FI MU

Faculty of Informatics
Masaryk University Brno

Attackers in Wireless sensor Networks Will Be Neither Random nor Jumping – Secrecy Amplification Case, Extended Version

by

Radim Ošťádal
Petr Švenda
Václav Matyáš

FI MU Report Series **FIMU-RS-2016-04**

Copyright © 2016, FI MU **09 2016**

**Copyright © 2016, Faculty of Informatics, Masaryk University.
All rights reserved.**

**Reproduction of all or part of this work
is permitted for educational or research use
on condition that this copyright notice is
included in any copy.**

Publications in the FI MU Report Series are in general accessible via WWW:

`http://www.fi.muni.cz/reports/`

Further information can be obtained by contacting:

**Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic**

Attackers in Wireless Sensor Networks Will Be Neither Random nor Jumping – Secrecy Amplification Case, Extended Version

Radim Ošťádal
Masaryk University
ostadal@mail.muni.cz

Petr Švenda
Masaryk University
svenda@fi.muni.cz

Václav Matyáš*
Masaryk University
matyas@fi.muni.cz

September 14, 2016

Abstract

Partially compromised network is a pragmatic assumption in many real-life scenarios. Secrecy amplification protocols provide a significant increase in the number of secure communication links by re-establishing new keys via different communication paths. Our paper shows that so far research in the area of secrecy amplification protocols for wireless sensor networks has been based on rather simplified foundations with respect to attacker models. The attacker does not behave randomly and different attacker capabilities and behaviour have to be considered. We provide means to experimental work with parametrisable attacker capabilities and behaviour in realistic simulations, and evaluate the impact of the realistic attacker properties on the performance of major amplification protocols.

We also show which secrecy amplification protocols perform best in different attacker settings and help to select a protocol that exhibits good results in a prevalent number of inspected scenarios.

This is the extended version of our paper that is to be presented at 15th International Conference on Cryptology and Network Security (CANS 16) in Milan, Italy, November 14-16, 2016.

*This author was partly supported by the Czech Science Foundation project GBP202/12/G061.

1 Introduction

Ad-hoc networks of nodes with varying capabilities (including quite limited ones) often handle sensitive information and security of such networks is a typical baseline requirement. Such networks consist of numerous interacting devices, price of which should often be as low as possible – limiting computational and storage resources, also avoiding expensive tamper resistance. Lightweight security solutions are preferable, providing a low computational and communication overhead. When considering key management, symmetric cryptography is the preferred approach, yet with a low number of pre-distributed keys. While all results we present can be applied to general ad-hoc networks, we present them directly on wireless sensor networks (WSNs) as typical representatives.

Attackers in such an environment can be categorised into different classes with respect to link key management. The most prevalent node-compromise model [5] assumes that the attacker is able to capture a fraction of deployed nodes and to extract keying material from captured nodes. No tamper resistance of nodes is assumed because of their low production cost. The weakened attacker model was defined in [1]. In this model, an attacker is able to monitor only a small proportion of communications within a network during the deployment phase when the link keys are being established.

Substantial improvements in resilience against node capture or key exchange eavesdropping can be achieved when a group of neighbouring nodes cooperates in an additional secrecy amplification (referred to as amplification protocols hereafter) after the initial key establishment protocol. Amplification protocols were shown to be very effective, yet for the price of a significant communication overhead. The overall aim is to provide amplification protocols that can secure a high number of links yet require only a small number of messages and are easy to execute and synchronize in parallel executions in the real network. Different types of amplification protocols were studied – node-oriented protocols, group-oriented protocols, and hybrid-design protocols.

Previous work on amplification protocols considered the close connection between the attacker model and a key establishment scheme used [1, 5, 13]. Partially compromised networks with only two different compromise patterns were inspected throughout literature – *random compromise* and *key infection* patterns. Random compromise pattern is the result of the node compromise attacker model together with a probabilistic

pre-distribution key establishment scheme [5]. The key infection pattern assumes the weakened attacker model together with the key establishment where link keys are being exchanged in plaintext. After an initial compromise, a global passive attacker that is able to monitor all communication on the entire network was expected in both cases.

We argue that those scenarios are not sufficient and in this work, we provide a more realistic setting for the attacker. Firstly, we question the initial compromise patterns inspected so far, as the attacker presence in the network during the deployment and a relatively short initial key establishment phase is a strong assumption. We focus on a network where all neighbours already share unique link keys, which means the key establishment protocol is not important. The attacker is able to initially compromise several nodes and extract all keys shared with its neighbours. We inspect multiple compromise patterns resulting from different attacker strategies, not only the random compromise pattern. Secondly, we do not expect the global attacker during the amplification phase as this would not be the case in real life (e.g., wireless receiver sensitivity limiting the attacker eavesdropping range). A realistic attacker has to be present in the network and will need to keep her stronghold in the network during the amplification phase. She has to eavesdrop as many random nonces used during the amplification process as possible. The attacker is parametrised by her capabilities and behaviour (e.g., initial compromise pattern, eavesdropping range, attacker movement and her speed etc.).

Apart from the attacker characteristics, we want to move the amplification protocol simulation to a more realistic setting. A significant part of recent work is based on results from SensorSim, a dedicated simulator developed specifically for security analysis of key distribution protocols and message routing by the authors of [16]. We extend the KMSforWSN framework that was introduced in [6]. Our extension is available as open source [1]. The advantages and disadvantages of both simulators are further discussed in the next section.

