Paper

Intrusion Detection in Heterogeneous Networks of Resource-Limited Things

Adam Kozakiewicz, Krzysztof Lasota, and Michał Marks

Research and Academic Computer Network (NASK), Warsaw, Poland

Abstract—The paper discusses the threats to networks of resource-limited things such as wireless sensors and the different mechanisms used to deal with them. A novel approach to threat detection is proposed. MOTHON is a movement-assisted threat detection system using mobility to enhance a global threat assessment and provide a separate physical secure channel to deliver collected information.

Keywords—client honeypot, Internet of Things, intrusion detection, wireless sensor network.

1. Introduction

The terms like computer or network are becoming less clear as the technology advances. No more than ten years ago, most of the nodes in the Internet were stationary computers with a wired connection. On the server side of the network this is still an accurate depiction at least of the physical setup, although admittedly more and more inaccurate on the logical side, as virtualization advances and the additional cloud layer isolates the servers from their hardware.

On the client side, the network changed completely. Most of new devices are wireless. Also, the name "device" is quite appropriate, as more and more of them do not look like traditional computers (even if that's what they essentially are). The trend is not only directed at mobility of computing, as in case of laptops, smartphones, tablets, etc., but also towards expanding the computational abilities of other things, leading to ideas such as smart home or smart city.

The side effect of this approach is that the network is becoming full of devices with at least one of the following limitations: battery power, meaning that energy conservation becomes crucial factor, or limited computing power, due to lowering costs, lowering energy consumption or preventing heating. These limitations, the fact that many of the devices (especially smart things) are designed by companies with little experience in computing and the thing status, meaning that users are unlikely to participate in installation of updates (so either the things will never be updated or will have a fully automated update mechanism, creating a tempting target for attacks) lead to a rather difficult situation from the security standpoint. While most of the things are seen as not worth attacking, the situation becomes worse when the entire heterogeneous network is seen as a single system. Unsafe devices are points of entry to the network, threatening other resources. They can also be used in orchestrated attacks, e.g. providing multiple consistent but false data streams leading to wrong decisions. With diminishing isolation, security of things becomes crucial.

The paper focuses on the client side of such a heterogeneous network – the (logically) local network of things, using multi-hop ad-hoc connections if transmission range is too short. The energy and power limitations, specialized hardware, wireless communication and minimal manual configuration characterizing most of smart things are also typical in wireless sensor networks (WSNs), making them a proto-example of a network of things. Many of the results of research in WSNs security are therefore almost directly applicable to other devices. The main difference is that a single WSN is usually rather homogeneous, while in case of the Internet of Things (IoT) the devices may be completely different in both hardware and software.

Detection of compromised nodes is most complicated in this resource- and energy-limited part of the heterogeneous network, where the extra load introduced by the detection mechanisms becomes too large. The tradeoff between having an insecure network or expending energy and resources on protective measures could be eliminated by introducing more powerful nodes dealing with this task. Unfortunately, providing sufficient network coverage would require many such nodes, effectively multiplying the system cost beyond sensible limits.

In this paper a workaround limiting the cost of detection nodes is proposed by allowing each detector to monitor multiple locations through mobility.

The paper starts with a discussion of major threats classes to such networks of small devices in Section 2. Section 3 provides an overview of the approaches toward securing such networks. Section 4 presents authors idea of a mobile intrusion detection system (IDS). A short conclusion is given in Section 5.

2. Threats to the Network

There are many possible modes of attack against a sensor-like network of things [1]–[3]. In general, they can be grouped depending on several factors, such as the activity of the attacker (active or passive), computing power (sensor class or laptop class), location (logically inside or outside

the network), target layer and attack goals (communication obstruction, data capture, modification or data fabrication).

Attacks in physical layer are hard to prevent, as neither the electromagnetic medium nor the sensors themselves are (usually) physically protected. Active jamming attacks are therefore effective, although rather easy to detect. Passive sniffing is effective unless encryption is used. Physical attacks on nodes are possible even without any tools (destruction or theft of nodes). More advanced physical attacks are dangerous to the network as a whole, because of the virtually unlimited possibility of tampering with the hardware and programming (e.g. using JTAG interface).

Attacks on the data layer are more limited, usually focusing on flooding the medium with messages, or using standard violations such as long frames to cause collisions.

