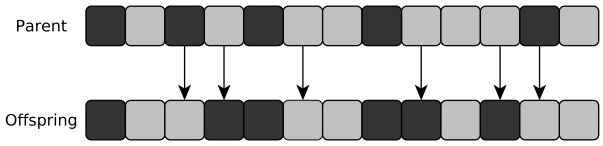

Figure 1: Mutation Genetic Operator.

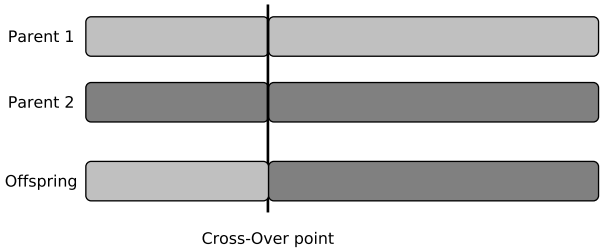

Figure 2: Cross-Over Genetic Operator.

Each gene includes an integer value that represents the index or an identifier of some cluster. For example, consider *chromosome* $\{C_1, C_5, C_8, C_{20}\}$. This candidate solution describes a scenario with an adversary constrained to $R_{Phy} = 4$, attacking clusters C_1, C_5, C_8 and C_{20}. In case this chromosome is selected for mutation, a possible offspring chromosome could be $Chromosome'$ $\{C_1, C_{11}, C_9, C_{20}\}$.

At each evolution step the chromosomes are evaluated based on the value they carry for the attacker. In order to compute this utility, the second combinatorial optimization problem in hand needs to be addressed. A solution to that problem will yield the optimal allocation of the R_{Crypto} resources that the attacker possesses across the clusters defined by the chromosome under evaluation.

Different heuristics could be applicable for the Cluster Selection problem. These alternatives can be investigated in future work. We chose to use GAs for various reasons. For starters, due to their probabilistic nature, they can easily overcome local optima and evolve towards better solutions since they explore candidate solutions from various chromosomes of the population in a parallel manner. In addition, the maximization of the value gained by the adversary can be in fact a process of multi-objective optimization (for example increasing the damage inflected while reducing the cost of the attack). Although this case is not considered in this paper, employing Genetic Algorithms renders possible the exploration of more complex adversarial rationality that cannot be expressed analytically.

3.2 Resource Allocation per Selected Clusters

Let $F_{cost} : [0, Max] \rightarrow \mathbb{N}$ be the function that assigns a cost to each cluster. The cost for launching a majority attack against a cluster C_i is $F_{cost}(C_i) = F_{ben}(C_i) + 1$. In the context of attacks against data aggregation let function $F_{cost}(C_i, \Delta, \delta) \rightarrow \mathbb{N}$ be the amount of malicious nodes required in order to produce a deviation from the average aggregate equal to Δ by reporting values that are a percentage δ of the average value produced by the rest of the nodes in the cluster. Variable δ models the flexibility of malicious nodes in reporting erroneous values. For a chro-

mosome with N genes the problem of optimal allocation of R_{Crypto} is formulated as follows:

$$\text{Maximise:} \quad \sum_{i=1}^{N} x \cdot F_{val}(C_i)$$

$$\text{subject to:} \quad \sum_{i=1}^{N} x \cdot F_{cost}(C_i) \leq R_{Crypto},$$

$$x \in \{0, 1\}$$

This maximization program is in fact the formal definition of the *0-1 Knapsack Problem* mentioned in the beginning of this section. Various algorithms have been proposed to solve it in an efficient manner. One of them is based on *Dynamic Programming* that solves the problem in pseudo-polynomial time. In the case of resource allocation this is translated into a complexity of $O(R_{Phy} \cdot R_{Crypto})$. The output of this algorithm is a vector $X = \{X_1, X_2, ..., X_{R_{Phy}}\}$ where $X_i \in \{0, 1\}$. This vector defines which clusters of the chromosome under evaluation should be selected in order to achieve maximum utility within the resource constraints set by R_{Crypto}.

Recall that our framework uses a GA for cluster selection. In each evolution of the genetic algorithm, chromosomes are evaluated based on the maximum utility they can achieve. After a defined number of iterations (*evolutions* of the algorithm), the GA converges to an optimal (or near optimal in case of large scale networks) resource allocation of both R_{Crypto} and R_{Phy}.

4. ANALYSIS

In this section we provide an evaluation of the attacking tactics based on the proposed model through a series of simulations. Our findings show the gains from the allocation strategy defined by our model to be high even though adversary is significantly constrained.

4.1 Simulation Setup

We implemented our model using the *JGAP* [5] genetic algorithm package. For the dynamic programming part of the model we implemented a basic dynamic programming algorithm. The setup of the experiments included configurations with clusters assigned a number of benign sensor nodes. In every simulation, the attacker was provided with a number of compromised nodes and a number of cryptographic resources. These numbers were defined as significantly lower than the benign physical and cryptographic resources. In Table 1 we present the main simulation parameters along with their respective values.