Our goals are:

1. To evaluate the impact of the realistic attacker properties on the performance of major amplification protocols.

2. To move the evaluation of amplification protocols to more realistic environment (suitable and realistic simulator).

[1] Full details, paper supplementary material and source codes can be found at http://crcs.cz/papers/cans2016.

3. To select one (or two) amplification protocols that exhibit good results in a prevalent number of inspected scenarios for further implementation and deeper analysis. Those are left as a future work.

The paper roadmap is as follows: the next section provides an overview of related work on different attacker models, current state of amplification protocols research and advantages and disadvantages of different simulators. Section 3 describes parametrisable attacker capabilities and behaviour together with experiment settings and network lifetime from deployment up to evaluation. Section 4 evaluates the impact of attacker parameters on success rate of 7 major amplification protocols. The best performing amplification protocol is selected and conclusions are provided in Section 5.

This is the extended version of our paper that is to be presented at 15th International Conference on Cryptology and Network Security (CANS 16) in Milan, Italy, November 14-16, 2016.

2 Related work

2.1 Attacker models

Several different attacker models were defined in the literature. We differentiate two basic categories based on a level of attacker interaction with the network. The global passive attacker is able to monitor all communication around the entire sensor network without influencing it. The global active attacker is the classic attacker from the Needham-Schroeder model [12]. She is able to alter and copy any message, replay messages or inject any false material. She might drop part of the communication at her will. Those attacker models define the attacker capabilities during the amplification process.

Another two attacker classes were introduced in literature with respect to initial network compromise – a node compromise model [5] and a real world attacker model [1]. The node-compromise model is an extension of the Needham-Schroeder model with these additional assumptions:

- The key pre-distribution site is trusted, i.e., nodes might be pre-loaded with secrets in a trusted environment before actual deployment.
- The attacker is able to capture a fraction of deployed nodes as no physical control over deployed nodes is assumed. This happens especially in the case when nodes are deployed in a hostile environment.

- The attacker is able to extract keying material from a captured node.

Real world attacker model (also called weakened attacker model) is defined in [1] and assumes the following conditions:

- The attacker does not have a physical access to the deployment site during the deployment phase.
- The attacker is able to monitor only a small proportion on the communications of the sensor network during the deployment phase. Once the key exchange is complete, she is able to monitor all communication at will.
- The attacker is unable to execute active attacks (such as jamming or flooding) during the deployment phase. Once the key exchange is complete, she is free to launch any kind of attack.

2.2 Compromise patterns

A compromise pattern provides us with a conditional probability that link *Y* is compromised when another link X is compromised after a relevant attack. The characteristics of a particular compromise pattern may significantly influence the success rate of a amplification protocols executed later.

The random compromise pattern arises when a probabilistic key pre-distribution scheme of [5] and many later variants of [2, 4, 10, 11] are used and an attacker extracts keys from several randomly captured nodes. The random compromise pattern exhibits an almost uncorrelated pattern. To the contrary, the key infection compromise pattern forms a significantly correlated pattern due to an eavesdropper locality. During the key establishment, link keys are being exchanged in plaintext. The original idea of key infection was presented in [1] and later extended by [3, 8, 16].

2.3 Secrecy amplification

The secrecy amplification concept was originally introduced in [1] for the key infection plaintext key exchange, but can be used for any partially compromised network. A secrecy amplification protocol can be executed to secure (again) some of the compromised links, resulting in a less compromised network. During the amplification protocol, a group of neighbours cooperates together to exchange random nonces that will be later

used to update original link keys. Nonces have to be securely transported to both nodes to update the mutual key. A particular amplification protocol specifies the exact way the generated nonces are transported.

A network owner usually does not know which concrete link key was compromised by an attacker and which was not. Secrecy amplification can be executed as a response to a (presumed) partial compromise already happened or as a preventive measure for potential future compromise. Secrecy amplification can be also executed as another layer of protection even if a particular link key might not be compromised at all.

Amplification protocols can try all possible paths, yet for the price of a huge communication overhead. Proposed amplification protocols therefore aim to find a good tradeoff between the number of paths tried and the probability of finding at least one secure path for nonce delivery.

Different classes of amplification protocols use different means to improve a security throughout the network. Although all amplification protocols aim to setup new (possibly more secure) link key, three main distinct classes of amplification protocols exist:

A node-oriented protocol sends key updates via every possible neighbour or neighbours by a simple protocol. Note that node-oriented protocol is executed for all possible k-tuples of neighbours in the network. A number of such k-tuples can be high, especially for dense networks. We further inspect five selected node oriented protocols: Pull [3], Push [1], Multi-hop Pull (M-Pull) [3], Multi-hop Push (M-Push) [1] and NO Best [16].

A group-oriented protocol shares new key values inside a bigger group of cooperating nodes identified by their geographical areas in a form of relative distance to selected nodes [16]. Group-oriented protocols were never implemented because of their prevailing disadvantages – problematic synchronisation of parallel executions and complicated security analysis due to a high number of nodes involved. We omit group-oriented protocols from further analysis.

