Universität Zürich[UZH]

Assignment — Communication Systems Group, Prof. Dr. Burkhard Stiller

Guideline for Mapping Security Solutions of Wireless Sensor Networks to Security Fundamentals

Stefan Mussato
Zürich, Switzerland
Student ID: 09-752-270

Supervisor: Corinna Schmitt, Thomas Bocek
Date of Submission: September 29, 2014

University of Zurich
Department of Informatics (IFI)
Binzmühlestrasse 14, CH—8050 Zürich, Switzerland

ifi

Assignment
Communication Systems Group
Department of Informatics (IFI)
University of Zurich
Binzmühlestrasse 14, CH—8050 Zürich, Switzerland
URL: http://www.csg.uzh.ch/

1 Zusammenfassung

Mit der steigenden Verbreitung von kabellosen Technologien und Geräten werden auch deren Anwendungsmöglichkeiten breiter. Dies nicht zuletzt auch deshalb, weil verfügbare Hard- und Software heute Lösungen ermöglichen, die noch vor Jahren utopisch anmuteten. Eine Kategorie solcher Anwendungen wird heute unter dem Begriff ‚Wireless Sensor Networks - WSN' zusammengefasst. Den allermeisten dieser Anwendungen gemeinsam sind die hohen Anforderungen an die Hardware bedingt durch die stark eingeschränkten Ressourcen der einzelnen Netzwerkknoten aber auch die Erfordernisse in Bezug auf Datensicherheit.

Diese Arbeit befasst sich mit der Erarbeitung von einfachen Richtlinien für die Auswahl von geeigneten Sicherheitsprotokollen bei der Planung eines WSN. Dazu werden bekannte und oft Zitierte WSN-Protokolle auf einer Matrix den verschiedenen Sicherheitsanforderungen gegenübergestellt. Anhand eines Beispiels einer WSN-Lösung für die Gebäude-Zutrittskontrolle werden die erarbeiteten Richtlinien angewendet.

Das Resultat der Analyse bestehender Sicherheitslösungen zeigt, dass die starken Ressourcenbeschränkungen eines WSN-Knotens zusammen mit den oft rechenintensiven Sicherheitsmassnahmen eine möglichst klare Priorisierung der Sicherheitsanforderungen notwendig machen, damit geeignete Protokolle ausgewählt und implementiert werden können.

Die Verweise auf weiterführende Arbeiten zeigen unter anderem, wie insbesondere Anwendungen für WSN's im Bereich des ‚IoT - Internet of Things' also dem Internet der Dinge eine dieses Thema in eine neue Ära bringen bezüglich deren Anwendungsvielfalt.

2 Abstract

Starting with a short introduction of security fundamentals and know forms of attacks to Wireless Sensor Networks (WSN) this work puts known security protocols into a matrix with the different security requirements they address. The protocol we selected are DSS/FHSS, TinySec, LEDS, MASA, VEBEK, LEAP, Minisec, DTLS, SPINS and ZigBee. Attacks are then linked to the different communication layers commonly applied to Wireless Sensor Networks. Then, a selection of most cited and well-known protocols for Wireless Sensor Networks are described in a few words with their specific characteristics. Proceeding to the final aim of this document, a guideline is given for the selection of appropriate security solutions when planning a new Wireless Sensor Network. In a following chapter the developed guideline is used to support the selection of an exemplary WSN application for a building access control system.

Table of Content

3 Introduction

3.1 Motivation

A lot of research has been done over the past years in the field of developing suitable solutions for Wireless Sensor Networks (WSN). This interesting field gets a lot of attention not only because of its wide area of potential applications ranging from military to commercial and research, but because it challenges engineers and researchers from different fields and with different focus. Physical constraints, such as operation under harsh environmental conditions and very limited energy resources attract the attention of computer hardware and sensor developers. Limitations in computational power challenge software engineers specializing on implementing lightweight operating systems and others concerned in their daily work with developing suitable security solutions to protect and secure communication on a wireless transport media. Current solutions for WSN are available on a variety of different hardware platforms with significant differences in computational power and memory. Depending on the selected hardware and the specific security requirements of each potential WSN application, the choice of best-fit security algorithms may vary pretty much.

3.2 Description of Work

Starting with a characterization of most often cited security fundamentals, this work attempts to summarize currently available security algorithms and solutions and match them in a matrix with security requirements or fundamentals. The final aim is to support the implementation of the concept known as privacy by design. To do so, this work tries to provide advice on what path to follow and which algorithms to discard when tasked with implementing security into a specific WSN solution. A short description of commonly known potential attacks and threads to WSN‘s is included in order to better introduce the peculiarities of each listed security algorithm. The two-dimensional grid is then explained in a more detailed way by giving pros and cons of each algorithm. Then final chapter of this work adds a concrete real world example of a potential WSN solution for access control in building automation and demonstrates how the given matrix and explanations may support decision making for selecting the most appropriate security solution.

Yong Wang et al. classify security mechanisms used in WSN's into five categories [1]: cryptography, key management, secure routing, secure data aggregation and intrusion detection. We did not focus on intrusion detection in this present short work. Wang's paper provides a good overview and addition to this text. A lot of research papers and conferences summaries are available that discuss requirements on WSN's for different types of dedicated applications. But one of the most important applications for WSN's lies in applications for the 'Internet of Things'. This is because of the mere unlimited number of devices that may become 'inter-networked' in near future. As further reading, Dr. C. Schmitt‘s work [17] addresses this highly interesting field of future WSN usage by showing concepts for adaptation of existing IPFIX protocol and giving details about a possible adaptation named TinyIPFIX. T. Aura explains another very interesting security mechanism in his paper about DoS-resistant Authentication with Client Puzzles [9]. DoS attacks may be orchestrated at different levels of communication. Client Puzzles described in Aura's work show an interesting technique to counter DoS attacks attempting to exhaust a node's resources. This technique may be interesting even outside of the specific requirements of a WSN.

