Intrusion Detetction Systems and Wireless Sensor Netowrks Lifetime Degradation

Krishna Doddapaneni, Enver Ever, Orhan Gemikonakli, Leonardo Mostarda
School of Engineering & Information Sciences
Middlesex University
London, UK
Email:{k.doddapaneni, e.ever, o.gemikonakli, l.mostarda@mdx.ac.uk}@mdx.ac.uk

Alfredo Navarra
Dipartimento Di Informatica
University of Perugia
Perugia , Italy
Email:{navarra@dmi.unipg.it}

***Abstract*—**
A Wireless Sensor Network (WSN) consists of spatially distributed autonomous sensors that monitor environmental data such as temperature, humidity, light, speed and sound. WSNs poses new security challenges because of their unattended nature and limited resources. Although prevention measures such as encryption and firewalls have been successfully applied, the attacker can physically access the node and modify it. Intrusion Detection Systems (IDSs) are a second line of defence that can be used to mitigate this problem. Building IDSs for WSNs is a new challenge because of the limited resources of the WSN nodes. IDS solutions for sensor networks should try to minimise the use of battery of the sensor nodes in order prolong the network lifetime. In this paper we analyse different solutions that have been proposed for intrusion detection in wireless sensor networks. More specifically we analyse the impact of popular intrusion detection systems on the life time of the WSNs. Our study is quite general since we consider IDSs that are distributed on the sensor nodes and continuously monitor the networks for evidence of attacks. We also consider IDSs that are event triggered, which means that they require agreement between nodes when a suspicious activity is detected. The agreement is used to detect the attack and isolate the attacker. We analyse the effects of IDSs on battery life . The results show that, popular oral message algorithm of Byzantine generals problem should be considered for small scale WSNs because of the overhead introduced in terms of messages exchanged for decision. We conclude our paper with properties and recommendations for IDSs working for WSNs and some future works.

***Index Terms*—Wireless Sensor Networks; Intrusion Detection Systems; Energy consumption.**

I. Introduction

A Wireless Sensor Network (WSN) consists of spatially distributed autonomous sensors that monitor environmental conditions in order to accomplish a task. Today WSNs are successfully applied for safety critical applications such as health care [1], fire alarm systems [2], home automation [3] and battlefield [4]. In these applications, security is one of the main concerns since lives and livelihoods are depending on the WSN.

Wireless sensor networks poses new security challenges because of their unattended nature and limited resources. Although prevention measures such as encryption [5] and firewalls [6] can be used, the attacker can physically access the WSN and tamper with the nodes in order to subvert the correct WSN behaviour. Intrusion Detection Systems (IDSs) can be used to mitigate the problem. They are a second line of defence that analyses the observable behaviours of a system in order to recognise malicious behaviours.

There are two main types of intrusion detection techniques: *misuse* and *anomaly*. Misuse detection systems [7] are explicitly programmed to recognise well-known attacks. These systems recognise intrusions by matching the pattern of observed data with the set of predefined (intrusion) signatures. They can perform focused analysis thus having a low false alarm rate. However, they cannot detect unknown types of attacks. Anomaly detection systems assume that an attack will cause deviation from normal behaviours, thus detection can be done by comparing actual activities with known correct behaviours. Different approaches have been used to model normal behaviours: statistics-based [8], rule-based [9] and formal specification [10]. The advantage of this kind of systems is the ability of detecting unknown attacks. However, it is not easy to define what is a normal behaviour and set up anomaly thresholds in order to have a good detection efficiency and a low positive rate.

Intrusion detection in WSN is a particularly challenging task because of the limited resources of the nodes. WSNs can operate in two different modes called as continuous periodic sensing and transmission or event-triggered sensing. The decision on which mode of operation to use is highly dependant on the application. For WSNs, while the IDS enhances security, it can shorten the lifetime of the WSN since the IDS system may require to run in promiscuous mode [11], [12]. More precisely in promiscuous mode, each IDS can continuously eavesdrop the radio in order to check the correct behaviour all other nodes. This solution not only makes impossible to optimise the duty cycle (nodes can never sleep) but also requires the nodes to be in the same range. However promiscuous mode is not really suitable for applications with event-triggered sensing.

