Security Considerations for the WirelessHART Protocol

Shahid Raza, Adriaan Slabbert, Thiemo Voigt
Swedish Institute of Computer Science (SICS)
SE-16429 Kista Stockholm, Sweden
{shahid, adriaan, thiemo}@sics.se

Krister Landernäs
ABB Corporate Research
SE-72178 Västerås, Sweden
krister.landernas@se.abb.com

Abstract

WirelessHART is a secure and reliable communication standard for industrial process automation. The WirelessHART specifications are well organized in all aspects except security: there are no separate specifications of security requirements or features. Rather, security mechanisms are described throughout the documentation. This hinders implementation of the standard and development of applications since it requires profound knowledge of all the core specifications on the part of the developer.

In this paper we provide a comprehensive overview of WirelessHART security: we analyze the provided security mechanisms against well known threats in the wireless medium, and propose recommendations to mitigate shortcomings. Furthermore, we elucidate the specifications of the Security Manager, its placement in the network, and interaction with the Network Manager.

1. Introduction

WirelessHART [14] is the first IEC approved [7] open standard for Wireless Sensor Networks (WSNs) designed primarily for industrial process automation and control systems. The applications of WirelessHART include process and equipment monitoring, environment and energy monitoring, asset management, and advanced diagnostics. The WirelessHART network is a collection of wired entities: Network Manager, Gateway, Security Manager, and Plant Automation Hosts (PAH); and wireless devices: Field devices, Adapters, Routers, Access Points, and Handheld devices [6]. The Network Manager provides overall management, network initialization functions, network scheduling and monitoring, and resource management. The Network Manager collaborates with the Security Manager for the management and distribution of security keys. The wireless devices are connected using a mesh network where each device acts as a router and must be directly connected with at least two neighboring devices to provide path diversity. The protocol stack is based on a seven layer OSI stack with additional Security and MAC sub layers. WirelessHART is a self healing and self organizing wireless protocol, in that the devices are able to find neighbors and establish paths with them, and detect network outrages and reroute.

The WirelessHART standard is developed by the HART Communication Foundation (HCF) [1] consisting of authorities in process automation and control. The WirelessHART specifications are very well designed and almost complete in all aspects except security. The provided security is spread throughout the WirelessHART specifications and the standard lacks a comprehensive document that explains and specifies the security. The network designers and device vendors encounter ambiguities regarding the complete security architecture of the WirelessHART, the strength of the provided security, the security keys needed, and the functionalities and placement of Security Manager.

The WirelessHART standard has been recently released and we are the first to analyze and clarify its security features. Our main contribution is to provide a thorough understanding of the security features in WirelessHART. We discuss the strengths and weaknesses of the provided security mechanisms in the form of threat analysis: we analyze the WirelessHART security against the well known threats in the wireless medium and propose recommendations to mitigate the impact of these threats. We also explain the security keys and their usage as the standard does not illustrate them clearly. Finally, we elaborate the functions of the Security Manager, its placement in the network, and its interaction with the Network Manager.

2. WirelessHART Security

The legacy HART protocol (HART 6 and earlier) uses only single parity check coding schemes [17] to detect communication errors. However, WirelessHART (HART 7) is a secure and reliable protocol for industrial automation. The field devices collect data about processes and securely send it, as an input, to other field devices. The routing information, security keys, and the timing information are sent to the devices in a secure way. In short, all data in the WirelessHART network travel in the form of WirelessHART commands and the confidentiality, integrity, and the authenticity of the commands are ensured. We can divide the provided security in the WirelessHART

978-1-4244-2728-4/09/$25.00 ©2009 IEEE

standard into three levels: End-to-End, Per-hop, and Peer-to-Peer.

2.1. End-to-End Security

End-to-end security is enforced to secure the communication between the source and destination from malevolent insiders.The Network Layer is used to provide end-to-end security; any data that is passed from the network layer to the data-link layer is enciphered (except for the NPDU header) and only the destination device is able to decipher it. All field devices in the WirelessHART network have unicast and broadcast sessions with the Gateway and Network Manager. Two field devices always communicate via the Gateway [1]. The Network Protocol Data Unit (NPDU) is shown in the Table 1.