A hybrid-design protocol uses sub-protocols (similarly to node-oriented), relative distances (similarly to group-oriented) and additionally utilise several repetitions of the whole process to achieve required success rate. We further inspect two hybrid designed protocols: HD Final and HD Best [13].

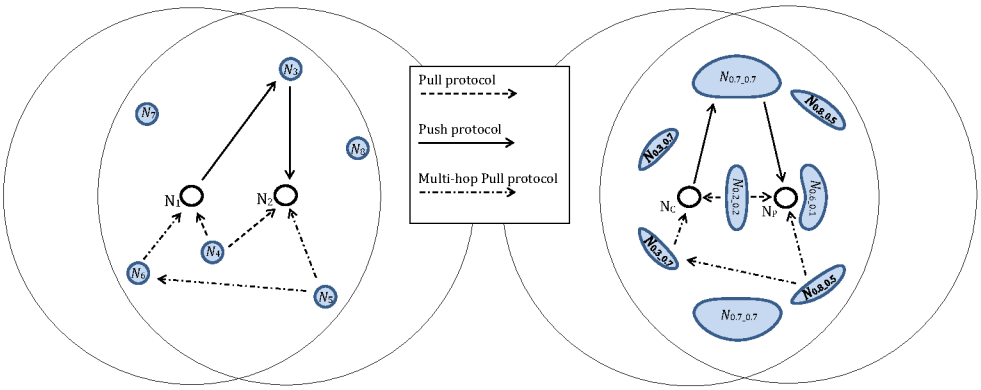

Figure 1: An example of Pull, Push and Multi-hop Pull amplification protocols. The distance between nodes N_1 and N_2 (N_C and N_P) is 0.5 of the maximal transmission range. Node-oriented protocols are shown on the left and examples of a basic hybrid designed amplification sub-protocols on the right. Selected node-relative identification (distance from N_C and N_P) of involved parties are displayed as the geographic most probable areas, where the intermediate nodes will be positioned.

Figure 1 shows examples of node-oriented and hybrid designed protocols. Details of all inspected protocols are provided in Appendix B.

2.4 Simulation environment

An essential part of amplification protocol evaluation usually is a simulation environment. The evaluation on a real sensor network is usually not possible due to its cost and time requirements. We provide a brief comparison of the SensorSim simulator used during the amplification protocols design in previous work and the KMSforWSN framework used to conduct our research.

The SensorSim simulator is a tool for very fast evaluation of existing amplification protocols [16]. New amplification protocols can be also generated using evolutionary algorithms [16]. The main advantage of SensorSim is the speed of simulation. However, it lacks many essential components for a realistic simulation, like radio signal propagation or MAC layer collisions. All protocols in SensorSim are evaluated based only on a set of properties, such as a number of nodes, node positions or defined communication range.

KMSforWSN framework is a tool for an automated evaluation of KMS properties in WSNs, built on top of MiXiM [9], a WSN framework for the OMNeT++ simulator [15]. We extend it with two new modules to reflect different attacker models and also

to implement a secrecy amplification capability. The definition of channel and physical layer settings is based on previous research on real parameters of TelosB sensors for outdoor environment [14]. In our work, we simulate the network execution not only as a graph discovery problem (as is the case for SensorSim), but full emulation of code running on virtual nodes is provided, with an application logic executed and messages passed to the communication stack.

3 Parameterized attacker

Our aim is to define the attacker with fully parametrisable capabilities and behaviour. Attacker parameters can be divided into two separate groups – *behaviour* parameters and *resource* parameters. The *behaviour* parameters characterise attacker strategy and behaviour during the attacker activity (e.g., different movement pattern or starting position). The *resource* parameters define available resources and attacker capabilities, both initial and extended (e.g., number of cooperating attackers or eavesdropping range).

We summarise particular phases of the entire simulation and provide a definition of all inspected attacker parameters. Baseline values are assigned to every parameter at the end of this section.

3.1 Network lifetime

The entire simulation of attacker against amplification protocols and its evaluation consist of several phases. Firstly, a network with several fixed parameters is deployed to a given area (particular parameters are described in the next paragraph). After the deployment, an attacker conducts the initial compromise. Secondly, the nonce distribution phase of amplification protocols and attacker eavesdropping take place simultaneously. Lastly, nonces are mutually confirmed among neighbouring nodes and the simulation is concluded with a protocol evaluation. The description of particular phases follows.

Network deployment: A network consists of one hundred legitimate nodes. All nodes are deployed randomly over the plane of 115x115 metres. The size was chosen purposefully to have a network with a density of 7.5 neighbours on average.

Initial compromise: Considering the link key security, the prevalent attacker model is node compromise. The attacker compromises all keys stored on a particular node. This initial compromise is done before the actual amplification protocol is

executed. We evaluate all compromise patterns defined in [7]. The total number of compromised links is 50%.

Nonce distribution phase: All nonces are distributed during this phase. The phase should influence network operations shortly as possible, so we limit the length to only 100 seconds. This length is enough for single hop node-oriented protocols (Pull, Push) and for hybrid designed protocols (HD Final, HD Best). Multi-hop node-oriented protocols (M-Pull, M-Push, NO Best) exhibit the loss of about 10 percent of nonces due to message collisions. An impact of the loss on the success rate of those protocols is very limited due to a high number of nonces generated and distributed.