4 Security Fundamentals

The NIST - National Institute of Standards and Technology describes Computer Security as: " The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability and confidentiality of information system resources." [3]. The definition only lists three basic security requirements. Nowadays, the number of papers discussing security fundamentals or requirements is enormous. The majority of papers almost always mention seven basic security requirements as a common denominator, including those stated in the NIST handbook.

4.1 Description of Security Fundamentals

According to the focus of this paper, where we want to match security solutions in WSN to security requirements they support, we will also consider additional requirements in our matrix and list and explain even som of the less cited requirements, just to attempt to give a full overview of existing security fundamentals or requirements.

4.1.1 Basic Fundamentals

We decided to specify the following security fundamentals as being basic ones because we found them in almost every listing of fundamentals. This does not mean that others listed under extended or less cited are not as important, but it seems that the basics are addressed in almost every WSN application. [1], [2], [3], [5]

AVAILABILITY

This requirement is not only about safeguarding against potential attacks such as denial of service making the sensor network or at least part of it unavailable, but it includes the requirements to assure that "the system works promptly" [3], meaning that network nodes must be supplied with sufficient energy and ready to reply within a suitable timeframe.

AUTHENTICATION

It is a general requirement for all kind of data communications to be able to verify and confirm that the origin of the data is a known and accepted source. This applies to parts of communication setup as well as to the payload data sent once a communication channel is established between two or more partners.

CONFIDENTIALITY

There are many reasons why data being exchanged needs to be kept secret. Normally, data confidentiality is achieved by using encryption techniques we will discuss later. Although it is a basic security fundamental, its implementation is not always of highest importance; there are situations and applications, where keeping data information secret is not a mandatory requirement.

INTEGRITY

Different attacks try to manipulate data in transfer. Furthermore communication channels may not be the most reliable ones and thus loss of partial data or data corruption may occur. In both cases data integrity has to be achieved by applying suitable protocols and solutions such as message authentication (MAC) procedure that appends an authenticity key to the message sent.

AUTHORIZATION

Describes the process of checking whether a user, node or entity in the network requesting to perform a specific operation is authorized to do so. Authorization often follows authentication since in a logical process flow, granting specific rights to entities requires previous identification (authentication) of the requesting entity.

NONREPUDIATION

Denotes a service that is assuring the integrity and origin by relying on a third party for verification. Non-repudiation in WSN means that an entity cannot deny having sent a previous message or having executed a specific operation.

FRESHNESS

Guarantees that only new data is being sent and data from attackers replying old messages is identified and filtered. Commonly a time stamp or a data packet counter is added to the data to implement this requirement.

4.1.2 Extended Fundamentals

In addition to the basic fundamentals, the following three extended ones have been cited in different papers, but often only if their specific task was relevant to the individual work of the paper. To give an example: depending on the security priorities of a specific WSN, it may or may not be necessary to implement a secure localization scheme. Therefore these fundamentals are summarized as extensions to the more cited basic ones. [2], [7]

SELF-ORGANIZATION

Implementing single sensor nodes in a way that allows them to re-organize themselves after changes such as unavailability of a node or nodes happened to the network or to part of it. This is of course still an important capability, but we listed it as extended fundamentals and not as basic one because this capability, especially when it comes to selecting suitable key distribution schemes, is not as important as others, but because it may be interpreted more as a general requirement for WSNs and it is not strictly linked to security requirements.

SECURE LOCALIZATION

Different techniques used to protect against attacks rely or require accurate sensor location information. If the information about the sensor position is not accurate, attacker will have higher chances of succeeding in disturbing or breaking the WSN. Again this requirement is seen as not strictly related to security concerns, because there may exist alternative security solutions to protect against same kind of attacks that do not necessarily need location information.

TIME SYNCHRONIZATION

Same as for secure localization requirement, some security techniques need reliable time information to be able to work. Furthermore, applications themselves may need a synchronized time stamp to allow correct interpretation of the data being transferred. There do exist secure synchronization protocols that can be used to implement this requirement.

4.1.3 Less Cited Fundamentals

To complete the list of identified fundamentals some very special ones have been mentioned and discussed in single papers. On some of them it could be argued whether they are real security fundamentals or just additional requirements to specific WSN applications. For example the scalability fundamental cited in [1], [2] and [4] may not be a real requirement in a static structures of a WSN [1], [2], [4], [6]

FORWARD SECRECY

If a sensor, node or entity, of a WSN leaves the network, it should be granted that it would no longer be able to decrypt data and messages that will be sent over the network in future.

BACKWARD SECRECY

When new sensors or entities join a network, they should prevented from being able to decrypt previously sent messages.

ACCESS CONTROL

Being similar to the authorization requirement, it regulates and limits the access to host systems. Access control requires previous identification (authentication) of an entity.

SCALABILITY

With the size of a WSN growing in number of nodes, its overall security should not be compromised by the choices made e.g. for key management and it should not add or increase the communication overhead. This requirement also explicitly mentions that new nodes must be allowed to join a network after the first deployment.

5 Type of Attacks

5.1 Categorization of Known Types of Attacks or Threads

Attacks can be divided in outsider versus insider attacks, in passive versus active and in mote-class versus laptop-class attacks. In- or outsider specifies whether the attackers use an existing, legitimate node of the network to manipulate the system (insider) or whether the attack is done using an external, malicious node (outsider). Active attacks attempt to modify data stream, passive ones mainly listen to the data traffic sent over the network. Last, a most-class attack uses hardware similar to the one of a standard network node with its given constraints but with probably better level of camouflage. Using a laptop's (or any other computers) much higher computing power, an attacker has much more options for different types of attacks. Some forms of known attacks require high computational and transmission power anyway to be orchestrated.