NPDU Header	Security Sublayer	NPDU Payload

Table 1. WirelessHART Network Layer PDU

The NPDU payload in Table 1 is a Transport Layer PDU (TPDU) that is always encrypted using the Advanced Encryption Standard (AES) with a 128 bit key. The Security Sub-layer consists of the Message Integrity Code (MIC), the Counter, and the Security Control Byte. The NPDU header is needed for routing of data; its details can be found in the specifications [3]. The three fields in the Security Sub-layer are used as follows:

i. Security Control Byte: It is used to define the type of the security employed. The first four bits are reserved for future security enhancement and the next four bits define the security types. In HART 7.1, only three types are identified, see Figure 1 for details.

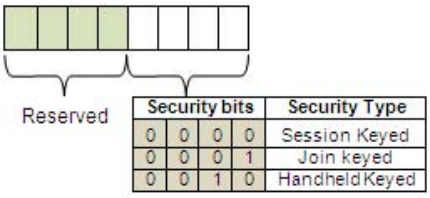

Figure 1. Security Control Byte

ii. Counter: A four-byte counter that is used to create the nonce.

iii. MIC: Keyed MIC is used for data integrity and source integrity (authentication) between source and destination. The MIC is calculated on the whole NPDU by setting the Time To Live (TTL), Counter, and MIC to zero. Four byte-strings are needed to calculate the MIC, including: NPDU header (*a*) - from control byte to MIC, NPDU payload (*m*) - the encrypted TPDU, *Nonce* - 13 byte long and provides defense against reply attacks, *AES key* - a 128 bit key needed for calculating the MIC. The same key is used for encrypting NPDU payload.

The Network Layer in the WirelessHART protocol stack provides three security services: confidentiality, integrity, and authentication. The AES in Counter with CBC-MAC (CCM) mode [27] is used for calculating the MIC to provide authentication and data integrity, and encrypting the NPDU payload to provide confidentiality. The same key is used for both encryption and MIC calculation. The CCM mode is the combination of *Cipher Block Chaining-Message Authentication Code* (CBC-MAC) and *Counter* modes. The two methods are highlighted below:

i. AES-CCM in CBC-MAC mode

In CBC-MAC, the message is enciphered using a block cipher algorithm in CBC mode and the last cipher block called MAC/MIC is constructed. In WirelessHART, the CBC-MAC mode is used to calculate the MIC at the network and the data-link layers. CBC-MAC can be used for both plain text and cipher text. This mode needs the exact number of blocks and padding is used to equalize the last block. Only Encryption is used for calculating and verifying the MIC. A formatting function is applied on the unencrypted NPUD header, the encrypted NDPU payload, and the Nonce to produce the blocks B0, B1, B2...Bi; for details about this formatting function and block formation please refer to [12]. Figure 2 shows the operations to calculate MIC using CBC-MAC mode.

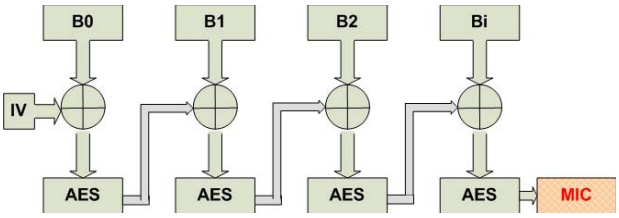

Figure 2. CBC-MAC mode for calculating MIC

ii. AES-CCM in Counter mode

The Counter mode is used for the encryption/decryption of the WirelessHART NPDU payload. Here too, the message blocks are created in the same fashion as above, but no padding is required and blocks can be manipulated in parallel. The cipher text C0, C1, C2,... will form an encrypted NPDU payload. The counter mode is shown in the Figure 3.

2.2. Per-Hop Security

The Data-Link Layer (DLL) is used to provide per-hop security between the two neighboring wireless devices us-

[1] It is possible to create peer-to-peer session between the two field devices but the WirelessHART standard prohibits such direct connections due to security reasons.