Attacker eavesdropping: The attacker is present in the network during the whole nonce distribution phase and she tries to eavesdrop on as much communication as possible. Her success depends on particular values assigned to examined attacker parameters.

Nonce confirmation phase: Each pair of neighbours has to confirm (mutually) the common nonces that will be used for update of a shared key. After this confirmation, the key is refreshed immediately and used for all further communication. The confirmation phase follows the distribution phase sequentially.

3.2 Attacker parameters

Every parameter from both defined groups can be assigned of a value from a specific set of possible values described in this section. Possible values for attacker *behaviour* parameters follow.

(I) Initial compromise pattern: We are inspecting four different patterns: 1) Random nodes are selected and compromised in the *random* attacker pattern. 2) The attacker walking around the network and picking outermost nodes presents the *outermost* attacker pattern. 3) The attacker moving directly to a centre of the network from a random location on an edge of the network, picking up nodes in a close vicinity of his trajectory, presents a *direct centre* attacker pattern. 4) The *centre drop* attacker pattern simulates the possibility of parachute drop or digging under the network. The closest nodes to the centre of the network are compromised up to a selected threshold.

(II) Movement pattern of the attacker: During the nonce distribution phase, attackers move within the WSN deployment area according to an assigned pattern. We evaluate several different patterns to see how they influence the attacker success in eavesdropping nonce messages: 1) The *stationary pattern* is characterised by attackers staying in their initial positions and not moving at all. 2) Attackers move on a straight line with a constant speed in the *linear* movement pattern. When the attacker approaches an area border, she reflects at the same angle. 3) The *random* pattern is described by attackers choosing the point within the deployment area randomly (distributed uniformly over the area) and moving directly to it with constant speed. After reaching the point, attacker selects the next point, again in a random manner. 4) Attackers move in a circle of a particular diameter in the *circle* pattern. We inspect three different diameters of 10, 20 and 40 metres. 5) The square *patrol* pattern is characterised by the attacker systematically patrolling a square area with a side of different length – particularly 10, 20 and 30 metres.

(III) Initial location of the attackers: We inspect three different settings for the initial location of attackers when the nonce distribution phase starts: 1) All the attackers start from the same place in a *corner* of the deployment area. 2) Attackers are at *random* positions within the area. 3) Attackers cooperate and choose the *suitable* places to capture as much communication as possible (selected coordinates are [57.5, 57.5], [30, 30], [85, 30], [85, 85] and [30, 85] within the deployment area).

(IV) Movement speed of attackers: Attackers move at a constant speed. We inspect a range of speeds from a very slow walk up to the movement speed of a car or a flying drone.

The success of the attacker is closely connected with invested resources. The hypothesis to be verified is whether the more resources available, the more successful attacker is. We are also interested in a determination of a limit of attacker capabilities, where amplification protocols still represent meaningful strategy.

(V) Number of attackers: Several attackers might work together to eavesdrop as much communication as possible. A collaboration includes, but is not limited to, an exchange of captured nonces and compromised keys among individual attackers. More complicated scenarios are left to future research.

(VI) Eavesdropping range: A radius where attackers are able to intercept the communication (e.g., the three times the range of legal node). The range highly depends on an available equipment, and its sensitivity.

(VII) Number of malware infected nodes: During the initial compromise phase, the attacker installs her malware on compromised nodes. The node is under attacker's control, but the control remains passive – besides a monitoring purpose, the malware does not affect any behaviour of the node. We inspect the impact of increasing number of infected nodes on the success rate of amplification protocols.

3.3 Experiment setting

Evaluating the impact of particular attacker parameters on the overall success rate of amplification protocols, we successively inspect different values for selected parameter, while the rest of parameters are fixed to baseline values. As so, baseline values should be as little influencing as possible to have a clear result on the inspected parameter. Table 1 summarises the baseline setting for available attacker parameters.

All provided experimental results are an average of one hundred repetitions with different seeds for a random number generator. The evaluation was conducted on a dedicated machine with 96 cores at 2.00 GHz. The total computation time was more than 6 core years.

Attacker's *behaviour* parameters	
Initial compromise pattern	Random nodes are compromised
Movement pattern	Random movement
Initial locations of the attackers	Random positions selected
Movement speed of attackers	1.5 metres per sec ~ normal walk
Attacker's *resource* parameters	
Number of attackers	5 cooperating attackers
Eavesdropping range	30 meters reach
Number of malware infected nodes	No compromised/infected node

Table 1: Baseline settings for attacker parameters. Experiments varied one parameter at the time and observed influence on secrecy amplification success rate.

4 Experimental results

We have determined a ranking of amplification protocols based on their performance in a prevalent number of inspected cases. *The highest number of secured link keys is provided by the HD Best protocol, closely followed by the HD Final protocol.* Hybrid designed protocols provide better results than node-oriented protocols during the evaluation of all parameters. The NO Best protocol outperforms the rest of node-oriented protocols. Multi-hop Pull and Multi-hop Push protocols provide us with similar success rates and both outperform the Push protocol. The Pull protocol exhibits the worst results. Please note that the Pull protocol sends only a half of nonces compared to the Push protocol.

