Reconsidering Attacker Models in Ad-hoc Networks

Radim Ošťádal, Petr Švenda, and Vashek Matyáš
ostadal@mail.muni.cz {svenda, matyas}@fi.muni.cz

Masaryk University, Brno, Czech Republic

Abstract. Our paper aims to move the research of secrecy amplification protocols for general ad-hoc networks to more realistic scenarios, conditions and attacker capabilities. The extension of the current attacker models is necessary, including the differentiation based on types of attacker's manipulation with a node, monitoring capabilities and movement strategies. We also aim to propose suitable secrecy amplification protocols that can reflect the new attacker models in different examined scenarios, utilising genetic programming and manual post-processing.

Keywords: ad-hoc networks, attacker models, genetic programming, secrecy amplification, wireless sensor networks.

1 Background

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 global passive attacker is able to monitor all communication of the entire network. Monitoring might include the initial exchange of the keying material in an open form. The active global attacker comes from the classic Needham-Schroeder model [7]. She is able to alter and copy any message, replay messages or inject any forged material. She might drop part of the communication at her will. The node-compromise model [3] assumes that the attacker is able to capture a fraction of deployed nodes and to extract all keying material from a captured nodes. No tamper resistance of nodes is assumed because of their low production cost. The weakened attacker model was defined in [2]. In this model, an attacker is able to monitor only a small proportion of

the communications within a network during the deployment phase. Once the key exchange is complete, she is able to monitor all communication at will.

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 (SA) protocol after the initial key establishment protocol. SA protocols were shown to be very effective, yet for the price of a significant communication overhead. The overall aim is to provide SA 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 SA protocols were studied – node-oriented protocols, group-oriented protocols and hybrid-design protocols. We provide the basic comparison regarding the overall success rate and a number of sent messages in Figure 1 and Figure 2.

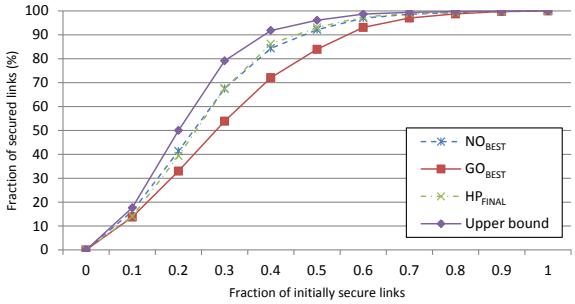

Fig. 1: An increase in the number of secured links after secrecy amplification protocols in the random compromise pattern. The best performing node-oriented protocol [14] is denoted as NO_{BEST}. The best performing group-oriented protocol [10] is denoted as GO_{BEST}. The best hybrid protocol [8] is denoted as HP_{BEST} and its optimised version as HP_{FINAL}. As can be seen, a strong majority of secure links ($> 90\%$) can be obtained even when the initial network had one half of compromised links.

Genetic programming was utilised to discover the best known node-oriented protocol so far, presented in [14]. Evolution was also the primary tool for a proposal of new kind of group-oriented SA protocols. This example might illustrate the fact that even when the evolved solution achieves good results, there might be other practical issues limiting the usability of the outcome. Group-oriented protocols suffer from the complicated synchronization of parallel executions and also from a complex security analysis due to the high number of nodes involved. Such complexities limits a practical use of group-oriented algorithms. For the hybrid-design solution, the genetic programming was used together with manual post processing. The whole process is described in [8]. In the same way, we would

like to develop suitable protocols to counter the new classes of attackers with different capabilities.

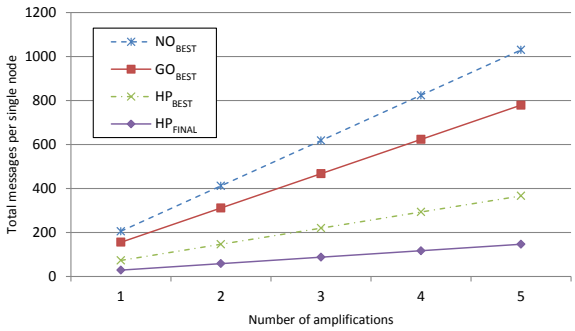

Fig. 2: Total number of messages per single node required to perform the best node-oriented and the best group-oriented secrecy amplification protocols (7.5 neighbours on average assumed). The hybrid protocol even with five amplifications (repetitions) sends considerably less messages than node or group oriented protocols with a single execution.