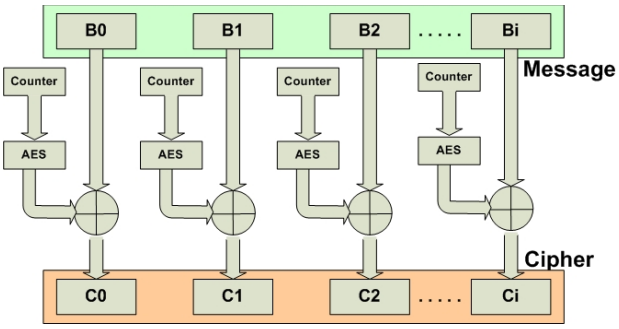

Figure 3. Counter mode for enciphering NPDU payload

ing the Network key. Per-hop security is a defense against outsiders, i.e. devices that are not part of WirelessHART network. The Network key is known to all authenticated devices in the WirelessHART network. The keyed MIC is calculated on the entire Data Link-layer PDU (DLPDU) using the AES-CCM mode as discussed above. The four parameters for the AES CCM mode are:

- *m*: the encrypted message; but as the DLPDU is not encrypted the length of this byte-string is zero in WirelessHART.
- *a*: the DLPDU from 0x41 to DLPDU payload [4].
- *N*: a 13 bytes byte-string that is formed by concatenating the Absolute Slot Number (ASN) and source address [4].
- *K*: the 128 bit Network Key.

The DLL ensures source integrity (authentication) of the message between the two neighboring devices. The DLL also offers data integrity using Cyclic Redundancy Check (CRC) [2] . The WirelessHART standard uses the 16-bits ITU-T polynomial [21] to calculate the CRC.

2.3. Peer-to-Peer Security

All traffic in the WirelessHART network flows through the gateway, but a Handheld device can create a direct one-to-one session with the field devices using the Handheld key [6]. In order to establish such connections, the Handheld device first joins the WirelessHART network using its Join key; after successful joining, the Handheld device requests the Handheld key from the Network Manager. The received Handheld key is used to create a peer-to-peer session with the field device that also receives Handheld key.

Summary

The WirelessHART standard provides data confidentiality, data integrity, authentication (source integrity), and availability (using FHSS [23] and time slotting [3]) but the standard does not enforce authorization, non-repudiation, and accounting services.

[2]CRC is not a cryptographic way to enforce integrity; rather it is a way to check communication errors.

3. Threat Analysis

A threat is an indication of a potential undesirable event [8]. The use of the wireless interface makes WirelessHART more vulnerable than legacy HART. We list possible threats against a WirelessHART network and discuss which threats are addressed by WirelessHART and which threats must be addressed. We propose recommendations to reduce the impact of the threat. The threat analysis will help developers, manufactures, and protocol designers to mitigate the impact of the threat in the design solutions.

3.1. Interference

Interference is an unintentional disruption of a radio signal; a signal with the same frequency and modulation technique can override the actual signal at the receiver. WirelessHART operates at the 2450 (2400-2483.5) MHz frequency band spectrum and has 16 channels; this spectrum can be shared with e.g. Wi-Fi, Bluetooth, WibRee (Bluetooth Low Energy Technology), ZigBee, and ISA100.11.a.

The WirelessHART standard uses Frequency-Hopping Spread Spectrum (FHSS) [23], uniquely assigned time slots using Time Division Multiple Access (TDMA), and path diversity which reduces the chances that interference causes actual harm to the operation of the network. With the reliability greater than 3-sigma (99.7300204%) [3] WirelessHART is the most reliable protocol among the current available solutions for industrial process automation especially if we compare it with ZigBee [16]. Nevertheless the strict and sensitive nature of a process automation system requires fail proof (100%) reliability and failure may produce catastrophic results. The growing number of Wi-Fi, ZigBee, Bluetooth etc. devices can make the WirelessHART frequency band more vulnerable to interference in the future.

3.2. Jamming

Jamming is normally considered an intentional interruption of radio signal when purposely introducing noise or signal with same frequency and modulation technique as used in the target network. WirelessHART is more vulnerable to jamming attacks than interference; the attacker can deliberately introduce radio signals using commonly used Bluetooth devices like cell phones or laptops.

WirelessHART uses the concept of channel Blacklisting. If a certain frequency channel is jammed or is a continuous source of interference, then it can be blacklisted. Blacklisting enhances the reliability of the WirelessHART network but at the same time it limits the number of channels available to send/receive traffic. In spite of FHSS with 15 available channels, the active attacker can jam the

WirelessHART network. The switching of channels in the FHSS is based on a pseudorandom sequence. Now if,

a. An attacker has knowledge of pseudorandom sequence (which is hard to find), he/she can calculate the actual channel. (ActualChannel = (ChannelOffset + ASN) % NumChannels) [4]

b. There are sufficient number of 2.4 GHz (Bluetooth, ZigBee, etc) devices in the range of the WirelessHART network

c. The manufacturing plant has legally deployed Wi-Fi networks in and around the WirelessHART network

d. The manufacturing plant produces sufficient amount of noise signals (which is very common there)

e. Some of the channels are already blacklisted,

then the active attacker can jam the WirelessHART network [15]. This jamming of the whole or a part of the WirelessHART network can block or even damage the machinery or plant assets.