4.1 Impact of compromise patterns

We have inspected four different initial compromise patterns (parameter I). In all cases, a compromised portion of link keys is 50%. This results in a different number of compromised nodes for every pattern. The lowest number of compromised nodes (29.73) is in the *random* pattern as there is a low number of overlappingly compromised links among compromised nodes. To the contrary, the highest number of compromised nodes (48.29) is in the *outermost* pattern as legitimate nodes on the boundary have the lowest number of connections to other nodes and more, links are shared among compromised nodes. We observe comparable numbers for *direct centre* and *centre drop* patterns (39.56 and 35.71 nodes, respectively). Links are shared among compromised nodes that are concentrated in one area.

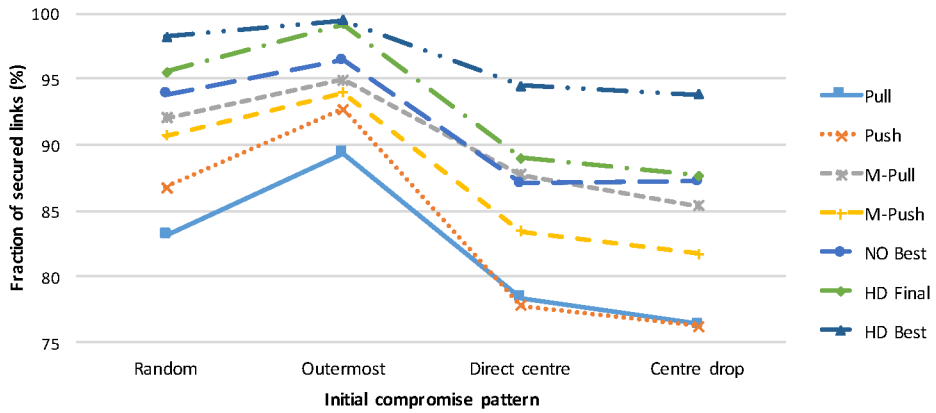

Figure 2: Success rate of amplification protocols for different initial compromise patterns. The initial compromise rate is 50% of all link keys.

The performance of amplification protocols for different initial compromise patterns (parameter I) is showed in Figure 2. Significantly worse results are provided by all amplification protocols on *direct centre* and *centre drop* compromise patterns due to the high concentration of compromised nodes in one area. *However, hybrid designed protocols are able to achieve nearly 95% of secured links from initial 50% even with those unsuitable settings.* The initial compromise pattern is the only parameter where we can observe a significant difference in performance of HD Final and HD Best protocols.

4.2 Impact of movement patterns

Different movement patterns (parameter II) were examined and results are shown in Figure 3. *The most successful strategy for the attacker is to stay on the same place (stationary pattern) as she is able to eavesdrop all communication within a particular area.* Comparable results are achieved by the *circle* pattern with a small radius of 5 metres and by the systematic patrolling over a small square area. The reason for the attacker's success is the same as in the *stationary* case. The worst movement pattern for the attacker is *linear* as the attacker spends a lot of time in a border area where eavesdrops less communication. Altogether, amplification protocols are able to achieve 75% of secured link keys from the initial 50% even in the worst case of the *stationary* pattern.

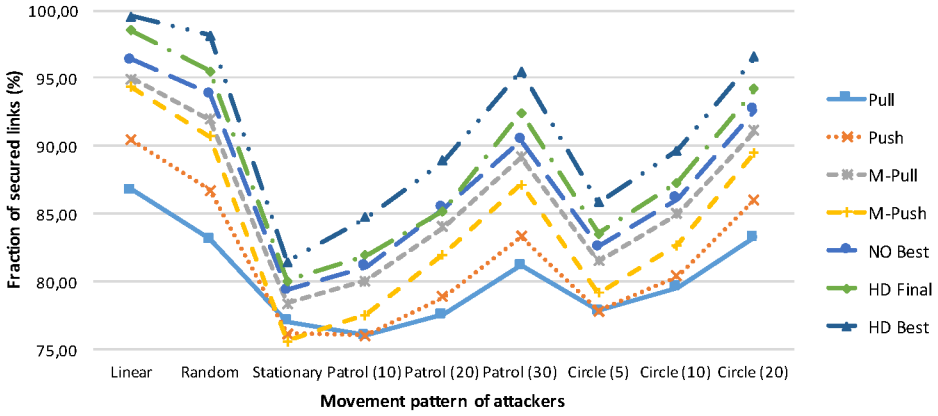

Figure 3: Success rate of amplification protocols for different movement patterns of attackers. The number in brackets after *Patrol* and *Circle* patterns denotes the length of square area side and the circle diameter respectively. The initial compromise rate is 50% of all link keys.

4.3 Impact of position and speed

Evaluation of results for different initial starting positions (parameter III) of attackers is shown in Figure 4. All amplification protocols exhibit the highest success rate for attackers starting in the *corner* of the deployment area. Attackers are able to monitor only a small part of the network from the beginning. We observe a constant drop in the success rate of 2% between *random* and *suitable* attacker's starting positions for all amplification protocols. Comparing the *corner* and *random* starting positions, the hybrid designed protocols exhibit the least drop in the success rate whereas the single hop node-oriented protocols show the highest drop.