Attacks in the network layer are potentially very effective, but made more difficult by the fact that routing in such networks is not part of the standard and may be done using a variety of algorithms. Attacks in this layer usually focus on affecting the routing decisions in order to either obstruct communication as such, or to maximize the effectiveness of the limited number of malicious nodes in the network. In the first case, providing false information in the path building phase or sending many unnecessary path queries are simple and quite effective sensor-class attacks, causing additional unnecessary communication and computation by network nodes, draining batteries. The second group of attacks uses advertising great connection quality (or other methods) in order to direct as much of the network's traffic as possible through a malicious node. Then, after routing is established, the node can be used to monitor the traffic (sniffing) or obstruct it, either by blackholing the communication or by selective forwarding increasing loss frequency.

Finally, attacks in the higher layers (transport – application) are also possible and potentially most useful in case of targeted attacks. The range of possibilities is too wide to describe here. As simple examples consider attacks on time synchronization algorithms, node location or key distribution. An attack in this layer, conducted with a deep understanding of the goals and implementation of the network, can turn the network into an extremely dangerous misinformation tool.

3. Security Measures

Due to the limited computing power of nodes and their need to conserve energy, any security measure that requires a lot of computing activity on part of the network nodes is a mixed blessing. Another problem is the broadcast-based, self-organizing dynamic nature of such networks – even if not mobile, they must reorganize to allow for node malfunctions, etc. There are no natural policy enforcement points apart from the base station – any node in the network may be routed around. These problems result in a reduced choice of security solutions for networks of things.

3.1. Intrusion Prevention

The first layer of defense is provided by protection measures aimed at preventing successful penetration. In case of wireless networks of resource-limited devices this layer is unfortunately not as strong as in wired computer networks. Since the medium is freely accessible, the prevention must be applied at every point in the network. However, application of advanced mechanisms is made difficult by the limited computing resources and the need to preserve energy. Still, some steps have been made towards provision of important information security protections.

Proper application of cryptographic techniques can provide privacy, authentication and data integrity. Unfortunately, software implementations require a lot of operations, lowering battery life. Hardware support reduces this impact and is available in radio modules implementing the IEEE 802.15.4 standard [4]. An unfortunate limitation of this solution is the use of a single symmetric key. A lot of work towards introducing cryptographic protections to higher layers and enabling efficient and secure key distribution has been performed in recent years, including e.g. TinySec [5], MiniSec [6], ContikiSec [7], ZigBee, LEAP/LEAP+ [8].

Another protective measure, most effective not in prevention of attacks, but in network protection against already malicious nodes, is trust management [9]. Due to limited memory in network nodes the most practical approach is reputation based. An example of its application to sensor networks can be found in [10].

3.2. Intrusion Detection

Once a successful attack has been performed, the network might be operating with one or more malicious nodes. This constant threat to information security is often more dangerous than the initial attack. Detection of misbehaving nodes allows proper mitigation techniques to be applied, including blacklisting the node and routing around it or even physically removing it from the network (if possible). Many methods have been proposed to detect malicious nodes. Most of them have a common problem – secure delivery of detection information to the base station or secure propagation of information between non-malicious nodes. If a single malicious node is detected, it might not be the only one. If another one is in path of the warning message, it can easily render the detection system powerless. Therefore IDS alerts require either a separate secondary channel for propagation, or effective protection if the primary channel is used (separate path, encryption, etc).

The simplest form of detection of malicious behavior is the watchdog mechanism [11], using the shared medium aspect of wireless networks. The node sending a message can monitor the medium to verify whether the receiving node forwarded it correctly. There are many variations to this mechanism in the literature. A different, more sophisticated approach is using more advanced, rule-based intrusion detection systems [12], capable of detecting many different kinds of attacks. Both approaches require modification of

all or selected nodes. A standard watchdog mechanism only detects malfunctioning neighbors, so it must be active in most of the nodes to cover the entire network. An IDS may be somewhat effective even if only implemented at the base station, but delivers less information.
One more approach, often used in classical networks, is a server honeypot – a service existing only as a target for attacks. This approach might be applicable to WSNs by emulating a node on a more powerful device, waiting for messages modifying its state in illegal ways. This detection method would be effective against previously unknown types of attacks as long as the identity of the honeypot node remains secret.
All of the previously described approaches are generally passive – they either base detection on received traffic only or (in case of the server honeypot) respond to messages, but never initiate communication as part of the detection activity. An active alternative is a client honeypot – a node, which sends messages in order to verify whether they are properly forwarded to the base station or other target. The actual detection is performed at the receiving node, where any changes to the message or variation from the normal loss rate can be easily identified. The mechanism requires that both the sender and the receiver agree on the nature of the testing message or sequence of messages. This can be done through predefinition or by using a secondary channel for transmission of this information. Note that predefined messages are easier to learn and avoid at malicious nodes.