If we briefly map security requirements to their general techniques used to implement them, we can summarize that confidentiality is achieved using appropriate encryption mechanisms, integrity requires different MAC algorithms, the access control fundamental is implemented using access control tables and authentication processes use authentication protocols. Non-repudiation is assured using public key infrastructure with asymmetric keys. Freshness finally requires some kind of time stamp and clock synchronization or packet counter value. [1]

5.2 Mapping of Threads to OSI Level(s)

Attacks can be orchestrated at different levels of OSI protocol stack. Here we list different known types of attacks to each of the five typical layers of a WSN. [1], [8]

5.2.1 Physical Layer:

Tasks such as frequency selection, modulation and signal deflection are handled at the lowest OSI layer. Tampering or DoS attacks such as radio jamming are relatively simple to execute and they effectively compromise a WSN's function. Most WSN nodes do not even attempt to secure it from tampering attacks because this would imply high costs to them. They generally try to secure sensitive information by appropriate security schemes that protect from reverse engineering the firmware or parts of it. Countermeasures to jamming on lowest layer include multi frequency hopping spread spectrum technique or Ultra Wide Band transmission.

5.2.2 Data Link Layer and MAC:

Link layer provides multiplexing of data, data frame detection, medium access and error control.

When choosing a suitable protocol for this low layer, it is important to be aware of the expected network load of the planned WSN. If a low load can be assumed, collision-handling schemes will be assigned lower priority then with assumed high bandwidth utilization. [10]

By continuously sending data on the given transmission channel an attacker may attempt to cause collisions. In a more sophisticated attack collisions may be triggered on specific packets such as ACK (acknowledgment) control messages. As a result, collision-handling schemes such as the exponential back off will lead to very low transmission rates or no transmission at all. With very simple collision-handling schemes this may finally lead to power exhaustion of single or multiple nodes. As a week form of DoS attacks, the unfairness attack periodically disturbs data transmission on low OSI layers. This seriously degrades performance of real-time applications. SMACS - Self-Organized Medium Access Control for Sensor Networks and EARS - Eavesdrop and Register are known and commonly applied data link layer protocols tackling these threads.

5.2.3 Network Layer:

Upper layer to the link layer includes packet forwarding logic and algorithms, address assignment, routing path finding and it is responsible for tracking device locations. Location awareness may be an important feature of a WSN to achieve specific requirements.

Routing information captured on network layer level may be altered or replayed. With this kind of common network layer attacks, routing loops are created or the network may be partitioned thus finally resulting in longer latency times of end to end data communication. Using suitable authentication mechanisms, routers in the network will be able to identify invalid senders and only accept routing information from valid (authenticated) sources. Selective forwarding attacks may attempt to intrude the network and then drop certain messages that were faithfully delivered to the compromised node. Adding sequence numbers of packets may allow the receiver to detect such attacks. In another type of network layer attack; the sinkhole attack, acompromised node will try to channel all data transfer through its own node and then drop packets. If neighboring nodes use paired keys and spread spectrum communication, the attack may no longer be successful, but with protocols using advertising of information, which are widely used in WSNs, the sinkhole attack is hard to overcome. The Sybil attack tries to create fake identities within the network where node IDs are assumed to be unique. It causes severe disturbance for protocols using geographic routing. Therefore the management of node identity is a key factor in defense against the Sybil attack. Another sort of network layer thread is known as the wormhole attack. By recording routing request packets and replaying them from a location far away from the original source, a wormhole can be created in a WSN. This lets other nodes assume that they are a neighboring node of a node that's far away in reality. Wormholes significantly disturb routing procedures in WSNs because their routing is often based on information broadcasts and dynamically adjusts data packet routing. Possible solutions to prevent wormholes is to use geographic based routing schemes such as GPSR - Greedy Perimeter Stateless Routing or GEAR - Geographic and Energy Aware Routing protocol. Hello Flood or generally flooding attacks on the network layer level take advantage of commonly used node discovery broadcasts (HELLO packets), by making nodes believe to be providing shortest routing paths. These attacks are similar to wormholes and countermeasures include verification of bi-directionality of routings or authentication of broadcasted 'hello' packets. TESLA - Time Efficient Stream Loss-tolerant Authentication is such an example of a protocol that provides authentication of initial packets. In attacks spoofing the acknowledgements on link layer level, which is used by many WSN protocols, the attacker convinces a node that a weak link is strong or that a dead node is still alive thus making the node send its data to those 'bad' nodes. Data loss will occur. Prevention of spoofing attacks include encryption and authentication protocols. Finally black hole attacks targeting network layer may attempt to attract all traffic to a compromised node and then may or may not drop packets it receives. This attack not always attempts to simply disrupt normal operation in a WSN, but it may be used to open doors for further types of attacks. A often cited protocol to counter black hole attacks is REWARD - receive, watch, redirect. The protocol is able to detect and then subsequently broadcast information on a compromised node to the network.

5.2.4 Transport Layer:

This layer takes care of assuring reliable transport of data packets sent. It manages end-to-end connections, takes care of data flow control and provides multiplexing.

We already mentioned flooding attacks on the discussion of the lower network layers. Flooding on transport layer attempts to send - flood- a node‘s receiver buffer with large amount of data packets in order to cause data buffer memory to overflow and then making the node unable to receive any further data packets from legitimate nodes. One proposed defense against transport layer flooding is that of implementing a 'client puzzle' [9] where a client must commit by putting some up-front effort

into the communication establishment and letting the receiving node verify its commitment in a simple and efficient way before allocating its limited resources to the communication establishment process. De-synchronization is another potential transport layer attack, where timed correctly, spoofed messages cause a host node to ask for retransmission of missed frames even if they were transmitted correctly. Eventually this leads to degradation of data exchange and waist of energy. Authenticating packets together with the header control information is one possible countermeasure to this attack.

5.2.5 Application Layer:

At this level the logic provides information and specification of how data sent to end-users is formatted. It presents required data to the application.

Sending large amount of data to 'overwhelm' a network node causing high bandwidth consumption or repudiation attacks are possible threads to the application layer. Countermeasures at this level are based on cryptography to prevent an attacker from even being able to start communicating with node of the WSN on application layer level.

6 Mapping of Existing Security Solutions

Many of the previously listed fundamentals rely on cryptography. And the level of security granted by cryptography again depends on the procedures and techniques used to secretly exchange cryptographic keys. Therefore one major focus for appropriate selection of protocols will be on key establishment techniques. [2]

6.1 Characterization of Selected Solutions

The figure below gives an overview of the protocols we will be comparing in this work. On the x-axis we mark the year of first presentaion to public. The y-axis represents the layer in which the single protocols operate. End the size of the bubbles represents the number of fundamentaly they address; protocols with smaller bubbles represent three and the bigger ones four fundamentals (see also Mapping Selected Solutions to Fundamentals on page 18).