3.3. Sybil

In a Sybil attack [11], an antagonist can hold multiple identities by introducing an adverse entity such as a node or piece of software into a network. The lack of a trusted central authority in the traditional wireless ad hoc and sensor networks make it possible for the adversary to own multiple identities.

The Network Manager in the WirelessHART network binds an entity with a unique identity. The Network Manager assigns a unique Nickname to all the connected devices. Also, every device has a globally unique ID where the ID is a combination of Device Type and Device ID. The WirelessHART Gateway maintains the list of the Unique IDs and the Network Manager maintains the list of the Nicknames; the wireless devices use these Unique IDs and Nicknames along with the session keys to maintain sessions with the Gateway and Network Manager respectively. This makes Sybil attacks almost impossible in WirelessHART networks.

3.4. Traffic Analysis

The broadcast nature of the wireless signals make them more prone to the traffic analysis than wired signals where the attacker should be physically connected to the network.

In WirelessHART networks, the NPDU header and the whole DLPDU are unencrypted and the adversary can easily analyze the WirelessHART traffic. The NPDU header fields e.g. source/Destination addresses, Security Control byte, Nonce counter, etc. are all sent in clear. These fields provide enough information to the rival to perform analysis of the network: finding new devices by analyzing join requests, work peak hours, device usage that can help to make other attacks more effective etc.

If the DLPDU payload were allowed to be encrypted with the Network key (which is also used to calculate the MIC over DLPDU) then the traffic analysis could be minimized, but then all the intermediate devices have to decrypt the NPDU at the DLL to find the destination address, routing information, etc; this will make it difficult to meet the timing requirement of 10ms which is already hard as pointed out by Song [22]. This trade off between the security and system performance makes traffic analysis attack relatively easy.

3.5. DOS

Denial-of-Service (DOS) is a common attack on all networked systems; it is against the *Availability* security service. The wireless nature of WirelessHART makes it more prone to the DOS attack than legacy HART. DOS attacks against a WirelessHART network can be launched by:

- Flooding the network with join requests as the join message is encrypted with the Well-known key at the DLL.

- Sending the fake Advertisements to the neighbors (also encrypted with the Well-known key).

- Continuously modifying the DLPDU and recomputing the CRC: Now the receiving device has to verify the message integrity by calculating the MIC (as the CRC is verified); the WirelessHART protocol uses AES in CCM for calculating MIC which is an expensive operation and requires strict timing (TsTxAckDelay =1ms) requirements [22] to verify the MIC. The unverified packet will be discarded, which results in the retransmission of the packet and consumption of network resources.

- Launching a jamming attack (see section 3.2).

3.6. De-synchronization

The attacker can disrupt the communication between two nodes by introducing false timing information in the network and engaging the devices to waste their resources in time synchronization.

The WirelessHART standard has strict timing requirements, and the Timer [14] is one of the primary modules in the network. The Timer module has to meet the timing requirements and keep the time slots (10ms) in synchronization. The MAC sub-layer is responsible for time slotting. Each time a node receives an ACK from its time source, it adjusts its clock. The timing source for a node can be a sender [22], and if the sender is compromised it can disrupt the timing between the two nodes. Hence the participating nodes waste their resources in time synchronization.

3.7. Wormhole

In a wormhole attack [10] the adversary creates a tunnel between two legitimate devices by connecting

them through the stronger wireless (by inaugurating radio transceivers at both ends) or wired links.

The potential WirelessHART devices that the attacker can use to launch wormhole attack are HART devices (wired) connected to WirelessHART network through the Adapters; the adversary can create a tunnel by connecting two field devices using their maintenance port. A tunnel can also be created by a wireless connection if the Network or Session keys are compromised.