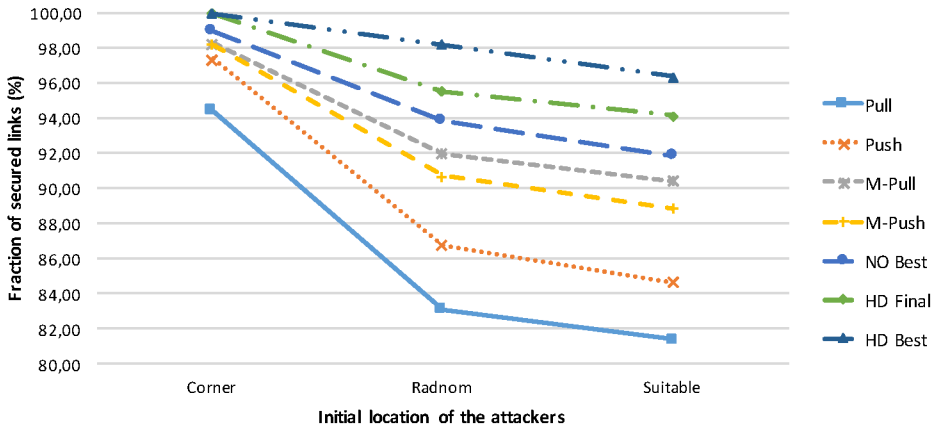

Figure 4: Success rate of amplification protocols for different initial position of attackers. The initial compromise rate is 50% of all link keys.

A comparison of different movement speeds of attackers (parameter IV) is shown in Figure 5. *The slower movement speed of attackers, the worse results achieved by amplification protocols in general.* The reason is that the attacker is able to eavesdrop most of the messages in a particular area and amplification protocols are not able to secure additional link keys in that region. This result is in line with the observation of the case of *stationary* movement pattern. Hybrid design protocols are able to face the challenge much better than node-oriented protocols and provide significantly better results for slow attackers up to a speed 1.5 metres per second. For a higher attacker speed, which means decreased attacker success, hybrid designed protocols provide still better success ratios, however, the differences are not so eminent as overall success rate is already high.

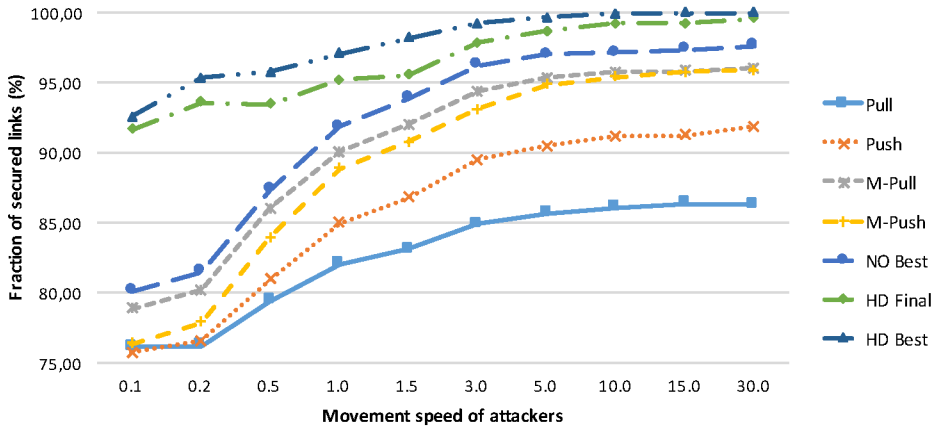

Figure 5: Success rate of amplification protocols for different movement speed of attackers. The initial compromise rate is 50% of all link keys.

4.4 Impact of *resource* parameters

The evaluation of parameters from the *resource* group supports the hypothesis stated at the beginning. The more resources available to the attacker, the more successful the attacker is. This holds for the increasing number of attackers, larger eavesdropping range and the increasing number of malware infected nodes. Hybrid designed protocols are able to provide reasonable improvement (85% of secured links from original 50%) for up to 10 cooperating attackers, 40 metres of attacker eavesdropping range, or up to 7 malware infected nodes out of 29 compromised. Detailed results are provided in Appendix A.

5 Conclusions

Our work shows how narrow the view of attackers in ad hoc networks has been so far. We provide a more realistic view of that attacker, with a definition of her capabilities and behaviour. With respect to the protocols examined, we show that the hybrid designed protocols outperform the rest in all scenarios we examined, and that these protocols are quite robust across different attacker behaviour and capabilities. Note that the NO Best protocol provides almost same results as the HD Final protocol, yet this comes at the price of an enormous increase of messages sent. We also demonstrate that the hybrid designed protocols use a low number of messages and provide a great improvement for the link key security. Our results do not assume a particular compromise scenario dur-

ing key establishment and are concerned only about the final fraction of compromised links, implying that the results can be generalised. Our work is based on realistic simulation of all components, which often get overlooked in protocols analyses coming right from particular protocol designers – we consider network communication (MAC, collisions), physical layer setting, etc. and we implemented the application to run directly on virtual nodes.

We found that one of the most significant parameters influencing the final performance of amplification protocols is the initial compromise pattern. This is the first work with analysis of additional initial compromise patterns apart from the random one.

Last but not least, we point out that often the most favorable strategy for an attacker is to stay on one place during the whole secrecy amplification process.