In this paper we study how different intrusion detection solutions affect the lifetime of the wireless sensor networks. More specifically we compare IDSs that continuously monitor the network and IDSs that use some kind of agreement in order to discover the attackers and isolate them. The agreement we consider is based on the Byzantine Generals solution introduced by Lamport [13]. We conclude with some remarks and properties that intrusion detection systems for wireless sensor networks should have.

The rest of the paper is organised as follows: Section II describe the related work; Section III provides the system model and the attack model; Section IV describes the simulation scenario, and tool employed; Section V describes results; Section VII summarises the paper.

II. Related Work

Distributed systems are subject to a variety of failures and attacks. A survey study on intrusion detection systems is presented by Mishra et al. in [14]. They identify security vulnerabilities in mobile ad-hoc networks and propose intrusion detection schemes. Sun et al. [15] presents a survey of intrusion detection techniques in mobile ad-hoc and wireless sensor networks. They also present existing solutions for secure localization and secure aggregation of data. Intrusion detection systems are predominantly classified as signature-based or anomaly-based. In the first approach, signatures containing typical attack characteristics or defined patterns are used to detect the attacks. On the other hand, anomaly-based detectors attempt to detect any type of deviations from the predefined profile of normal network behaviour. Signature based techniques are not capable of detecting new attacks; whereas anomaly based techniques can possibly detect new attacks. However, this advantage comes at the cost of possible false alarms, thus depleting the network life time.

The Byzantine perOblem, and appropriate solution approaches are popular especially for ad-hoc wireless networks. The problem of secure network communications in the presence of Byzantine problems has been extensively studied in [16][17][18]. Lamport et al. introduces the concept of Byzantine Generals Problem in [13] and this ia further developed by Dolev in [19]. Byzantine generals problem describes a problem where one Commander and $n-1$ lieutenants communicate with each other. It is an abstraction of the problem of reaching an agreement in a system where the nodes may exhibit arbitrary behaviour. A Byzantine failure is defined as an arbitrary fault arising during the execution of an algorithm in a fault tolerant distributed computing system. Although a lot of emphasis is on secure networking for ad-hoc and sensor networks, the research work is still limited in distributed detection and data fusion in the presence of Byzantine problems [20]. In [21], Kosut et al. considered information theoretic investigation of data fusion in the presence of Byzantine problems. However, the authors were mainly interested in retrieving the data at the fusion centre and not in the detection performance.

Byzantine failure detectors provide an elegant abstraction for solving security problems. Without using cryptographic mechanisms, Byzantine problems can be tolerated by sending correct messages that outnumber the potential false messages [22][23][24][25].

Lamport et al. [13] have introduced an algorithm than can tolerate Byzantine failures. The application implements two Byzantine agreement protocols. Oral Message and Signed Message Algorithms.

Oral Message Algorithm: To cope with m traitors the authors proposed a solution that works for 3m+1 or more lieutenants. The algorithm works in rounds where messages are exchanged between the lieutenants in each round.

Signed Messages Algorithm: Every lieutenant sends an unforgeable signed message, preventing a traitor lieutenant from sending a value other than what he receives. The number of exchanged messages is minimized since only unforgeable messages are sent. Byzantine fault tolerance techniques such as the state machine replication which can tolerate a bounded number of Byzantine faults can be used to protect the systems [26]. Security issues in sensor networks are similar to ad-hoc networks but due to the energy restrictions of sensor networks, the defence mechanisms developed for ad-hoc cannot be directly applicable for sensor networks. Many ad-hoc network security mechanisms have been proposed for authentication and secured routing protocols in the literature based on the public key cryptography [27][28][29] [30][31][32] [33][34]. Fewer secure routing protocols have been proposed based on the symmetric key cryptography [18][35][36][37]. Cryptography is very expensive for sensor nodes [38]. The Byzantine Generals problem under various hypotheses can be used to implement reliable computer systems. However, these solutions are inherently very expensive in terms of both number of messages required and also the amount of time, especially when the fraction of faulty nodes is high.