WirelessHART can be subjected to wormhole attack if it uses graph routing (that supports redundant paths). However, if source routing is used then the device must use device-by-device route from source to destination. Source routing provides defense against wormhole attacks but is not reliable, since if any of the intermediate links fail a packet will be lost. One of the recommended solutions to prevent wormhole attack is packet leashing [13]. The physical protection of devices can avoid wired connected wormholes.

3.8. Tampering

Tampering or modification attack is the changing of stored secrets or data in transit. If the message is protected with CRC or hash, the attacker usually modifies the data and recalculates the hash or CRC. The stored secrets can be tampered by physically capturing the device and changing the data.

The WirelessHART standard uses the keyed MIC at the Network and Data-link layer to enforce integrity and provide defense against a data tampering attack. Without the knowledge of this specific key the attacker is unable to perform this attack. It is easier to perform a modification attack in the DLL than in the Network layer as the Network key is shared among all the devices and hence easy to find while session keys are device specific. Knowing the Network key and the unencrypted DLPDU, an adversary can seriously damage the normal operations of the WirelessHART network by tampering with the DLPDU and re-calculating the MIC to make it authentic.

Regular changing of the Network key is highly recommended. The physical protection of the device provides defense against the tampering of stored secrets.

3.9. Eavesdropping

Eavesdropping refers to the surreptitious listening of private communication. The *Confidentiality* security service is used to protect data from eavesdroppers.

The actual WirelessHART message consists of aggregated commands. These Commands, the Transport Byte, and the Device Status collectively form a NPDU payload that is encrypted with an AES 128 algorithm using unicast session key. Although some attacks [9] [19] have been identified against AES, none of them are able to crack it and AES is still a NIST USA recommended standard. For an attacker it is very hard to find a session key as it is short lived and unique for each device; hence message eavesdropping is difficult in the WirelessHART network. Also, the use of FHSS does not allow the eavesdropper to intercept the signal without having the pseudorandom sequence [23].

3.10. Selective Forwarding Attack

Here the compromised node selectively drops packets; the worst form is when the node does not forward any packet and creates a black-hole [20], but normally the node selectively discards packets so that it is considered as legitimate and cannot be detected by the recovering mechanisms. The Selective Forward attack is more effective if it is backed by traffic analysis.

The Network Manager in the WirelessHART network is responsible for general monitoring of the network; the Handheld device is used to monitor the specific device. They should collectively monitor the network on regular basis to detect and eliminate these attacks. The WirelessHART command 779 (Report Device Health) can be useful in detecting this attack.

3.11. Exhaustion

Any device that supports the WirelessHART protocol stack and has knowledge of network parameters (Network ID, Device ID, etc.) can send messages to the neighboring devices using the Well-known key. A fake device can use the Well-known key for calculating the MIC over the DLPDU and can use a fake Join key to encrypt and authenticate the NPDU. Although this message will be discarded when received by the Network Manager (as it uses a faked Join key) it consumes network resources along the route from the field device to the Network Manager. If a series of such join attempts are initiated by an active attacker then it can give rise to a serious DOS effect/risk.

In WirelessHART networks, the attacker can only send these messages using the join slot which will not affect the communication among other networked devices. The protection of non-cryptographic secrets (Network ID, Device ID, etc.) can also eliminate this attack.

3.12. Spoofing

Field devices in the WirelessHART network use the Well-known key not only for joining the network but also for advertisements [3]. The adversary can spoof the new joining device by sending fake advertisements and on receipt of the join request it can simply discard it. If the fake device has access to the valid Network key then the spoofing attack is more effective since the device can announce its presence to the other legitimate networked devices. Moreover, this can result in a serious blockage of network traffic.

The use of different devices while joining the network can overcome this attack. The regular monitoring and changing of the Network key by the Network Manager can minimize this attack as well.

[3] WirelessHART devices have Advertisement slots that are used to publish the device presence to the new potential devices who wish to join the network.

3.13. Collision

Collisions can occur when two or more devices try to access the same frequency channel at exactly the same time; collision can be intentional or unintentional. An attacker can also introduce collision in small portion of the packet [20].

The combination of time diversity and frequency diversity is used to minimize the collision and CRC-16 is used to detect the collision in the WirelessHART network. To *minimize* the collision, the WirelessHART protocol provides scheduled data transmission based on time slotting; TDMA and channel hopping is used to control access to the network [4]. The CRC is used to *detect* the collision based on ITU-T polynomial (aka CRC-16) [21].