Table 1: Simulation Parameters and their values

# of Clusters	The number of clusters the WSN is composed of [0,N]
R_{Phy}	The amount of physical adversarial resources [0, # of Clusters]
R_{Crypto}	The amount of cryptographic keys the adversary possesses (as a ratio of the total number of benign nodes) [0,1]
Population	The number of chromosomes after each evolution of the GA [0,∞]
Evolutions	The number of evolutions the GA will perform [0,∞]
Δ	Desired deviation from AVG [0 - AVG]
δ	Flexibility of erroneous reporting (ratio of AVG) [0 - 1]

4.2 Results

In Fig. 3 (a), we can see that the gain U_{mal} of the adversary is close to 30% of U_{total}. In this simulation the adversary has physical presence in at most 20% of the clusters. In addition, the cryptographic keys the adversary possesses are equal to 10% (upper line) and 20% (lower line) of the total number of benign keys. The objective of the adversary is to achieve majority in as many clusters of the network as possible. The results of this simulation illustrate that an intelligent adversary can inflict a significant damage to the network.

In Fig. 3 (b) we examine what happens in case the adversary tries to affect the *AVG* data aggregation function by introducing deviations from the real aggregate value (i.e launch a data stealthy attack). In this scenario we examine three cases for [$\Delta = 0.1$, $\delta = 0.2$] (upper line), [$\Delta = 0.1$, $\delta = 0.5$] (middle line) and [$\Delta = 0.2$, $\delta = 0.2$]. Again, the outcome is that a rather constrained adversary (with physical presence in at most 20 out of 50 clusters ($R_{Phy} \leq 20$) and with cryptographic resources equal to 20% ($R_{Crypto} = 0.2$) of the total number of benign nodes) can use an intelligent allocation strategy that allows her to reach a high U_{mal} (more than 55% of U_{total} when $\Delta = 0.1$ and $\delta = 0.2$).

In Fig. 3 (c) we provide a comparison of the impact (i.e U_{mal}) of stealthy attack against data aggregation compared to local majority attacks. The purpose of this comparison is to illustrate the ease (in terms of resources the adversary needs) of launching an attack against data aggregation. As it can be seen, with the same amount of physical and cryptographic resources, the adversary manages to gain significantly higher utility when launching attacks against aggregation compared to launching majority attacks.

Fig. 3 (d) shows that our model manages to extract 80% of the optimal strategy (this means that the Hamming distance between the generated chromosome and the optimal chromosome was equal to 1) for the case where the total number of clusters is 125. This corresponds to the optimal chromosome being 1 out of $\binom{125}{5} \simeq 3 \cdot 10^{12}$ possible chromosomes. The larger the network is (more clusters), the harder it becomes to extract the optimal strategy (because the optimal strategies will correspond to a decreasing fraction of all the available options). This illustrates the fundamental trade-off between the available computational resources and the optimality of the computed solutions whenever heuristics, such as GAs are used. Nevertheless, by carefully specifying the parameters of GAs we can effectively extract close to optimal solutions.

The research literature on the security of WSN usually assumes that the adversary is rather limited and controls a small fraction of nodes. With our results we manage to show that this kind of constrained adversary can cause a significant damage to the data collection process of a large part of the network. This conclusion is a point that should be taken into consideration by network and protocol designers of mission critical WSN.

5. RELATED WORK

Many mission critical applications have been envisioned for WSN [13, 20, 9]. Due to the importance of such applications there is a plethora of security mechanisms and