2 New attacker models

Previous research mostly expected the global passive attacker together with the node-compromise attacker model, sometimes weakened to the real world attacker model. We define the following new classes of attackers, differentiating the node-compromise attacker model from the following cases based on different kinds of manipulation with a node:

Key exfiltration model: The attacker is able to extract a part of the keying material from the compromised node (might be even all keys). After the key exfiltration, the node continues working in the original – uncompromised – manner.

Passive node control model: The attacker compromises the node, extracts all keying material and installs her malware. The node is under attacker's control, but the control remains passive – besides the monitoring purpose, the malware does not affect any behaviour of the node.

Active node control model: The attacker compromises the node in the same way as described in the previous point, but the attacker actively influences the behaviour of the node. She could discard some messages, change their content or even generate new messages at her will. In the context of the secrecy amplification protocols and link key security, she is able to influence steps of a particular protocol, initiate new amplification or attempt to make

the protocol fail completely. Another example could be the disruption of an SA protocol by manipulating with relative distances from parties relevant in group-oriented and hybrid protocols.

According to the attacker capabilities, we distinguish global and local attackers. A global attacker was described in the previous section, and its passive version was used in multiple pieces of research. As far as we know, the local attacker was never considered regarding the link key security and SA protocols.

Local attacker with limited eavesdropping capabilities. This one might be even split into several subcategories. The examples of influencing parameters are the range that the attacker is able to monitor (e.g., the three times the range of legal node) and speed of the attacker movement. We could also consider several local attackers and their possible cooperation.

Besides the extension of the studied attacker model, our primary objective is to provide suitable secrecy amplification protocols that can counter different attacker models in various scenarios.

3 New secrecy amplification protocols

We do not expect the existing secrecy amplification protocols to perform perfectly when new attacker models and the new attacker strategies are assumed. We would like to employ the genetic programming to develop suitable SA protocols and also the manual post-processing to identify the similarities among protocols to construct the SA protocol suitable for most of the scenarios.

We also consider the new way how to evaluate the success rate of SA protocols, in other words, how to compute the fitness function for genetic programming. Only the fraction of non-compromised link keys was used so far. We present the additional views:

- Percentage of secure communication. Legal nodes periodically communicate with their neighbours. The fraction of communication that is not eavesdropped by the attacker is used to evaluate the success of an SA protocol.
- Compromising ratio of messages that are sent from the node to the base station. Nodes emit those messages at regular intervals.

We use the KMSforWSN framework for the simulation of different parametrizable attackers and the evaluation of SA protocols. The KMSforWSN framework was introduced in [4]. It is a tool for an automated evaluation of KMS properties in WSNs built on top of MiXiM [6], a WSN framework for the OMNeT++ simulator [13].

We extended the architecture with two new modules to reflect the different attacker models and also to implement the secrecy amplification capability. The overall changes to the architecture are necessary as the original purpose covered only the key establishment protocols as well as different approaches for the success rate measurements.

We will also use the optimization framework developed originally for the evolution of intrusion detection systems in WSNs [11]. The framework is prepared to work together with the OMNeT++ environment and is also capable of distributing the tasks to BOINC, the distributed computing platform [1]. We expect to use BOINC on tens of CPUs to evaluate several candidate solutions in parallel.

4 Parametrisable attacker and experimental results

The main decisions to define a particular attacker are to select the attacker type and attacker capabilities (global or local). Even after this, we still can define the number of parameters for such an attacker. Those include but are not limited to:

- Initial compromise pattern regarding the attacker movement strategies. Several patterns were defined in [5] – random attacker strategy, outermost attacker strategy, direct centre attacker strategy and centre drop attacker strategy. Additional movement strategies will be defined.
- Number of local attackers. Several local attackers might work together. Collaboration could be only in the exchange of compromised keys, but also, the coordinated movement strategies have to be considered.
- Eavesdropping range is radius where the local attackers are able to intercept the communication, in meters (e.g., the three times the range of legal node).
- Initial location of attackers might be selected randomly or predefined (e.g., at the boundary of the network [0,0]). There is also a possibility for a cooperation of several attackers.
- Movement patterns of attacker during the execution of a SA protocol. Those range from simple random walk up to coordinated patrolling.
- Movement speed of attackers in meters per the second unit.