4. Movement-assisted Threat Monitoring in WSN

Taking into account limited resources of sensors, collection and analysis of data concerned with network security are usually performed in a periodic manner and carried out by selected devices implementing threat monitoring capabilities. However, in general, it is possible to extend the functionality of all nodes and to implement permanent monitoring. Regardless of the selected monitoring scheme, a common objective of all security systems is establishing a safe and reliable communication channel for exchanging security information i.e. reporting, alerting, control traffic, etc. between nodes. Therefore the communication can be organized in several ways:

- by utilizing the transmission channels already set up to propagate data in the system,
- by creating channels using disjoint logical connections within existing networks,
- by adding extra nodes to create a separate sensor network which shares the same transmission medium,
- by equipping sensor nodes with additional hardware modules (Wi-Fi, GSM, etc.) that can be used to establish an additional communication channel.

Finally, threat monitoring can be successfully supported by controlled geographic migration of sensors that have locomotion. A mobility of sensors is leveraged recently for many WSN applications. Using mobile platforms to assist sensor placement in a working space can significantly enhance the capability to monitor the data and detect attacks.

4.1. MOTHON System Overview

The authors have proposed a novel approach to threat monitoring in WSN. In presented threat detection system one or several mobile sensors implementing threat detection functionality are forced to move to desired directions. As it is presented in Fig. 1, due to the ability to change the location of a sensor node, information about security events can be passed directly to the network sink (IIIa – after completion of all tasks, IIIb – after threat detection) or indirectly via other nodes from another area of the monitored network.

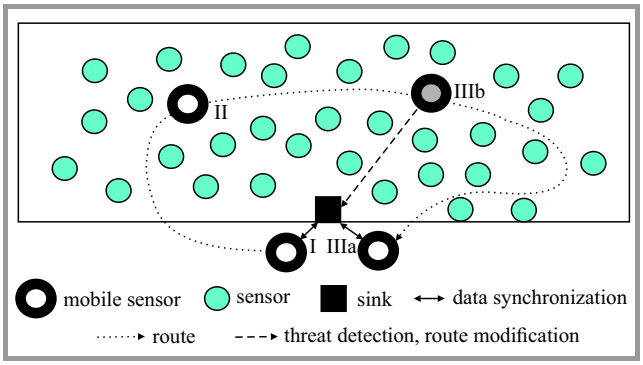

Fig. 1. The concept of movement-assisted threat monitoring: I – task order phase, II – performing actions phase, III – reporting phase (a – after completion of all tasks, b – after threat detection).

The MOTHON (MObile THreat mONitoring for WSN) system implements the concept depicted in Fig. 1. It is composed of three components: a mobile platform (MP), threat monitoring sensor (TMS) and management station (MS) responsible for controlling of MP and TMSs (see Fig. 2). It is assumed that the mobile platform can carry one or several TMS sensors. All these sensors can be placed in any locations in a workspace.
The MOTHON operates in three stages. The management station initiates the first stage. The aim of the this stage is to synchronize and gather data about a given network (topology, characteristics of nodes, etc.) and define monitoring mode and plan. Passive and active modes of threat detection are considered. Next, all data related to decisions done by MS are transferred to a MP and TMS sensor (or sensors) dedicated to threat monitoring, and the system switches to the second stage.
Threat monitoring sensor is carried to the desired destination by mobile platform (MP). After placing at the target location TMS starts to perform assigned tasks concerning the threats detection. The third stage relates to re-synchronization of information between TMS and MS which can occur in two cases – after completion of all