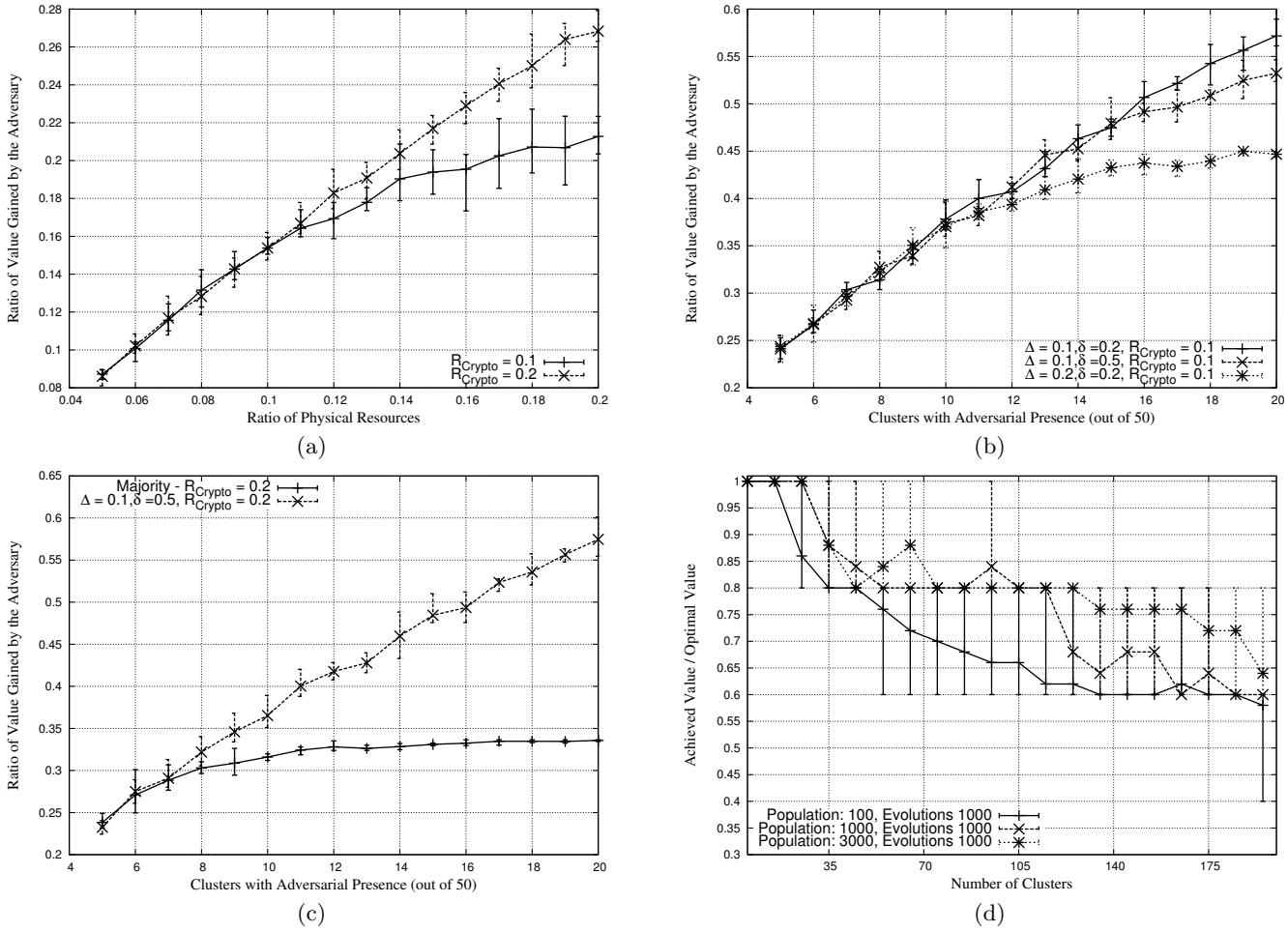

Figure 3: Evaluation of Proposed Model. (a) Malicious Value gained (Local Majority Attack). (b) Malicious Value gained (Attack against Data Aggregation). (c) Comparison of Local Majority Attack and Attack against Data Aggregation scenarios. (d) Accuracy Evaluation

protocols in the research literature. Due to space limitations we cannot cover all of them, instead we briefly review some among the most representative ones. Protocols and frameworks that provide the security services for message confidentiality and integrity along with some efficient key establishment and key management schemes are presented in [16, 6, 4, 2, 14].

For secure sensor aggregation several schemes have been proposed. In [19] a mathematical framework to assess the security of data aggregation functions in various network settings is provided. Additional work on the security of aggregation is presented in [3] where a secure in-network aggregation protocol is defined for arbitrary aggregator topologies. In [21] secure aggregation trees are constructed in order to detect and evict possibly misbehaving nodes. [8] describes a scheme for detecting faults and excluding wrong-doers during the data aggregation process. In [17] the authors provide a framework that detects compromised or faulty nodes in a proactive manner and prohibits them from participating. Finally, the iPDA scheme [10] ensures confidentiality and integrity of data aggregation with the use of disjoint aggregation trees.

Securing the communications and the aggregation process of WSN is critical but orthogonal to the theme of this paper. In this work, we are primarily interested in faulty data contribution by compromised or malicious nodes acting as insiders whose misbehavior would not be detected. To the best of our knowledge such a system wide investigation of adversarial tactics has not been discussed so far.

6. CONCLUSIONS AND FUTURE WORK

We consider resilience in the presence of strong and intelligent adversaries. While the adversary cannot in principle have overwhelming power (e.g., a network-wide majority of compromised nodes and cryptographic keys), it is very well possible it takes under her control a significant fraction of devices. It is important to understand the ramifications of the presence of such an attacker. To do so, it is necessary to see how the attacker can maximize the impact of her exploit. Where and how to attack, in other words, which parts of the network to take over or deploy adversarial nodes in, to harm the network operation the most? Worse even, how to do this while remaining undetected?