References

[1] Ross Anderson, Haowen Chan, and Adrian Perrig. Key infection: Smart trust for smart dust. In *12th IEEE International Conference on Network Protocols*, pages 206–215. IEEE, 2004.

[2] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Security and Privacy*, pages 197–213, 2003.

[3] Daniel Cvrček and Petr Švenda. Smart dust security-key infection revisited. In *Electronic Notes in Theoretical Computer Science*, volume 157, pages 11–25. Elsevier, 2006.

[4] Roberto Di Pietro, Luigi V. Mancini, and Alessandro Mei. Random key-assignment for secure wireless sensor networks. In *1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, pages 62–71, 2003.

[5] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *9th ACM Conference on Computer and Communications Security, Washington, DC, USA*, pages 41–47. ACM, 2002.

[6] Filip Jurnečka, Martin Stehlík, and Vashek Matyáš. Evaluation of key management schemes in wireless sensor networks. In *Security and Trust Management*, pages 198–203. Springer, 2014.

[7] Filip Jurnečka, Martin Stehlík, and Vashek Matyáš. On node capturing attacker strategies. In *Security Protocols XXII*, pages 300–315. Springer, 2014.

[8] Yong Ho Kim, Mu Hyun Kim, Dong Hoon Lee, and Changwook Kim. A key management scheme for commodity sensor networks. In *4th International Conference on Ad Hoc and Wireless networks, LNCS 3738*, pages 113–126. Springer, 2005.

[9] Andreas Köpke et al. Simulating wireless and mobile networks in OMNeT++ the MiXiM vision. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 71. ICST, 2008.

[10] Donggang Liu and Peng Ning. Establishing pairwise keys in distributed sensor networks. In *10th ACM Conference on Computer and communications security*, pages 52–61. ACM Press, 2003.

[11] Donggang Liu, Peng Ning, and Rongfang Li. Establishing pairwise keys in distributed sensor networks. *ACM Trans. Inf. Syst. Secur.*, 8(1):41–77, February 2005.

[12] Roger M Needham and Michael D Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12):993–999, 1978.

[13] Radim Ošťádal, Petr Švenda, and Václav Matyáš. A new approach to secrecy amplification in partially compromised networks. In *Security, Privacy, and Applied Cryptography Engineering – 4th International Conference, SPACE 2014, LNCS 8804*, pages 92–109. Springer, 2014.

[14] Andriy Stetsko, Martin Stehlik, and Vashek Matyas. Calibrating and comparing simulators for wireless sensor networks. In *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, pages 733–738. IEEE, 2011.

[15] András Varga. Using the OMNeT++ discrete event simulation system in education. *Education, IEEE Transactions on*, 42(4):11–pp, 1999.

[16] Petr Švenda, Lukáš Sekanina, and Václav Matyáš. Evolutionary design of secrecy amplification protocols for wireless sensor networks. In *Second ACM Conference on Wireless Network Security*, pages 225–236, 2009.

Appendix A Other results

A.1 Number of attackers and their eavesdropping range

Success rates of amplification protocols for a growing number of attackers (parameter V) and their increasing eavesdropping range (parameter VI) are shown in Figure 6 and 7, respectively. Single hop node-oriented protocols provide a reasonable improvement (85% of secured links from original 50%) up to 5 attackers, their multi-hop versions up to 7 attackers and hybrid design protocols up to 10 attackers.

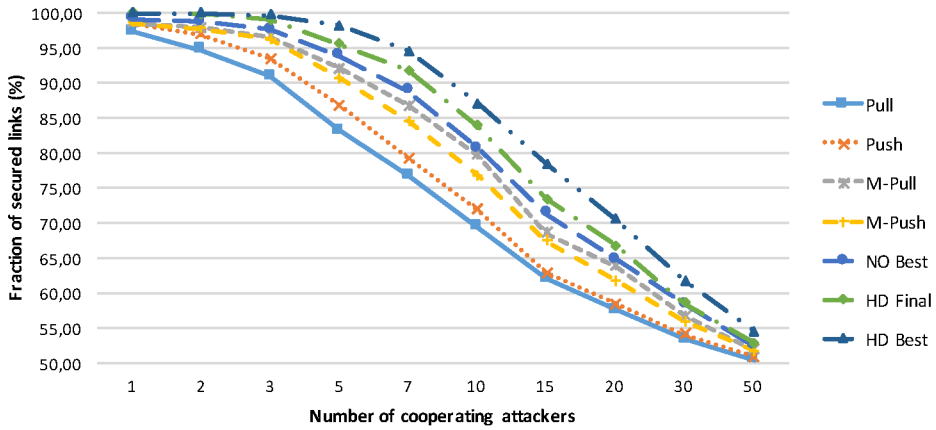

Figure 6: Success rate of amplification protocols for different number of attackers. The initial compromise rate is 50% of all link keys.

Hybrid designed protocols show nearly no drop in the success rate for eavesdropping range up to 30 metres and exhibit a big drop for 40 metres, where the covered area is nearly doubled. The success rate of node-oriented protocols degrades much faster than the rate of hybrid protocols.

A.2 Number of malware infected nodes

Success rates of amplification protocols for an increasing number of malware infected nodes (parameter VII) are shown in Figure 8. A drop in the success rate per one additional infected node is about 1-2% for hybrid designed protocols and 0.5-1.5% for node-oriented protocols in average. The success rate of all protocols decreases linearly. Comparing differences among particular amplification protocols success rates, we see that the higher number of infected nodes, the smaller the difference in performance of protocols.