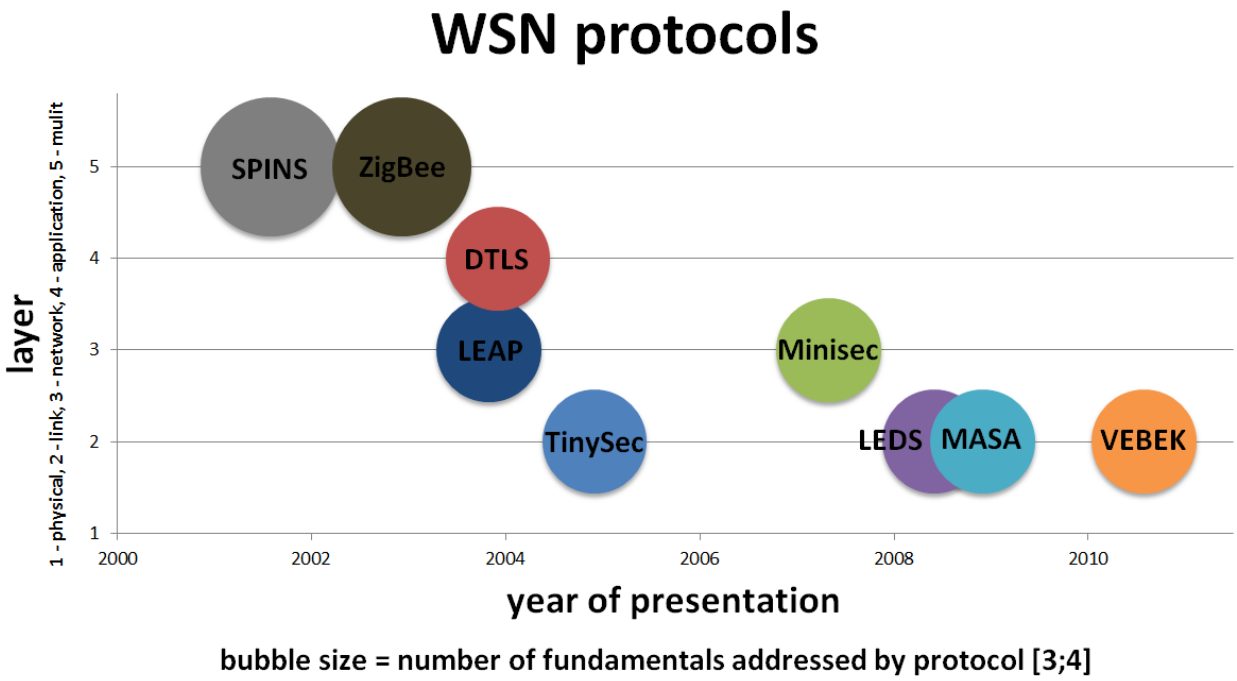

Figure 1: Protocols and year of presentation plus number of fundamentals they address

The above graph does not list the DSSS/FHSS technique located at the physical layer. Both techniques cover one single security fundamental and their presentation date would extend the x-axis to a mulitple of the twelve years represented above. Spread spectrum technique(s) enhance availability of a WSN significantly. FHSS was first mentioned in 1915 and then patented in 1942. DSSS respectively FHSS described below are widely used in different known technologies such as WLAN or Bluethooth. It addresses one single fundamentals availability.

DSSS/FHSS

Countering DoS attacks, or more concrete radio jamming attacks on the physical layer is a difficult task especially for wireless data transmission mediums. Possible techniques include adaptive antenna systems or spread spectrum modulations. Furthermore error correcting codes can help improving data transmission capability even under bad conditions. Reed-Solomon codes are the standard choice for error correction in WSN's [8] A typical WSN node has a simple omnidirectional

antenna because more sophisticated types of antennas would imply higher cost and they would need to be angled to a specific other node to be able to improve data reception. But because a WSN node in a standard application must be able to communicate with multiple nodes, directing its transmission to a single neighbor is not an option. Possible solutions to counter DoS attacks at physical layer are 'DSSS - Direct Sequence Spread Spectrum' or 'FHSS - Frequency Hopping Spread Spectrum'. DSSS is difficult to implement in a WSN because it requires tight time synchronization and uses higher energy when sending data. The DSSS technique is used in IEEE 802.11 (WLAN), GPS or UMTS communication. FHSS occupies less bandwidth than DSSS and is used in the Bluetooth standard [10]. It increases data transmission robustness by switching its carrier frequency according to a pseudo-random sequence only known to a network node and does not require as expensive radio devices as needed for DSSS. Thus FHSS can be used in WSN to counter radio-jamming attacks.

TinySec

TinySec was developed at the University of Berkeley by C. Karlof, N. Sastry and D.Wagner and resented in 2004 as a dedicated protocol addressing WSN constraints such as small memory, low energy consumption and short data packet sizes. It is therefore portable to almost all available WSN hardware platforms. TinySec describes a full solution in all details including packet definitions and interfaces. Karlof et al state that their design choices where driven by WSN capabilities and given realities. The protocol implements three of the basic fundamentals but leaves all others unaddressed. Integrity and message authentication is provided by an included message authentication code with each packet sent. Replay protection would require to include counters to be able to detect if a message is fresh or if it is just a replay of an old one. As this is not seen as a Link layer mechanism by the creators of this protocol, TinySec does not attempt to counter this potential attack at all. Semantic security is an important feature required even in networks with limited resources such as a WSN. Common technique which is also applied in TinySec, is the use of initial vectors to add some randomness to the recurring encryption algorithm. Two options are available when using TinySec; authenticated encryption or authentication only. The encryption is done using cypher block chaining. The block cypher selected as default is Skipjack. According to the analysis done by the developers of TinySec, Skipjack performs better than conventional Triple-DES, AES or even RC5 cyphers. The protocol adds an additional 1 to 5 bytes to the standard TinyOS packet length, depending on wehther encryption is used or not [12]. As the name implies TinySec is based and implemented on TinyOS. With only 400 Bytes of data and instruction memory this operating system is an ideal solution for wireless embedded systems [20]. TinySec comes with the public release of TinyOS. [12]