The CRC-16 might not be able to detect the insertion attack (see security consideration in [21]). This attack can be avoided by better implementation and active coordination between the Physical and Data-link layer especially when the physical layer connection state changes.

Summary

The WirelessHART standard is secure enough to provide defense against most of the attacks. However, wormhole, de-synchronization, jamming, traffic analysis, spoofing, and exhaustion attacks need more attention.

Other than these attacks, the physical protection of the WirelessHART devices is very important. If the device is captured by the attacker it should self destruct because otherwise it can be cloned and the secret contents can be revealed. When a device is disconnected from the network, it should wipe out its volatile memory.

4. WirelessHART Security Manager

The Security Manager is an integral wired device in the WirelessHART network. Some of the critical points about the WirelessHART Security Manager are:

- One Security Manager can serve more than one WirelessHART network but there is only one active Security Manager per network.
- The Security manager is an application that meets the security needs of the wireless network. It can reside in a standalone device; it can be a function in the PAH; and it can be integrated in the black box consisting of Gateway, Network Manager, and Security Manager.
- The Security Manager cannot create sessions with the wireless devices; also, it is completely hidden from the Gateway.
- The interface between the Security Manager and Network Manager is not defined by the standard.
- The Security Manager provides security keys to the Network Manager that distributes them to the respective wireless devices.

Based on these prerequisites, we propose that the Security Manager should be directly connected using a dedicated link with the Network Manager at one end and with the wired/core network at the other end. This way, the Security Manager is capable of serving both the wired and wireless networks. Also, the Security Manager can serve more than one Network Manager, but the other Network Managers should be connected to the core network at one end (the other end may be connected with the Gateway). Figure 4 shows the placement of the Security Manager (SM) in the WirelessHART network.

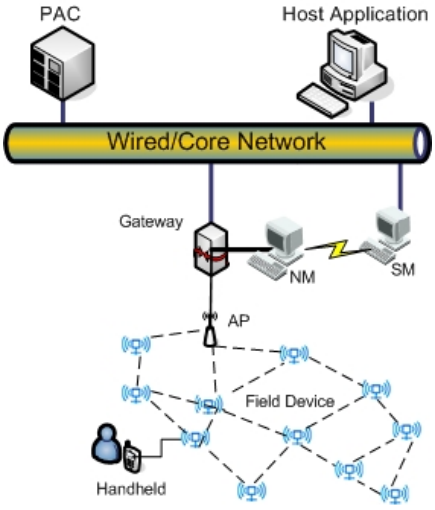

Figure 4. Our Proposed Placement of Security Manager in the network

According to the WirelessHART standard, the core responsibility of the Security Manager is to manage security keys. However, as a key manager the Security Manager is responsible for generation, storage, revocation, and renewal of keys. The Security Manager is not responsible for the distribution of keys to the wireless devices; instead the Security Manager provides keys to the Network Manager that in turn distributes them to the devices. The commands [5] for the key distribution are listed the Table 2.

Keys	Commands
Session Keys	Command 963 (Write Session)
Network Key	Command 961 (Write Network Key)
Handheld Key	Command 823 (Request Session)
Join key	Command 768 (Write Join Key)

Table 2. Key Distribution Commands in WirelessHART

As of other security functionalities, the security keys are not clearly mentioned in the WirelessHART standard and therefore we elucidate them. In a WirelessHART network at maximum eight different keys can be used to en-

crypt/decrypt the NPDU payload and to calculate the MIC at the Network and the Data-link layer. These are:

1. *Network Key*: Used to calculate the MIC over the DLPDU. It is also used for changing the broadcast session keys.

2. *Join Key*: Used to secure the NPDU [4] during the joining process. It is also used when changing the unicast session keys both of the Network Manager and the Gateway.

3. *Unicast-NM*: Used to secure the NPDU during the communication between the Network Manager and a specific Field device. It is also used for changing the Join key.

4. *Unicast-Gateway*: Used to secure the NPDU during the communication between the Gateway and a specific Field device.

5. *Broadcast-NM*: Used to secure the NPDU during a Network Manager broadcast to all the field devices. It is also used for changing the Network key.

6. *Broadcast-Gateway*: Used to secure the NPDU during Gateway broadcast to all field devices.

7. *Handheld key*: Used to secure the NPDU during the communication between the Handheld device and the connected Field device.