A popular intrusion detection technique in WSNs is the *watchdog approach* [**?**] . Each packet transmitted in the network is not only received by the sender and the receiver, but also from a set of neighbouring nodes within the senders radio range. Nodes use this information in order to detect anomalous behaviour. In other words in a watchdog approach nodes control with each other. This solution not only makes impossible to optimise the duty cycle (nodes can never sleep) but also requires the nodes to be in the same range. Furthermore promiscuous mode is not really suitable for applications with event-triggered sensing.

III. System model and attack model

We assume a WSN is composed of a set of nodes communicating by means of send and receive primitives. Nodes can be of different types such as temperature, smoke and sprinkler. Each type of sensor can have several instances.

We assume each node has a unique address. We assume the implementation of a transport layer protocol that allows the end-to-end communication between node. We also consider unreliable links and unpredictable delays for the wireless links. We also assume each node has a public and private key pair that can be used to sign messages.

The attacker can physically compromise one or more nodes. Once the attacker has compromised a node it can install any code and it can use node's credentials to send signed messages. A compromised node does not follow the protocol and can send malicious messages in oder to subvert the correct system behaviour. For instance a malicious node could not allow the detection of a fire or it could enable the water flow of the Sprinklers when no fire is present. We assume there is always communication between two honest nodes. We assume there is secure communication path between an honest node and the the base station. This is used to deliver alert messages. We assume the attacker does not compromise all the sensor nodes but majority of the nodes are honest that is they follow the correct protocol.

IV. Simulation

In this study, Castalia WSN simulator is used together with OMNET simulation package. Castalia is ideal for WSNs for initial testing of protocols and/or algorithms with a realistic node behaviour, wireless channel and radio models. It is highly tunable, can simulate a wide range of platforms, and it is used to evaluate different platform characteristics. Castalia features an accurate radio model based on the work of the authors in [39]. It also features physical process model, considering clock drift, sensor energy consumption, CPU energy consumption, sensor bias etc. Unpredictability of the wireless channel, energy spent in transmission/receiving packets, performance degradation experienced by duty cycles, and collisions are usually overlooked by simple simulators. However these details are well established in Castalia [40]. All main components that affects the energy consumption of sensor nodes are considered that are the micro-processor, the sensor module, wireless transmitter/receiver and the path loss.

The case study considered is a home monitoring system of building environment which is used for WSNs quite often. Please note that, the case study chosen is a typical example of a WSN system where the sensors are event trigered, or the time between observations is relatively long (in other words the number of packets exchanged between the base station and the nodes is spread in a long time period). For these kind of applications, although the number of messages exchanged for establishment of communication is not a real burden, in case of external attacks, the use of IDSs can be the main cause of energy loss and processing delay.

Home monitoring systems include emergency control systems (e.g. fire alarms). The fire alarm system is composed of different temperature sensors and smoke detectors that are distributed uniformly inside the building. There are also sprinkler actuators used to enable the water flow in case of fire. When a temperature sensor reads a value that exceeds a specified threshold; it sends an alert message to the smoke detector. The smoke detector receives the alert and checks for smoke. An alarm is raised when the smoke is detected. In this case the smoke sensor also activates all the sprinklers. We evaluate the life time of the fire alarm system when intrusion detection facilities are introduced. We run the following experiments:

- a fire alarm system without any IDS facility
- a fire alarm system in which an agreement based on Byzantine protocol is used to detect and isolate the attacker
- a fire allarm system in which the IDS is distributed on each node and runs in promiscuous mode that the IDS component continuously eavesdrop the network

The following simulation parameters are used: CC2420 radio defined by the Texas instruments is used. The packet rate is kept at 250 kbps, the radio bandwidth is 20 MHz and the simulation is run for 9000 sec. The nodes are deployed uniformly across the area as shown in figure 1.