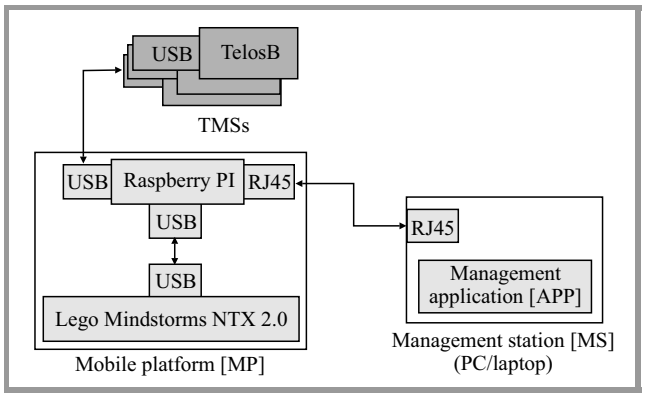

Fig. 2. MOTHON prototype architecture.

tasks or after threat detection. TMS transfers collected information, e.g. detected threats, additional statistics about traversed route, etc. to MS. It can be implemented in two ways:

- all data gathered by TMS are stored in MP, which transfers this data to MS directly – just after threat detection or after completion of all tasks,
- using existing communication channel via other nodes from another area of the monitored network.

In both cases MP can leave the TMS (with default network software) to avoid the occurrence of "temporary" nodes in the system. Moreover one mobile platform can carry multiple TMS sensors and place them in different locations in workspace.

4.2. Detection Methods

MOTHON can employ either active or passive methods for threat detection. Data analyses can be carried out either on-line by the TMS and mobile platform, or post factum by the mobile platform and management station.

Passive methods, which are based on analysis of information received from neighbors (IDS) or data sniffed from a shared medium (watchdog IDS), can keep a copy of the observed traffic for further analysis. This is not effective approach in case of threat monitoring with static nodes, but can be very valuable in case of mobile platform returning to sink from time to time. Moreover, simultaneous monitoring of the communication channel from several locations in the workspace can ease analysis by allowing detection of hidden and exposed nodes problems.

In contrast to passive methods, active solutions are not limited only to verification of individual sensor actions (correct operation of protocols, transmitted information, etc.). Active methods provide tools for verification of the network operation as a whole, e.g. verification the service packet forwarding over the network to the sink by sending a specific content, at the specific time and from specific place.

4.3. MOTHON Prototype Architecture

The prototype system composed of three elements is presented in Fig. 2. The management application provided by MS is a console tool is written in C++.

Mobile platform consists of two hardware components: Lego Mindstorms NTX 2.0 and Raspberry Pi single board microcomputer. Moreover it is expected, that MP will be equipped with GPS module or other localization system [13], [14].

Fulfilling all the tasks assigned by management application requires from this platform significantly greater capabilities in terms of processing power and energy resources. The key software components of MP are:

- **Control module** – the main module, responsible for communication with all other modules and creating the logic of solution based on information obtained from the management station.
- **Mobility module** – responsible for motion trajectory planning movement and speed calculation.
- **Threat analysis module** – used for data gathered from TMSs modules analysis and threat detection.

TMS is implemented over TelosB platform using Contiki OS. Eventually, to complete the system, an automation process enabling wireless communication between MP an MS must be added.

5. Conclusion

Starting with a review of threats and security measures applicable to wireless networks of resource-limited things, a new approach, introducing mobility as a way of overcoming the limitations of existing methods has been presented.

Mobility of a threat detection sensor should improve overall security state of monitored networks without any need to perform their reconfiguration or upgrade. The approach can be used in existing networks without any modifications to installed devices. The implementation details, such as means of mobility, depend on the target network – obviously a different solution is appropriate inside a building than in case of a network of oceanic drones.

Various methods of threat detection in MOTHON are currently under development. In future work authors plan to conduct experiments in testbed network to show the effectiveness of detection against different kinds of attacks.

References

[1] H. K. Kalita and A. Kar, "Wireless sensor networks security analysis", *Int. J. of Next Gener. Netw.*, vol. 1, no. 1, pp. 1–9, 2009.

[2] Z. S. Bojkovic, B. M. Bakmaz, and M. R. Bakmaz, "Security issues in wireless sensor networks", *NAUN Int. J. Commun.*, vol. 2, no. 1, pp. 106–115, 2008.

[3] I. Butun, S. D. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks", *IEEE Commun. Surveys & Tutorials*, vol. 16, no. 1, pp. 266–282, 2014.