The experiment is performed and evaluated on a network with 100 nodes randomly distributed on a playground of 115 m x 115 m. Definition of the channel properties and a physical layer setting are based on measurement done for TelosB motes in outside environment, available in [12]. All result are the average of ten random executions. The average density of the network is 7.34 neighbours per node. We use the node-oriented protocols for the comparison of different attackers. Detail evaluation of Pull, Push, Multi-hop Pull (M-Pull), Multi-hop Push (M-Push) and Best NO could be found in [9].

For the first experimental comparison of different attackers, we chose the key exfiltration model. Within this model, we have two cases: 1) Random keys are compromised – this corresponds to previously inspected random key compromise pattern. 2) Random nodes are compromised – all link keys from the compromised node are exfiltrated. Regarding the attacker capabilities, we compare the global attacker and local one with following parameters: 1, 3 and 5 cooperating attackers, eavesdropping range of 30 meters, the initial position of all attackers on coordinates [0, 0], random movement pattern and speed of 5

meters per second. The initial compromise rate of the network is 50% of all link keys, and the process of sending all nonces takes 100 seconds. Every attacker walks randomly 500 meters in total.

The results are summarised in Table 1. The 100 seconds assigned for SA protocol to distribute nonces are not sufficient for the four-party protocols (Multi-hop Pull, Multi-hop Push and Best NO). The nonce packet loss ratio increases up to 12 percent. Nevertheless, it influences the success ratio only slightly due to high redundancy of protocols. Amplification protocols achieve better results for the random key compromise pattern than for the random node compromise pattern in general. The concentration of compromised links around particular node makes it harder to re-secure such links. Multi-hop protocols together with Best NO achieve quite constant success rate for local attackers, regardless of the random key or random node compromising. Again, the reason is high redundancy, compare the number of messages of Pull protocol and its multi-hop version. Push protocol is the best one for both compromise patterns for a local attacker considering the negligible difference in success rate compared to Best NO and number of messages they send.

The Push protocol gives significantly better results than the Pull protocol for both random key and random node compromise patterns. The Push protocol is better probably due to a particular protocol implementation and the timing for nonce distribution. The Push protocol initiates the protocol in a randomly generated time (0-100 seconds), and the intermediate node resends the message immediately. To the contrary, the Pull protocol generates the two messages with the same nonce, and every message is sent in a different randomly generated time (again 0-100 seconds). In the second case, the local attacker has a higher probability of intercepting at least one message.

Protocol	Pull	Push	M-Pull	M-Push	Best NO
Original compromise ratio	50.00	50.00	50.00	50.00	50.00
Messages sent per node	55.35	88.93	1074.33	811.01	1158.26
Nonce loss ratio	00.47	00.69	10.95	07.47	12.17
Random key compromise					
Local attacker (1)	97.86	99.17	98.63	98.76	99.23
Local attacker (3)	94.96	97.86	97.78	98.01	98.94
Local attacker (5)	93.08	96.12	96.80	97.20	98.19
Global attacker	84.17	84.41	89.22	89.34	92.42
Random node compromise					
Local attacker (1)	97.35	99.03	98.32	98.65	99.26
Local attacker (3)	87.03	96.03	96.41	97.47	98.40
Local attacker (5)	79.77	90.50	93.26	94.83	95.68
Global attacker	50.00	50.00	50.00	50.00	50.00

Table 1: Success ratios for attackers with different properties.

The experiment runtime varies according to the complexity of the protocol. The Pull and Push protocols are simulated in one minute; their multi-hop versions take eight minutes and the most complex protocol Best NO is simulated in ten minutes. All measurement are done on a double core CPU @ 2.4 GHz. Optimisation of the source code will be necessary to run the genetic evolution as thousands of generations (multiplied by a simulation runtime of a single scenario) will be required.

5 Conclusions and future work

Our goal in this work was to initiate a discussion about realistic attacker capabilities and behaviour. We showed a large difference between the previously assumed random key compromise pattern and the more realistic attacker with the random node compromise pattern. Our future work will focus on parametrisation of attacker behaviour and her capabilities as presented in Section 4.