SPINS

SPINS - Security Protocols for Sensor Networks' is a suite of protocols developed by Perrig, Szewczyk, Wen, Culler and Tygar at Berkeley University and presented at 2001 MOBICOM. It includes ‚SNEP - Secure Network Encryption Protocol' which provides confidentiality, authentication and freshness plus ‚µTESLA - Timed, Efficient, Streaming, Loss-tolerant Authentication' for key establishment. Both, SNEP and µTESLA have been designed as well by the inventors of SPINS. Their prototype implementation of SPINS was done on a TinyOS operating system. The protocol addresses the challenging task of broadcasting authenticated data through the introduction of a loosely synchronized local clock in each node. Standard broadcast authentication techniques use asymmetric encryption and would be too heavy for an implementation in WSN nodes. Using symmetric encryption only though does not allow suitable broadcast authentication security. SPINS introduces asymmetry by a delayed key disclosure plus a one-way function key chain. Data integrity requirement is not addressed directly but is achieved through the implementation of the stronger data authentication property. SNEP itself has a low communication overhead . It achieve a week freshness through the introduction of a counter value. This counter is

not included in the communication, but increased locally at both communication ends upon each data packet transmission. The important semantic security property is also achieved by using the local counter value. This has the advantage of not adding any further transmission overhead to the communication. The authentication of the broadcast is done by the μTESLA part using only symmetric keys. Time synchronization a is a necessary prerequisite. When a node receives a broadcasted data packet, it verifies if the MAC key used by the broadcast station was not yet disclosed. The packet is temporarily stored in the nodes memory until the base station broadcasts the verification key to all receives. If the verification key is identified as being correct, the node can use it to authenticate the buffered data packet. SPINS is a data centric protocol that is highly adequate for WSN application but has limited scalability and a long delay at boot-up of large scale networks due to the large base station initial unicasts. [19]

LEAP

Presented by Sencun Zhu, Sanjeev Setia and Sushil Jajodia from George Mason University Fairfax, VA at the 10th ACM conference on Computer and Communication Security in New York 2003, this dedicated WSN key management protocol addresses different security requirements by using four types of shared keys. The basic idea of the design is guided by the observation that different types of messages within a network require different security measures and therefore a single keying mechanism is not suitable to meet all requirements. The four types of keys used in 'LEAP - Localized Encryption and Authentication Protocol' are one key per node shared with the base station, a pairwise key shared between neighboring nodes, a clustering key shared among all neighbors and one network wide key. One positive effect of these keying structure is that a compromised node will node affect security of the entire network data communication but it will rather be restricted to the local neighborhood of the compromised node itself. The protocol supports in-network communication which enhances its capability to aggregate data or to let single nodes decide whether to report an event or not, given hat it is able to hear a neighboring node's report about the same event. LEAP also uses the previously described μTESLA protocol for authenticating base station broadcasts in order to assure every node is able to identify(authenticate) the base station. In difference to other protocols, LEAP uses a one-way key chain based authentication to counter impersonation attacks. It does this by attaching the next key that will be used to every packet sent to a neighboring node. With LEAProtocol HELLO FLOOD attacks or sinkholes can be prevented thanks to its authentication of local inter-node traffic. Also wormhole attacks can only be executed at a very short, specific time of pairwise key establishment process when a node gets to know its neighbors. [21]

Minisec

Compared to TinySec or SPINS, this protocol developed in 2007 at Carnegie Mellon University in Pittsburgh and University of Maryland College Park by M. Luk and V.Gligor, is even more energy efficient and requires less data overhead. Its packet format is very similar to the one used by TinyOS. The authors claim that Minisec combines high level of offered security from ZigBee implementation with low energy consumption of TinySec solutions. Having two tailored modes of operation for single-source and for multi-source broadcast communication it scales well also to large networks. By using a special block cipher mode, authenticity and secrecy is achieved in a one pass only process. The initial vector is only transmitted partly thus enhancing security as well. The two mode operation optimizes radio energy consumption by using synchronized counters plus extra computation for unicast communication. MiniSec implementation requires a slightly increased memory size if compared to TinySec. [22]

LEDS

The 'Location aware End to end Data Security' protocol provides end-to-end data security. This increases security level and attack resistance significantly if compared with commonly used hop-by-hop communication in other WSN protocol stacks. In 2008, K.Ren, W. Lou and Y.Zhang presented their security protocol for WSN's to the public. One of the novelties of LEDS is that keys used for different security tasks are linked to a node's geographical position. Among others this has the positive effect of limiting a security breach of a compromised node to its close neighborhood. The protocol implements a one-to-many data communication scheme and achieves good performance in countering different types of DoS attacks thanks to this scheme. LEDS implements a new location-aware, flexible key management framework. It virtually divides the geographic area over which the network is deployed into multiple cells. This cell information is then integrated into each nodes symmetric keys. The end-to-end security is achieved by a shared secret key between a node and the base station. Each packet sent to the sink is encrypted with this secret key. Thanks to the one-to-many communication where all neighboring nodes are capable of decrypting a data packet sent by their direct neighbors, even if single nodes are compromised, others will still receive, decrypt and forward a data packet to the sink. This increases availability of data in a wireless network and successfully counters selective forwarding attacks. In terms of data communication overhead LEDS adds twenty-five percent to a standard 36 Byte TinyOS data packets. [23]