This is a computationally hard problem; we develop an efficient and effective heuristic that can guide the adversary, notably the allocation of the adversary's resources. We find that the gains from such an intelligent attack can be high and that our approximative solution is near-optimal. As the scale of the system grows, the attacker's efficacy is reduced in order to maintain relative efficiency. Still, albeit sub-optimal the gain remains high.

For future work, we will leverage our new understanding of the system vulnerability towards alternative adversarial models and especially refined adversarial knowledge. In addition, we intend to move towards more formal definitions of adversarial stealthiness similar to the one presented in [18], in order to grasp more complex adversarial tactics. Moreover, we will investigate further characteristic cases of adversarial deployment to achieve attack objectives and map those to relevant security mechanisms. This could be further extended to lead to specific classes of countermeasures and to a more detailed exploration of the space between an adversary needing "local majority" (or enhanced majority in case of stronger security) and better (stronger) adversarial cases. Overall, we will expand our investigation to make it a two-sided one, to cover both the attacker and the system security designer. The latter is in fact our ultimate target.

7. REFERENCES

[1] J. M. Bohli, P. Papadimitratos, D. Verardi, and D. Westhoff. Resilient data aggregation for unattended wsns. In *6th International IEEE Workshop on Practical Issues in Building Sensor Network Applications (IEEE SenseApp 2011), in conjunction with the 36th IEEE LCN*. IEEE, 2011.

[2] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, SP '03, Washington, DC, USA, 2003. IEEE Computer Society.

[3] H. Chan, A. Perrig, and D. Song. Secure hierarchical in-network aggregation in sensor networks. In *Proceedings of the 13th ACM conference on Computer and communications security*, CCS '06, New York, NY, USA, 2006. ACM.

[4] L. Eschenauer and V. D. Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on Computer and communications security*, CCS '02, New York, NY, USA, 2002. ACM.

[5] K. et al. Meffert. Jgap-java genetic algorithm package. `http://www.jgap.sf.net`.

[6] A. C. Ferreira, M. A. Vilaça, L. B. Oliveira, E. Habib, H. C. Wong, and A. A. Loureiro. On the security of cluster-based communication protocols for wireless sensor networks. In *In 4th IEEE International Conference on Networking (ICN'05), volume 3420 of Lecture Notes in Computer Science*. Springer, 2005.

[7] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[8] P. Haghani, P. Papadimitratos, M. Poturalski, K. Aberer, and J.-P. Hubaux. Efficient and Robust Secure Aggregation for Sensor Networks. In *Proceedings of the Third IEEE ICNP Workshop on Secure Network Protocols (NPSec)*, Beijing, China, October 2007.

[9] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, and B. Krogh. Vigilnet: An integrated sensor network system for energy-efficient surveillance. *ACM Trans. Sen. Netw.*, 2(1):1–38, Feb. 2006.

[10] W. He, H. Nguyen, X. Liuy, K. Nahrstedt, and T. Abdelzaher. iPda: An integrity-protecting private data aggregation scheme for wireless sensor networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, nov. 2008.

[11] J. H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA, 1992.

[12] E. Horowitz and S. Sahni. Computing partitions with applications to the knapsack problem. *J. ACM*, 1974.

[13] K. Lorincz, D. J. Malan, T. R. F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton. Sensor networks for emergency response: Challenges and opportunities. *IEEE Pervasive Computing*, 3(4), Oct. 2004.

[14] J. Luo, P. Papadimitratos, and J.-P. Hubaux. GossiCrypt: Wireless Sensor Network Data Confidentiality Against Parasitic Adversaries. In *SECON*, San Francisco, CA, USA, June 2008.

[15] B. Parno, A. Perrig, and V. Gligor. Distributed detection of node replication attacks in sensor networks. In *Proceedings of the 2005 IEEE Symposium on Security and Privacy*, Washington, DC, USA, 2005. IEEE Computer Society.

[16] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler. Spins: security protocols for sensor networks. *Wirel. Netw.*, 8, Sept. 2002.

[17] K. Ren, W. Lou, and P. J. Moran. A proactive data security framework for mission-critical sensor networks. In *Proceedings of the 2006 IEEE conference on Military communications*, MILCOM'06, Piscataway, NJ, USA, 2006. IEEE Press.

[18] D. Turgut, B. Turgut, and L. Boloni. Stealthy dissemination in intruder tracking sensor networks. In *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*, pages 22 –29, oct. 2009.

[19] D. Wagner. Resilient aggregation in sensor networks. In *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, SASN '04, New York, NY, USA, 2004. ACM.

[20] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, and J. Lees. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10(2), Mar. 2006.

[21] K. Wu, D. Dreef, B. Sun, and Y. Xiao. Secure data aggregation without persistent cryptographic operations in wireless sensor networks. *Ad Hoc Networks*, 5(1), 2007.