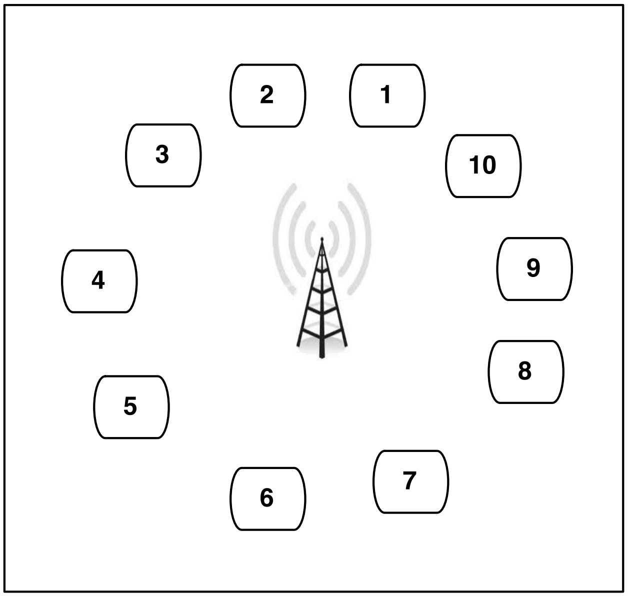

Fig. 1. Case study considered

V. Numerical Results and Evaluation

In this section numerical results are presented for the fire alarm system. Results are obtained to check the cost of two different intrusion detection systems. The first system is watchdog based approach in which the IDS is distributed in each node and the radio is assumed to be always on so that the IDS can capture and analyse all the messages. The second IDS uses the solution of Byzantine generals problem using oral messages solution as offered in [13] to detect and isolate the attacker. In order to demonstrate the cost of the considered IDSs more evidently, numerical results are provided for a system without IDSs as well.

Castalia simulation package is used which considers the stochastic processes; job arrivals and departures, in an event triggered fashion. Although the OMNET package is event triggered, the simulation time was quite extensive especially for the scenarios with more than 70 nodes. Results presented in figure 2 are given for up to 100 nodes. We assume that 60% and 30% of the nodes are sensors of temperature and smoke, respectively, while 10% of the nodes are sprinkler actuators. For all the scenarios the simulation is run for 9000 seconds.

The lowest line of Figure 2 shows the energy consumed by a single sensor of temperature when no IDS is introduced. More specifically the temperature node is sensing the temperature continuously and sending a reading to the smoke sensor every 30 seconds. The energy consumed is 30.82, 39.462 and 70.397 joules for a wireless sensor network composed of 10, 50 and 100 nodes, respectively.

The middle line of Figure 2 shows the energy consumed by a temperature node that runs for 9000sec and performs exactly one instance of the Byzantine agreement. We assume the temperature node is sensing the temperature and after it