MASA

In 2008, H. Alzaid and M. Alfaraj presented the ‚Mixture of asymmetric and symmetric approach'. MASA provides end-to-end data security. Similar to LEDS, this protocol integrates a virtual geographic grid where single nodes are assigned to cells of the grid. Using a private key to sign a hashed event notification in order to provide end-to-end confidentiality, authenticity and data integrity [2]. A data packet signed with the private key of a node is forwarded by the network hop-by-hop but not decrypted until it reaches the sink. Only there the private key is known. The introduction of a list of trusted neighbors and helper nodes provides strong data authenticity. Using the list, a node can decide to which following destination the data shall be addressed. MASA is comparably new development and it takes the improvements done on the typical WSN hardware used today into consideration. The availability of higher computing performance and larger memory space makes public key cryptography a feasible option even on WSN nodes nowadays. Still all heavy decryption operations are left to the sink which is assumed to have enough computation power and energy resources. MASA improves LEDS weaknesses of using broadcasts to send messages and removes LEDS computation overhead required in each node to decrypt end re-encrypt each message before forwarding it. [24]

VEBEK

Using RC4 encryption VEBEK, 'Virtual Energy-Based Encryption and Keying for Wireless Sensor Networks' achieves simple confidentiality. The RC4 key changes dynamically with each message depending on the remaining nodes energy ressources. Other nodes must keep track of the energy level to be able to decrypt the message. The 'energy efficiency' of VEBEK makes it a very interesting choice for WSN applications given that the sending procedure is the most energy consuming part of a WSN's node operation. Depending on the individual scenario, two variants of VEBEK can be implemented; VEBEK-I or VEBEK-II. The difference is that with version one of VEBEK, each node monitors all of its immediate neighbors and with VEBEK-II nodes leading towards the sink are monitored. Introduces in 2001 by A.S. Uluagac, R.A. Beyah, Y. Li and J.A. Copeland, the protocol detects malicious data immediately without wasting too much energy. [25]

DTLS

Just recently in 2013 the lists of existing WSN security protocols was extendedn by N. Madadugu and E. Rescorla with DTLS. Implemented at the application layer level, this adaptation of known Transport Layer Security - TLS protocol inherits the security properties of TLS, the successor of SSL. Features of this high level protocol include ease of use thanks to its usage of known techniques and a two way authentication. It uses RSA public key mechanism and may be implemented in conjunction with standardized IEEE 802.15.4 architectures. Being implemented at high protocol stack, it cannot rely on any lower level security mechanism. The only assumption it does is that packets sent will be received by the sink at one point. So it offers end-to-end security and implements authenticity, integrity as well as confidentiality in a suitable adaptation of schemes and techniques known from the Internet onto constraint devices used in WSN nodes. [26], [27] and [30]

ZigBee

Described as enhancement to the lower layer specification of IEEE 802.15.4 standard, ZigBee is already a well-established industry standard created by an idustry consortium in 2002 for low power, flexible network and security plus an extensive application framework. Similar to DTLS it uses well known standards such as AES-128 scheme on link layer or TLS 1.2 protocol to implement security requirements. On lower physical layer, the IEEE 802.15.4 standard employed DSSS (see also page 13) to increase availability of the network. Extended standards such as the Zigbee IP describe open standard for seamless interconnection with the Internet, offering an end-to-end connection with IPv6 network protocol. ZigBee is an already wide spread and well supported ultra-low power and low throughput technology. [28]

Additional Protocls, Not Listed in Matrix

PIKE

‚Peer Intermediaries for Key Establishment' is a key establishment mechanism that uses a third node as a trusted entity for key exchange. Any two nodes share a common node with a third one. Communication and memory overhead scale in lower order than liner and achieve a higher level of security in takling compromised nodes than other protocols. [29]

TRANS

‚Trust Routing for location Aware Networked Sensors' is another secure routing protocol for data centric networks that is based on μTESLA which adds location aware routing. [13]

GEAR

A variety of different geographical based routing protocols already exist; GPSR is such one, ‚GEAR - Geographical and Energy Aware Routing' is another one. GEAR saves energy and performs better in terms of packet delivery compared to older GPSR. This protocol does not scale to larger networks and it must be predetermined where nodes will be deployed geographically. [11]

Mapping Selected Solutions to Fundamentals

The following matrix should help giving an overview of discussed protocol and security solutions for a WSN application. It does not pretend to be complete at all given the large and growing number of existing protocls. Some solutions focus on single security tasks and others are a combination of different techniques which offer a much wider range of security features. The selection was done in view of the final target of this paper to provide some simple guidelines in selecting appropriate security solutions for WSN's.

The security fundamental 'authorization' was removed from the matrix following below because it is not addressed by the protocols chosen for comparison. Other solutions would be required in order to achieve a secure authorization. Nevertheless, we see authentication as the main prerequisite for even being able to grant or deny authorization. With a good level of security for authentication, it should not be too difficult to add a suitable authorization scheme..

Protocol	**Fundamental**						**Layer**
	Availability	Authentication	Confidentiality	Integrity	Non-repudiation	Freshness	
DSSS/FHSS	X						Physical
Tinysec		X	X	X			Link
LEDS	X	X	X				Link
MASA		X	X	X			Link
VEBEK		X		X	X		Link
LEAP		X	X	X			Network
Minsec		X	X			X	Network
DTLS		X	X	X			Application
SPINS		X	X	X		X	multi layer
ZigBee		X	X	X		X	multi layer

Table 1: matrix with WSN protocols and security fundamentals

With the selection of security protocols done in this paper an attempt is made to provide a balanced mixture of different layered protocols. Although DTLS is listed as the only protocol seated at the application layer, all other lower layer protocols would still allow to add additional high level security mechanisms. DTLS is mentioned because of its many parallels to known standard internet security mechanisms, it appears to be providing a very good basis for future usage in the growing field of 'Internet of Things'.

7 Guideline to Select Appropriate Solution

There are different possible starting points for choosing the most suitable solution when planning a new wireless sensor network. Depending on the prerequisites, the hardware platform to be used in the sensor nodes may be already defined or existing or specific requirements for the given application may be mandatory and thus some possible options for security protocols can already be excluded at this early stage. For example if the application requires good scalability of the WSN being deployed, protocols such as SPINS with limitation in this feature can be excluded. If in another example, we plan a network used to monitor environmental conditions in harsh environments where sensors are deployed at one point and will never be physically access anymore, a node must be able to reorganize itself within the network. Flexible routing mechanisms are important in this case. Also must the network be able to continue to work even if single nodes stop operating. Minisec is one suitable protocol selection for such an application. So one important guidline is to collect all existing prerequisites and identify the main task of the planned WSN in order to do a thight preselction of possible solutions.