8. *Well-known key*: Used to calculate the MIC over the DLPDU during the join process and while sending advertisements. The value of the Well-known key is always **777 772E 6861 7274 636F 6D6D 2E6F 7267.**

All wireless devices have a pre-shared Join key; the Security Manager stores all the Join keys as well. During the joining process the Network Manager asks the Security Manager for the Join key of a new joining device. This key is used to authenticate the NPDU payload and verify the MIC of the joining request. On successful authentication, all other keys are distributed to the devices.

Another important aspect the standard lacks is the interaction between the Security Manager and the Network Manager. The Security Manager manages the keys and the Network Manager uses or distributes them to the Field devices and the Gateway. The Network Manager can request a specific key from the Security Manager by providing the following parameter over a secure channel.

1. *Network ID*: As one Security Manager can serve more than one WirelessHART network each network is uniquely identified by the Network ID.

2. *Nickname*: The Network Manager maintains a list of 2-bytes Nicknames that are used to uniquely identify the WirelessHART devices. The Unique ID (UID) can be used but UIDs are maintained by the Gateway and the Security Manager cannot cannot communicate with the Gateway directly.

3. *Key Type*:The key type can be one of the seven key types listed above. The Well-known key is always the same and can be hardcoded in the Network Manager.

The WirelessHART standard does not specify the security in the wired part of the network. However, the capabilities of the Security Manager can be extended to secure the connection between the wired devices based on asymmetric or public key cryptography [24].

5. Security Limitations of WirelessHART

Although the WirelessHART standard is designed to be a secure and reliable protocol intended to be used for industrial process automation the current release of the standard has some security limitations. These include:

- The WirelessHART protocol does not support public key cryptography which makes it unable to provide certain security services such as non-repudiation. Strong authentication, i.e. authentication without sending the security secrets over the network is not possible either.

- No mechanisms have been specified to provide authorization and accounting security services. We need accounting when the cost of WirelessHART device is attached to its usage.

- The complete key management system is not specified; however, the commands for distribution of keys have been specified.

- Security in the wired part of the network is neither specified nor enforced.

- Secure multicast communication among the Field devices is not supported.

- Secure integration of wireless and legacy HART is not specified in the WirelessHART standard.

- The architecture of the Security Manager and the interface between the Security Manager and the Network Manager is not specified in the standard.

6. Conclusions and Future Work

We have thoroughly discussed the security features in the WirelessHART standard and analyzed the specified security features against the available threats in the wireless medium. We have also identified some security limitations in the standard. However, the provided security in the wireless medium, although subjected to some threats due to its wireless nature, is strong enough to be used in the industrial process control environment. The physical

[4] For encrypting/decrypting the NPDU payload and calculating the MIC over entire NPDU.

protection of the WirelessHART devices is very important to avoid device cloning and stealing security secrets which will lead to other security attacks. Also, the careful implementation of the Network Manager is very important. The WirelessHART standard does not enforce security in the core/wired network but the connections between the wired devices must be secured. The standard provides core security services including *Confidentiality*, *Integrity*, *Authentication*, and *Availability*; however, other necessary services such as *Non-repudiation*, *Authorization* or *Access Control*, and *Accounting* are yet to be provided.

The reserved security bits (see Security control byte [3]) can be used to enhance WirelessHART security with public key cryptography [18] [26]. Although PKI is avoided in embedded devices, having a central trusted authority (Network Manager/Security Manager) and relatively high processing power and energy resources makes WirelessHART devices different from traditional sensor devices. Research in implementing ECC and RSA on sensor nodes have shown the potential for PKI in WSNs [25]. The WirelessHART's counterpart ISA100.11.a:2008 [2] also uses public key cryptography. One way to enrich the standard with security features is to identify and specify ways to provide additional security services such as *accounting* and *access control/authorization*.

Acknowledgments

This work has been performed within the SICS Center for Networked Systems funded by VINNOVA, SSF, KKS, ABB, Ericsson, Saab Systems, TeliaSonera and T2Data. This work has been partially supported by CONET, the Cooperating Objects Network of Excellence, funded by the European Commission under FP7 with contract number FP7-2007-2-224053.

References

[1] *HART Communication Foundation (HCF)*. 9390 Research Blvd., Suit I-350 Austin TX 78759 USA. http://www.hartcomm2.org/index.html.

[2] *The ISA100 Standards: Overview & Status*. http://www.isa.org/source/ISA100.11a_Release1_Status.ppt.