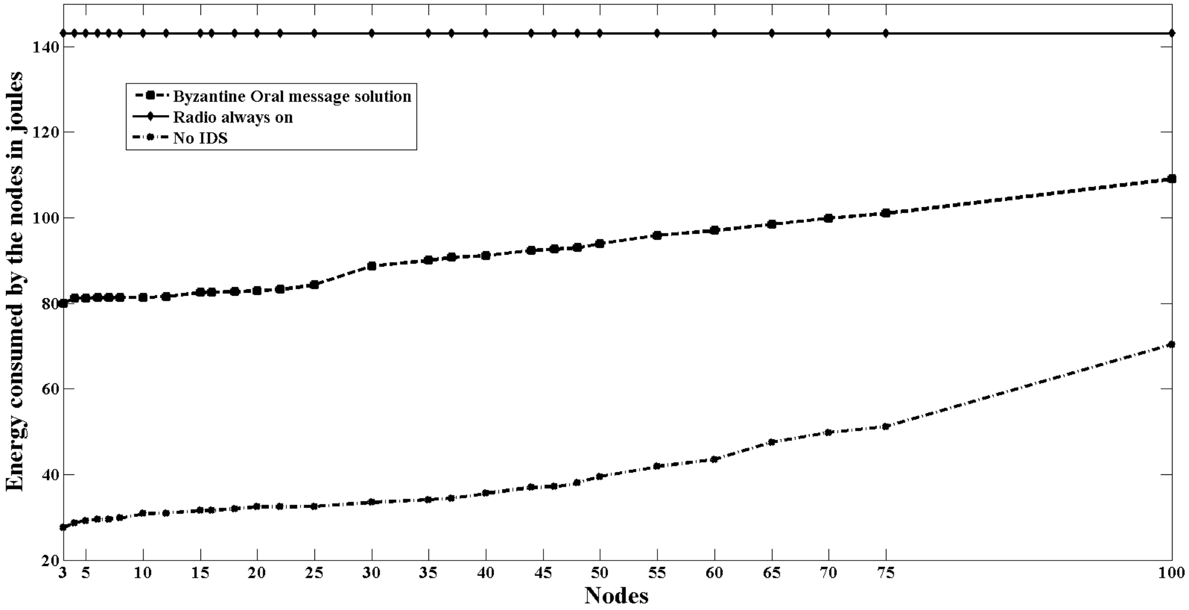

Fig. 2. Energy consumed for the IDSs as a function of number of nodes

runs the Byzantine agreement. This is run with all remaining temperature nodes in order to understand whether or not the temperature is indeed high or some of the nodes is trying to attack the system. After running the Byzantine agreement the temperature node continues to sense temperature and sends a message to all the smoke sensors every 30 seconds. The energy consumed is 81.389, 93.92 and 109.004 joules for a wireless sensor network of 10, 50 and 100 nodes , respectively.

The highest line of Figure 2 shows the energy consumed by a node that has the radio receiver always on that is 143.125 joules. This is the case in which the node is hosting an IDS component that is continuusly eavesdropping the network.

The results presented clearly show that the energy consumed because of the Byzantine protocol increases as the number of nodes in the system increases. Although, it was expected to have rapid increases for the energy consumed is almost linear. The main reason of having this behaviour is that there is an upper-bound for the maximum energy that can be consumed by each node. The maximum energy that can be consumed is equal to the energy consumed when the radio is always on. Since for a Byzantine system with n nodes the number of messages delivered can be computed as $(n-1)$ for the first step $(n-1)(n-2)$ for the second step and $(n-1)(n-2)\dots(n-k)$ for step k, for systems with large numbers of nodes, the radio is always kept on in order to deal with incoming and outgoing packets.

The delay introduced by the intrusion detection system should also be considered. As we discussed in [41] the average time in order to verify a rule for intrusion detection is about about 2.8ms. It means that when the IDS is running in promiscuous mode and the sensor node is not overloaded with messages from its neighbours the node should be able to perform all sensing and detection activities. When the byzantine agreement is considered the delay before the node restarts its reading and sensing activities can be quite long. For instance if we consider only 10 nodes and an average round trip time of 40 ms a node can take around 3sec.

VI. Discussion and remarks

A novel intrusion detection system should not eavesdrop the network all the time nor should run a Byzantine agreement involving too many nodes. The former solution would deplete the energy of the node quite quickly while the latter would require each node to run the agreement for a long time suspending the sensing and reading activities.

An intrusion detection system for wireless sensor networks should have the following characteristics:

- it should not run in promiscuous mode all the time;
- it should locally check the information on each node and start interaction with other nodes when a suspicious activity is detected;
- the detection of an attack (after a suspicious activity is detected) should require agreement between few nodes without involving all the nodes of the WSN

We believe most of the time the attack is localised on some part of the network. There should be an agreement to allow only those nodes should in the detection of the attacks.

Byzantine solution based on signature can also be explored. Although this can reduce the number of messages, since computing the signature for a sensor node can be quite energy consuming.

VII. Conclusion and future work

This paper considers the cost of intrusion detection systems in the context of WSNs. Although there are a number of security measures offered, and implemented for WSNs and/or wireless ad-hoc networks, the discussions of performance of the implemented algorithm should go beyond how secure the network becomes, what kind of attacks can be detected etc. Especially in case of WSNs, the matter becomes very delicate since the energy consumption can have severe effects on the network lifetime while the network is being protected.

In this paper we have evaluated the impact on the energy consumption when intrusion detection systems are employed together with wireless sensor networks. We have considered a fire alarm system case study and we have evaluated the energy consumption under the following three settings: (i) system is assumed to use a finite state machine in each node, therefore the radio is assumed to be always on to be able to capture all the messages in an attempt to update the FSM; (ii) a byzantine agreement is run in order to detect and isolate the attacker; and (iii) there is no intrusion detection system installed. The results show that, the byzantine agreement can be used in case of small scale networks. If the checking is not going to take place very often, in other words if the byzantine protocol does not run very often, the overhead introduced can be tolerated. On the other hand if the number of nodes is high, the energy consumption approaches to its upper bound, where the radio is always on (the first setting).