[4] IEEE standard for part 15.4: IEEE Std. 802.15.4, IEEE, New York, Oct. 2011.

[5] C. Karlof, N. Sastry, and D. Wagner, "TinySec: A link layer security architecture for wireless sensor networks", in *Proc. 2nd ACM Conf. on Embedded Netw. Sensor Syst. SenSys 2004*, Baltimore, Maryland, USA, 2004, pp. 162–175.

[6] M. Luk, G. Mezzour, A. Perrig, and V. Gligor, "MiniSec: a secure sensor network communication architecture", in *Proc. 6th Int. Conf. on Inform. Process. in Sensor Netw. IPSN'07*, Cambridge, MA, USA, 2007.

[7] L. Casado and P. Tsigas, "ContikiSec: A secure network layer for wireless sensor networks under the Contiki operating system", in *Proc. 14th Nordic Conf. on Secure IT Syst.: Identity and Privacy in the Internet Age NordSec 2009*, Oslo, Norway, 2009, pp. 133–147.

[8] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks", *ACM Trans. on Sensor Netw.*, vol. 2, no. 4, pp. 500–528, 2006.

[9] A. Felkner, "How the Role-based trust management can be applied to wireless sensor networks", *J. Telecommun. Inform. Technol.*, no. 4, pp. 70–77, 2012.

[10] G. Saurabh, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks", *ACM Trans. on Sensor Netw.*, vol 4, no. 3, 2008.

[11] F. Hu, J. Ziobro, J. Tillett, and N. K. Sharma, "Secure wireless sensor networks: problems and solutions", *J. Syst., Cybernet. and Inform.*, vol. 1, no. 4, pp. 90–100, 2003.

[12] G. Huo, X. Wang, "DIDS: A dynamic model of intrusion detection system in wireless sensor networks", in *Proc. IEEE Int. Conf. on Inform. Autom. ICIA 2008*, 2008, ZhangJiaJie, China, pp. 374–378.

[13] M. Marks, E. Niewiadomska-Szynkiewicz, and J. Kolodziej, "An integrated software framework for localization in wireless sensor network", *Comput. and Inform.*, vol. 33, no. 2, pp. 369–386, 2014.

[14] M. Marks, E. Niewiadomska-Szynkiewicz, and J. Kolodziej, "High performance wireless sensor network localization system", *Int. J. Ad Hoc and Ubiquitous Comput.*, vol. 17, no. 32, pp. 122–133, 2014.

Adam Kozakiewicz got his M.Sc. in Information Technology and Ph.D. in Telecommunications at the Faculty of Electronics and Information Technology of Warsaw University of Technology (WUT), Poland. Currently he works at NASK as Assistant Professor and Manager of the Network and Information Security Methods Team, also as part-time Assistant Professor at the Institute of Control and Computation Engineering at the WUT. His main scientific interests include security of information systems (especially industrial networks), parallel computation, optimization methods and network traffic modeling and control.

E-mail: adam.kozakiewicz@nask.pl
Research and Academic Computer Network (NASK)
Wawozowa st 18
02-796 Warsaw, Poland

Krzysztof Lasota works as Research Associate at Network and Information Security Methods Team in the Research Division of NASK. He received his B.Sc. in Telecommunications (2010) and M.Sc. in Telecommunications (2011) from Warsaw University of Technology, Faculty of Electronics and Information Technology and is currently a Ph.D. student there. He participated in security-related projects at NASK, including FISHA, HoneySpider Network and Secure workstation for special applications. Currently he participates in the project "The system of secure IP communication provision for the power system management". Furthermore, his research aims at developing new methods for threat detection in wireless sensor networks.

E-mail: krzysztof.lasota@nask.pl
Research and Academic Computer Network (NASK)
Wawozowa st 18
02-796 Warsaw, Poland

Michał Marks received M.Sc. (2007) in Computer Science and Ph.D. (2015) in Automation and Robotics from the Warsaw University of Technology. Since 2007 with Research and Academic Computer Network (NASK). The author and co-author of over 30 journal and conference papers. His research area focuses on wireless sensor networks, global optimization, distributed computation in CPU and GPU clusters, decision support and machine learning.

E-mail: michal.marks@nask.pl
Research and Academic Computer Network (NASK)
Wawozowa st 18
02-796 Warsaw, Poland