Another guideline starts with selected use cases being described for the intended application. By finding matches on how to implement these use cases, certain protocols will result as being more suitable than others. When generating use cases there's a high chance of finding good solutions to address some but not all of them. Therefore use cases must be evaluated and prioritized. In an exemplary use case for a specific WSN application an alarm may need to be triggered to detect a fire in a critical environment. Alarming almost always implies fast, real-time data forwarding and high availability. The obvious choice for possible security implementation following this requirement would be a reliable, protocol such as LEDS eventually even combined with additional techniques to further increase availability and counter potential DoS attacks on different layers. This evaluation starting point is a use case from where necessary prerequisites can be dedeucted and a preselction of protocols can be made.

A third guideline starts from specific security requirements that are absolutely mandatory to implement. Example: if data transferred is highly sensitive, a protocol offering good end-to-end security may be the best choice. If confidentiality is the only real concern, then many options may be considered. But if data freshness is added to the confidentiality requirement, ZigBee implementations would be one of the currently available best choices for this specific WSN. When evaluating security protocols based on security fundamentals, it is important not only to list all security requirements, but also to prioritise them.

Finally other requirements may exist which provide a selection of possible security solutions to implement. One example for additional requirements could be the necessity to seamlessly integrate the WSN to be deployed into existing network infrastructure such as wide spread TCP/IP networks. The 'Internet of Things' is an often heard buzzword nowadays. Different work and research has already been done in developing protocols highly compatible with known solutions and thus easy to integrate. DTLS is one good choice of WSN protocol solution that was developed with the idea of compatibility to existing network infrastructure in mind.

8 Example: Building Access Control Application

If we compare the implementation of new or changed security solutions in an existing building with the upfront planning of a completely new building, more constraints exists in terms of possibilities to interconnect access points to build an appropriate data network. In reality this is a very common scenario. And even if we plan a new building upfront according to current requirements for access control and building security, such a new system often needs to somehow integrate to another existing structure or control system, because the new building may just be built as an extension to an existing complex or has to integrate into an existing access control system within the same company. This is where a Wireless Sensor Network may provide a suitable solution, because it does not require hardwiring of each access point in order to create a data communication network that allows real time monitoring and control of building access.

8.1 Description of Application

In our application, we assume an existing, older building needing to be retrofitted with a modern access control system, partly because its existing system is no longer acceptable from an operational point of view and also because the building will be housing sensitive infrastructure such as laboratory or server rooms in future. Part of the old building is of historical value and it cannot be retrofitted with wired sensors. The targeted application for the WSN is to be able to monitor status of doors or windows almost in real time and even generate alerts to security personnel if any abnormality is observed. Second task of the WSN is to implement online, remote programming of access control. The system should allow to register a new employee or visitor at the front desk and then remotely grant access to all needed parts of the building.

8.2 Challenges and Constraints

The very first constraint we face is that with increasing number of doors, windows and other containers such as safes all requiring controlled access, they cannot be hardwired to become interconnected and remotely controlled in a real time manner no matter whether the building was of special historical value or not, because the costs involved in installing all necessary cables would be enormous. This is where existing solutions on the market provide off-line retrofit cylinders or stand-alone safe locks that can all be programmed and controlled using off-line tools. These programming tools will need to be carried to each point of control at regular intervals. They generally apply time limitations for access, thus these off-line components come with a built in real time clock. Furthermore they implement black and white lists to explicitly accept and explicitly deny access to specific RFID cards, tags or finger print holders. And that is why we may opt for using a secure WSN network to interconnect all our remote access points to finally achieve the target to be able to have online status monitoring plus almost real time access control.

Security is one of the main constraints in such a WSN application. To be even more precise it is the confidentiality requirement that is of utmost importance because we must protect sensitive user information such as codes, card numbers or finger print identities from being stolen and later on replicated by an attacker to get unauthorized access to the facility. Then, also availability and freshness are two additional key issues for this application. Network must be able to guarantee that data is up to data (real time) and no important communication, e.g. such as a duress situation is not being transmitted properly due to un-availability.

Power limitations mainly apply to the control cylinders or locks themselves, but not to other nodes of the wireless solution, because those devices could easily be placed close to an existing power plug

outlet. Still the power limitation constraint very much applies to the last node of the network sitting right at the mechanical component that physically blocks or grants access.

Finally the deployed WSN must be able to forward its collected data and to interact with other devices and with software systems running on computers and network by Ethernet TCP/IP communication.

8.3 Application of Guidelines

Security is of course one of the most important aspects of an access control system. This applies to the physical access as well as to any data communication used in such systems. The two major usages of interconnected access components are monitoring of door and device status, possibly in real-time or close to. The second usage is the remote programming and configuration option. Allowing remotely blocking or enabling access for individual users or during specific times can achieve a significant increase in access control security.

Referring to the guideline given in this paper, we could staart from the use case of a secrutiy officer having to remotely program access for employees. Alternatively we could start with guideline of special security requirement being confidentiality and availability. In both cases we opt for a flexible WSN protocol implementation which grants easy integration into existing software solutions, which already use standard Internet protocols. These requirements would highlight DTLS or ZigBee as possible solutions. Because we must be able to support and add as many different hardware solutions from different manufacturers as possible, we choose ZigBee. Being well standardized and supported by a large industry association it is the best suitable solution for physical building access control applications.

Having said this, we still need to consider higher energy consumption of different ZigBee implementations and will have to keep an eye on this important detail because although this application is a static WSN node deployment and nodes can always be accessed after their deployment, the necessity of a systematic battery exchange more frequently than once a year may make a commercial solution uninteresting for the market.