References

[1] J. Ko, C. Lu, M. Srivastava, J. Stankovic, A. Terzis, and M. Welsh, "Wireless sensor networks for healthcare," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1947 –1960, nov. 2010.

[2] S. Bhattacharjee, P. Roy, S. Ghosh, S. Misra, and M. S. Obaidat, "Wireless sensor network-based fire detection, alarming, monitoring and prevention system for bord-and-pillar coal mines," *Journal of Systems and Software*, vol. 85, no. 3, pp. 571 – 581, 2012, ¡ce:title¿Novel approaches in the design and implementation of systems/software architecture¡/ce:title¿. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0164121211002378

[3] D. Tudose, A. Voinescu, M. Petrareanu, A. Bucur, D. Loghin, A. Bostan, and N. Tapus, "Home automation design using 6lowpan wireless sensor networks," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, june 2011, pp. 1 –6.

[4] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, T. Bessell, M. Rutten, and S. Jha, "Wireless sensor networks for battlefield surveillance," in *in Proc. of the Land Warfare Conference*, 2006.

[5] C. Jia, Y. Liao, and K. Chen, "Secure encryption in wireless sensor network," in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, oct. 2008, pp. 1 –4.

[6] J. Ma, P. Yi, Y. Zhong, and S. Zhang, "Sfirewall: A firewall in wireless sensor networks," in *Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006.International Conference on*, sept. 2006, pp. 1 –4.

[7] S. T.Eckmann, G. Vigna, and R. A. Kemmer, "Statl: An attack language for state-based intrusion detection," *Journal of Computer Security*, vol. 10, pp. 71–104, 2002.

[8] H. S. Javitz and A. Valdes, "The nides statistical component description and justification," *Technical report - Columbia University*, March 1994.

[9] H. Vaccaro and G. Liepins, "Detection of anomalous computer session activity," *In proc. of the 1989 Synopsium on Security and privacy*, no. 1-3, pp. 280–289, May 1989.

[10] M. Stillerman, C. Marceau, and M. Stillman, "Intrusion detection for distributed applications," *Communications of the ACM*, July 1999.

[11] H. Chen, P. Han, X. Zhou, and C. Gao, "Lightweight anomaly intrusion detection in wireless sensor networks," in *Intelligence and Security Informatics*, ser. Lecture Notes in Computer Science, C. Yang, D. Zeng, M. Chau, K. Chang, Q. Yang, X. Cheng, J. Wang, F.-Y. Wang, and H. Chen, Eds. Springer Berlin / Heidelberg, 2007, vol. 4430, pp. 105–116, 10.1007/978-3-540-71549-8-9.

[12] A. Filipovic and A. Datta, "Building blocks of energy and cost efficient wireless sensor networks," in *Wireless Sensor Networks*, ser. Lecture Notes in Computer Science, H. Karl, A. Wolisz, and A. Willig, Eds. Springer Berlin / Heidelberg, 2004, vol. 2920, pp. 218–233.

[13] L. Lamport, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, pp. 382–401, 1982.

[14] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion detection in wireless ad hoc networks," *Wireless Communications, IEEE*, vol. 11, no. 1, pp. 48 – 60, feb 2004.

[15] B. Sun, L. Osborne, Y. Xiao, and S. Guizani, "Intrusion detection techniques in mobile ad hoc and wireless sensor networks," *Wireless Communications, IEEE*, vol. 14, no. 5, pp. 56 –63, october 2007.

[16] D. Dolev, C. Dwork, O. Waarts, and M. Yung, "Perfectly secure message transmission," *J. ACM*, vol. 40, no. 1, pp. 17–47, Jan. 1993. [Online]. Available: http://doi.acm.org/10.1145/138027.138036

[17] D. Malki and M. Reiter, "A high-throughput secure reliable multicast protocol," in *Computer Security Foundations Workshop, 1996. Proceedings., 9th IEEE*, jun 1996, pp. 9 –17.

[18] P. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks," *MOBILE COMPUTING AND COMMUNICATIONS REVIEW*, vol. 1, no. 2, pp. 27–31, 2002.

[19] D. Dolev, "The byzantine generals strike again," Stanford, CA, USA, Tech. Rep., 1981.