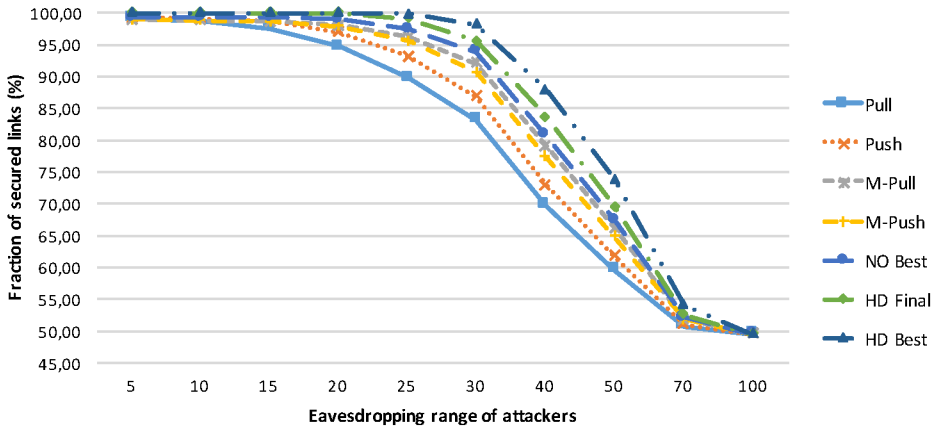

Figure 7: Success rate of amplification protocols for different eavesdropping range of attackers. The initial compromise rate is 50% of all link keys.

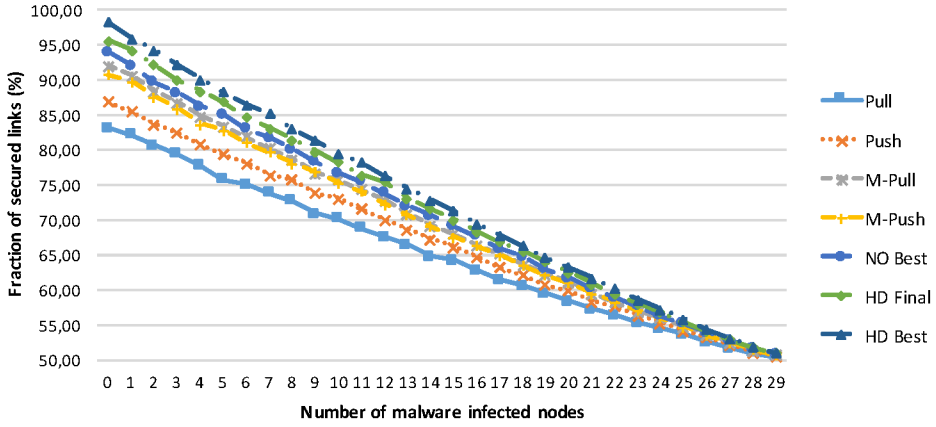

Figure 8: Success rate of amplification protocols for different number of malware infected nodes. The initial compromise rate is 50% of all link keys.

Appendix B Protocol description

We are providing a detailed description of all inspected amplification protocols. For additional information, you can also refer to [1, 3, 13, 16].

Pull protocol: Sensor node C selects two destination nodes A and B from its neighbours. A unique nonce is generated for all pairs of nodes A and B. Node C encrypts the nonce with respective link keys, resulting in two different messages containing the same nonce. The nonce is later used to update a key between nodes A and B.

Push protocol: Sensor node A selects a destination node B and an intermediate node C from its neighbours. A unique nonce is generated for all combinations of destination and intermediate nodes. Node A sends generated nonce via node C to node B, encrypted with existing link keys. The nonce is used to update a key between nodes A and B. This algorithm results in doubled number of generated nonces and messages sent compared to the Pull protocol.

Multi-hop Pull protocol: Sensor node C selects two destination nodes A and B, and one intermediate node D from its neighbours. A unique nonce is generated for all combinations of nodes A, B and D. The nonce is sent directly to node A and via node D to node B. The nonce is used to update a key between nodes A and B.

Multi-hop Push protocol: Sensor node A selects a destination node B and two intermediate nodes C and D from its neighbours. A unique nonce is generated for all combinations of nodes B, C and D. The nonce is sent to node D via node C. Finally, node D forwards the nonce to the destination node B. The nonce is used to update a key between nodes A and B.

NO Best protocol: The best node-oriented protocol is a combination of Push and Multi-hop Pull protocols.

HD Final protocol: The protocol is a variant of the Push protocol, where only a defined subset of intermediate nodes is used to forward a nonce. Sensor node A successively selects a destination node B from its neighbours. For every node B, two intermediate nodes C_1 and C_2 are selected from common neighbours of A and B. A selection is done based on relative distances from A and B. Those distances were determined by an extensive research published in [13]. Node A sends two generated nonces to node B, first via node C_1 and the second one via node C_2. Nonces are used to update a key between nodes A and B.

HD Best protocol: The best hybrid designed protocol is variant of the HD Final protocol. The only difference is that five intermediate nodes C_1-C_5 are selected instead of two.