9 Conclusion

With the Internet of things and with powerful and still low energy devices becoming affordable on a commercial scale, there's a high chance that WSN applications will become very popular in the next years. Still a lot of research and continuous development will be required to provide marketable solutions. Especially when addressing security threats, continuous improvements are needed to keep up in the race with potential attackers. ZigBee and Bluetooth are the two currently available WSN solutions that already set standards for WSN's in the industry. But as the number of WSN applications increase and they begin reaching into our daily lives, additional threads in terms of abuse of data e.g. for observation or surveillance of people become more and more important. As always with new technologies, the discussion of advantages versus threats will have to be continued on a higher level than purely technical feasibility and data security.

10 Bibliography

[1] Yong Wang et al.: A Survey of Security Issues in Wireless Sensor Networks, IEEE Communications Surveys, Q2 2006, volume 8

[2] Usham Robinchandra Singh: A Survey on Wireless Sensor Network Security and its Countermeasures: An Overview, International Journal of Engineering Science Invention ISS, Sept. 2013

[3] Barbara Guttman, Edward A. Roback: NIST Special Publications 800-12, An Introduction to Computer Security: The NIST Handbook, U.S. Department of Commerce, 1995

[4] C. Schmitt, T. Kothmayr, B. Ertl, W. Hu, L. Braun, G. Carle: TinyIPFX: An Efficient Application Protocol For Data Exchange In Cyber Physical Systems, IFI, Unversity of Zürich, Department of Computer Science TU München, Institute for Telecommunication Systems, TU Berlin, CSIRO, Brisbane, 2014

[5] William Stallings: Network Security Essentials: Applications and Standards, 4th edition, Person, ISBN 13: 978-0-13-610805-4

[6] Yan-Xiao Li, Lian-Qin, Qian-Liang: Research On Wireless Sensor Network Security, 2010 International Conference on Computational Intelligence and Security

[7] Jaydip Sen: A Survey on Wireless Sensor Network Security, International Journal of Communication Networks and Information Security (IJCNIS) Vol. 1, No. 2, August 2009

[8] Kalpana Sharma et al.: A Comparative Study of Various Security Approaches Used in Wireless Sensor Networks, International Journal of Advanced Science and Technology, Vol 17, April 2010

[9] Tuomas Aura, Pekka Nikander, Jussipekka Leiwo: DOS-resistant Authentication with Client Puzzles, Helsinki University of Technology, 2000

[10] Holger Karl, Andreas Willig: Protocols and Architectures for Wireless Sensor Networks, Wiley, 2007, ISBN 13 978-0-470-09510-2

[11] W.Ke, T.D.C. Little: Dynamic Routing Selection for Wireless Sensor Networks, MCL Technical Report No. 06-02-2008, Boston University, June 2, 2008

[12] Chris Karlof, Naveen Sastry, David Wagner: TinySec: A Link Layer Security Architecture for Wireless Sensor Networks, SenSys'04, Nov. 3-5, 2004, Baltimore, Maryland, 2004, ACM 1-58113-879-2/04/0011

[13] J.P. Walters, Z. Liang, W. Shi, V. Chaudhary: Wireless Sensor Network Security: A Survey, Dep. of Computer Science, Wayne State University, Auerbach Publications, CRC Press, 2006

[14] R.Panigrahi, K. Sharma, M.K. Ghose: Wireless Sensor Networks - Architecture, Security Requirements, Security Threats and its Countermeasures, Department of Computer Sc. & Engineering, SMIT, Majhitar, Sikkim, India, CS & IT-CSCP 2013

[15] A.Cavoukian: Privacy by Design, http://www.privacybydesign.ca, 2014

[16] Harald Vogt: Protocols for Secure Communication in Wireless Sensor Networks, Diss. ETH Nr. 18174, ETH Zürich, 2009

[17] C. Schmitt: Secure Data Transmission in Wireless Sensor Networks, Dissertation, Institute of Informatics, TU München, 2013

[18] A. Perrig, R. Canetti, J.D. Tygar, D. Song: The TESLA Broadcast Authentication Protocol, CryptoBytes, 5:2, Summer/Fall 2002

[19] A. Perrig, R. Szewczyk, V. Wen, D. Culler, D. Tygar: SPINS: Security Protocols for Sensor Networks, The Seventh Annual International Conference on Mobile Computing and Networking (MobiCom 2001), 2001

[20] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister: System architecture directions for networked sensors, Proceedings of ACM ASPLOS IX, pages 93-104, November 2000.

[21] S. Zhu, S. Setia, S. Jajodia: LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks, CCS '03: Proceedings of the 10th ACM Conference on Computer and Communications Security, New York, USA, 2003

[22] M. Luk, G. Mezzour, A. Perrig, and V. Gligor: Minisec: A Secure Sensor Network Communication Architecture, Proc. Sixth Int. Symp. Information Processing in Sensor Networks (IPSN '07), Apr. 2007

[23] K.Ren, W.Lou, Y.zhang: LEDS: Providing Location Aware End to End Data Security in Wireless Sensor Networks, IEEE Transactions on mobile computing vol. 7(5), May 2008

[24] Alzcuid.H, Alfaraj.M: MASA: End to End Data Security in Sensor Networks Using a Mix of Asymmetric and Symmetric Approaches, IEEE conference mobility and security, vol 25, 2008

[25] Arif Selcuk Uluagac, Yingshu Li: VEBEK: Virtual Energy Based Encryption and Keying for Wireless Sensor Networks, IEEE Transaction on mobile computing vol. 9, (7) July 2010

[26] Thomas Kothmayr, Corinna Schmitt, Wen Hu, Michael Brünig and Georg Carle: A DTLS Based End-To-End Security Architecture for the Internet of Things with Two-Way Authentication, TU München and CISRO ICT Centre Brisbane, 2012

[27] Thomas Kothmayr: A Security Architecture for Wireless Sensor Networks based on DTLS, Master's Thesis in the Software Engineering Elite Graduate Program at the University of Augsburg, 2011

[28] ZigBee Allinace, online at http://www.zigbee.org/Specifications/ZigBee/Overview.aspx, 25.09.14

[29] H. Chan and A. Perrig: Pike: Peer Intermediaries for Key Establishment in Sensor Networks. In IEEE Infocom 2005

[30] N. Modadugu, E. Rescorla: The Design and Implementation of Datagram TLS., Proceedings of NDSS 2004. 2004