[20] E. Shi and A. Perrig, "Designing secure sensor networks," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 38 – 43, dec. 2004.

[21] O. Kosut and L. Tong, "Distributed source coding in the presence of byzantine sensors," *Information Theory, IEEE Transactions on*, vol. 54, no. 6, pp. 2550 –2565, june 2008.

[22] V. Bhandari and N. H. Vaidya, "On reliable broadcast in a radio network," in *Proceedings of the twenty-fourth annual ACM symposium on Principles of distributed computing*, ser. PODC '05. New York, NY, USA: ACM, 2005, pp. 138–147. [Online]. Available: http://doi.acm.org/10.1145/1073814.1073841

[23] C. yuen Koo, "Broadcast in radio networks tolerating byzantine adversarial behavior," in *In PODC 04: Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*. ACM Press, 2004, pp. 275–282.

[24] C.-Y. Koo, V. Bhandari, J. Katz, and N. H. Vaidya, "Reliable broadcast in radio networks: the bounded collision case," in *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, ser. PODC '06. New York, NY, USA: ACM, 2006, pp. 258–264. [Online]. Available: http://doi.acm.org/10.1145/1146381.1146420

[25] A. Pelc and D. Peleg, "Feasibility and complexity of broadcasting with random transmission failures," *Theor. Comput. Sci.*, vol. 370, no. 1-3, pp. 279–292, Feb. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.tcs.2006.10.031

[26] M. Castro and B. Loskov, "Practical byzantine fault tolerance."

[27] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE NETWORK MAGAZINE*, vol. 13, pp. 24–30, 1999.

[28] J.-P. Hubaux, L. Buttyán, and S. Capkun, "The quest for security in mobile ad hoc networks," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '01. New York, NY, USA: ACM, 2001, pp. 146–155. [Online]. Available: http://doi.acm.org/10.1145/501436.501437

[29] J. Kong, Z. Petros, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks," in *Network Protocols, 2001. Ninth International Conference on*, nov. 2001, pp. 251 –260.

[30] M. G. Zapata, "Secure ad hoc on-demand distance vector routing," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 6, no. 3, pp. 106–107, Jun. 2002. [Online]. Available: http://doi.acm.org/10.1145/581291.581312

[31] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang, "Self-securing ad hoc wireless networks," in *Computers and Communications, 2002. Proceedings. ISCC 2002. Seventh International Symposium on*, july 2002, pp. 567 – 574.

[32] J. Binkley and W. Trost, "Authenticated ad hoc routing at the link layer for mobile systems," *Wirel. Netw.*, vol. 7, no. 2, pp. 139–145, Mar. 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1016633521987

[33] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Belding-Royer, "A secure routing protocol for ad hoc networks," in *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*, nov. 2002, pp. 78 – 87.

[34] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, "Adaptive security for multi-layer ad-hoc networks," in *SPECIAL ISSUE OF WIRELESS COMMUNICATIONS AND MOBILE COMPUTING*. Wiley Interscience Press, 2002, pp. 533–547.

[35] S. Basagni, K. Herrin, D. Bruschi, and E. Rosti, "Secure pebblenets," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '01. New York, NY, USA: ACM, 2001, pp. 156–163. [Online]. Available: http://doi.acm.org/10.1145/501436.501438

[36] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: a secure on-demand routing protocol for ad hoc networks," *Wirel. Netw.*, vol. 11, no. 1-2, pp. 21–38, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1007/s11276-004-4744-y

[37] Y.-C. Hu, D. Johnson, and A. Perrig, "Sead: secure efficient distance vector routing for mobile wireless ad hoc networks," in *Mobile Computing Systems and Applications, 2002. Proceedings Fourth IEEE Workshop on*, 2002, pp. 3 – 13.

[38] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," in *In First IEEE International Workshop on Sensor Network Protocols and Applications*, 2002, pp. 113–127.

[39] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, oct. 2004, pp. 517 – 526.

[40] (2011, Dec.) Castalia. [Online]. Available: http://castalia.npc.nicta.com.au

[41] G. Russello, L. Mostarda, and N. Dulay, "A policy-based publish/subscribe middleware for sense-and-react applications," *Journal of Systems and Software*, vol. 84, no. 4, pp. 638–654, 2011.