[3] *Network Management Specification,HCF_SPEC-085, Revision 1.1*. HART Communication Foundation, May 2008.

[4] *TDMA Data Link Layer Specification, HCF_SPEC-075, Revision 1.1*. HART Communication Foundation, May 2008.

[5] *Wireless Command Specification, HCF_SPEC-155, Revision 1.1*. HART Communication Foundation, May 2008.

[6] *WirelessHART Device Specification,HCF_SPEC-290, Revision 1.1*. HART Communication Foundation, May 2008.

[7] *IEC approves WirelessHART*. Control Engineering, Vol. 55 Issue 10 Pages 34-34, October 2008.

[8] C. Alberts and A. Dorofee. *Managing Information Security Risks: The OCTAVE Approach*. Addison Wesley, 09 July 2002.

[9] A. Bogdanoy. Multiple-differential side-channel collision attacks on aes, lecture notes in computer science. *10th international workshop on Cryptographic Hardware and Embedded Systems*, 5154(2):30–44, 2008.

[10] L. Buttyan and J.-P. Hubaux. *Security and Cooperation in Wireless Network*. Cambridge University Press, 2007.

[11] J. R. Douceur. The sybil attack. *1st International workshop on Peer-To-Peer Systems (IPTPS)*, March 2002.

[12] M. Dworkin. *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*. NIST Special Publication 800-38C, May 2004.

[13] Y.-C. Hu, A. Perrig, and D. B. Johnson. Wormhole attacks in wireless networks. *IEEE Journal on Selected Areas in Communications*, 24(2):370– 380, February 2006.

[14] A. N. Kim, F. Hekland, S. Petersen, and P. Doyle. When hart goes wireless: Understanding and implementing the wirelesshart standard. *IEEE International Conference on Emerging Technologies and Factory Automation*, pages 899–907, September 2008.

[15] Y. W. Law, M. Palaniswami, L. V. Hoesel, J. Doumen, P. Hartel, and P. Havinga. Energy-efficient link-layer jamming attacks against wireless sensor network mac protocols. *ACM Transactions on Sensor Networks (TOSN)*, 1(5):71–80, February 2009.

[16] T. Lennvall, S. Svensson, and F. Heklan. A comparison of wirelesshart and zigbee for industrial applications. *IEEE International Workshop on Factory Communication Systems*, pages 85–88, May 2008.

[17] C. Leung. Evaluation of the undetected error probability of single parity-check product codes. *IEEE Transactions on Communications*, 31(2):250–253, 1983.

[18] A. Liu and P. Ning. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks. *International Conference on Information Processing in Sensor Networks, 2008. IPSN '08*, pages 245–256, April 2008.

[19] R. C.-W. Phan. Impossible differential cryptanalysis of 7-round aes. *Information Processing Letters*, 91(1):33–38, 2004.

[20] H. K. D. Sarma and A. Kar. Security threats in wireless sensor networks. *IEEE A&E Systems Magazine*, June 2008.

[21] W. Simpson. *PPP in HDLC Framing, RFC 1549*. IETF, Network Working Group, Fremont, California 94538 USA, December 1993.

[22] J. Song, S. Han, A. K. Mok, D. Chen, M. Lucas, and M. Nixon. Wirelesshart: Applying wireless technology in real-time industrial process control. *Real-Time and Embedded Technology and Applications Symposium, 2008(RTAS-08)*, pages 377 – 386, April 2008.

[23] W. Stallings. *Data and Computer Communications*, pages 277–282. Prentice Hall, eighth edition, 2006.

[24] H. F. Tipton and M. Krause. *Information Security Management Handbook*, pages 1129–1135. Auerbach Publications, sixth edition, 2007.

[25] H. Wang and Q. Li. Efficient implementation of public key cryptosystems on mote sensors. *In Proceedings of International Conference on Information and Communication Security (ICICS)*, pages 33–38, 2004.

[26] H. Wang, B. Sheng, and Q. Li. Elliptic curve cryptography-based access control in sensor networks. *International Journal of Security and Networks*, 1(3/4):127–137, 2006.

[27] D. Whiting, R. Housley, and N. Ferguson. *Counter with CBC-MAC (CCM), RFC 3610*. IETF, Network Working Group, Fremont, California 94538 USA, September 2003.