We also present preliminary results for other two attacker models. Figure 3 shows partial results for the node compromise attacker together with the passive node control model. The active node control model requires a further inspection. The basic results of this model showed that the attacker is not able to improve her success by neither the manipulation with a nonce message nor by dropping the entire message.

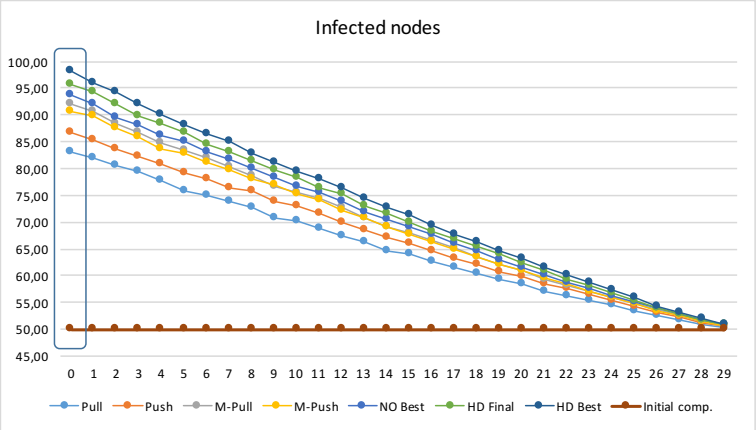

Fig. 3: Success rate of SA protocols for a different number of malware infected nodes. A decrease in percentage of secured links is linear. One can obtain reasonably secure network (more than 85% of secure links) even in case of 7 malware infected nodes considering the hybrid designed protocols are used.

Other results suggest that the most beneficial strategy for an attacker is to stay in one place and do not move at all. The attacker is then able to eavesdrop all communication in a particular area maintaining maximum of previously

compromised links. In a real network, a secrecy amplification protocol is not executed instantly as nodes need to synchronize send and receive multiple messages (another example of the difference between simplistic and realistic simulation). The moving attacker will, therefore, change her physical position relatively to start of a protocol execution, resulting in an ability to eavesdrop transmissions between a different set of nodes. As secrecy amplification protocol is composed of multiple (and often functionally redundant) steps, moving attacker may initially be able to prevent a change of particular compromised link into a secure one (when a local attack can still overhear and compromise newly transmitted key shares). But may fail to do so few seconds later, when the remaining steps of protocol are executed and a moving attacker is already out of reception range for these nodes. This is a surprising result that requires further verification. However, if confirmed, we would need to think over the way of protocol evaluation. Several other methods were suggested in Section 3. For a realistic simulation, the definition of standard network operations and message flow during a network lifetime will be required. Those are areas for future work.

References

1. D.P. Anderson. BOINC: A system for public-resource computing and storage. In *Grid Computing, 2004. Proceedings. Fifth IEEE/ACM International Workshop on*, pages 4–10. IEEE, 2004.
2. 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.
3. 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.
4. 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.
5. Filip Jurnečka, Martin Stehlík, and Vashek Matyáš. On node capturing attacker strategies. In *Security Protocols XXII*, pages 300–315. Springer, 2014.
6. Andreas Köpke, Michael Swigulski, Karl Wessel, Daniel Willkomm, PT Haneveld, Tom EV Parker, Otto W Visser, Hermann S Lichte, and Stefan Valentin. 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 (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.
7. 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.
8. 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, 2014.
9. Radim Ošťádal, Petr Švenda, and Václav Matyáš. On secrecy amplification protocols. In *9th International Conference on Information Security Theory and Practice, WISTP 2015, LNCS 9311*, pages 3–19. Springer, 2015.

10. Tobiáš Smolka, Petr Švenda, Lukáš Sekanina, and Vashek Matyáš. Evolutionary design of message efficient secrecy amplification protocols. In *12th European Conference on Genetic Programming*, pages 194–205, 2012.
11. Martin Stehlik, Adam Saleh, Andriy Stetsko, and Vashek Matyas. Multi-objective optimization of intrusion detection systems for wireless sensor networks. In *Advances in Artificial Life, ECAL*, volume 12, pages 569–576, 2013.
12. 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.
13. András Varga. Using the OMNeT++ discrete event simulation system in education. *Education, IEEE Transactions on*, 42(4):11–pp, 1999.
14. 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.