Security Protocols Suite for Machine-to-Machine Systems

PhD Thesis

By

Andrea Bartoli

Submitted to the Universitat Politècnica de Catalunya (UPC)

DOCTOR OF PHILOSOPHY

Co-advised by:

Dr. Mischa Dohler
Senior Research Associate
Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)

Dr. Juan Hernández-Serrano
Assistant Professor
Universitat Politècnica de Catalunya (UPC)

PhD program on Telematics

Barcelona, April 2013

Dissertation Committee

PRESIDENT: **Dr. Josep Pegueroles Vallés**
Universitat Politècnica de Catalunya (UPC)
Barcelona-Spain

SECRETARY: **Dr. Jesús Alonso Zárate**
Centre Tecnològic de Telecomunicaciones de Catalunya (CTTC)
Barcelona-Spain

MEMBER: **Dr. José María Sierra Cámara**
Universidad Carlos III de Madrid (UC3M)
Madrid-Spain

ACTING MEMBER 1: **Dr. Óscar Esparza**
Universitat Politècnica de Catalunya (UPC)
Barcelona-Spain

ACTING MEMBER 2: **Prof. Francisco Javier López Muñoz**
Universidad de Málga (UMA)
Málaga-Spain

Dedicated to all those who believe in the magnificence of the human being, the prestige of the knowledge and the immense gift of the willpower:

"Our greatest glory is not in never falling but in rising every time we fall",
Confucius.

Abstract

Nowadays, the great diffusion of advanced devices, such as smart-phones, has shown that there is a growing trend to rely on new technologies to generate and/or support progress; the society is clearly ready to trust on next-generation communication systems to face today's concerns on economic and social fields. The reason for this sociological change is represented by the fact that the technologies have been open to all users, even if the latter do not necessarily have a specific knowledge in this field, and therefore the introduction of new user-friendly applications has now appeared as a business opportunity and a key factor to increase the general cohesion among all citizens.

Within the actors of this technological evolution, wireless machine-to-machine (M2M) networks are becoming of great importance. These wireless networks are made up of interconnected low-power devices that are able to provide a great variety of services with little or even no user intervention. Examples of these services can be fleet management, fire detection, utilities consumption (water and energy distribution, etc.) or patients monitoring.

However, since any arising technology goes together with its security threats, which have to be faced, further studies are necessary to secure wireless M2M technology. In this context, main threats are those related to attacks to the services availability and to the privacy of both the subscribers' and the services providers' data. Taking into account the often limited resources of the M2M devices at the hardware level, ensuring the availability and privacy requirements in the range of M2M applications while minimizing the waste of valuable resources is even more challenging.

Based on the above facts, this Ph. D. thesis is aimed at providing efficient security solutions for wireless M2M networks that effectively reduce energy consumption of the network while not affecting the overall security services of the system.

With this goal, we first propose a coherent taxonomy of M2M network that allows us to identify which security topics deserve special attention and which entities or specific services are particularly threatened. Second, we define an efficient, secure-data aggregation scheme that is able to increase the network lifetime by optimizing the energy consumption of the

devices. Third, we propose a novel physical authenticator or frame checker that minimizes the communication costs in wireless channels and that successfully faces exhaustion attacks. Fourth, we study specific aspects of typical key management schemes to provide a novel protocol which ensures the distribution of secret keys for all the cryptographic methods used in this system. Fifth, we describe the collaboration with the WAVE2M community in order to define a proper frame format actually able to support the necessary security services, including the ones that we have already proposed; WAVE2M was funded to promote the global use of an emerging wireless communication technology for ultra-low and long-range services. And finally sixth, we provide with an accurate analysis of privacy solutions that actually fit M2M-networks services' requirements. All the analyses along this thesis are corroborated by simulations that confirm significant improvements in terms of efficiency while supporting the necessary security requirements for M2M networks.

Resumen

Hoy en día, el uso generalizado de dispositivos avanzados, como móviles inteligentes, muestra que hay una tendencia creciente hacia la adopción de tecnologías innovadoras para generar progreso; la sociedad está claramente dispuesta a confiar en los sistemas de comunicación de última generación para hacer frente a las crisis actuales en materia económica y social. La razón de este cambio sociológico es que las tecnologías se han abierto a todos los usuarios, incluidos los que no tienen un conocimiento tecnológico específico, y por lo tanto, el fácil uso de nuevas aplicaciones está generando nuevas oportunidades de negocio y es además un factor clave para el aumento de la cohesión global entre todos los ciudadanos.

Dentro de los actores de esta evolución tecnológica, las redes inalámbricas machine-to-machine (M2M) son cada vez de mayor importancia. Estas redes inalámbricas se componen de dispositivos interconectados de baja potencia que son capaces de proporcionar una gran variedad de servicios con poca o ninguna intervención del usuario. Ejemplos de estos servicios pueden ser la gestión de flotas, de detección de incendios, el consumo de servicios públicos (distribución de agua, de energía, etc.) o la monitorización de los pacientes.

Sin embargo, toda nueva tecnología conlleva un conjunto nuevo de amenazas de seguridad que deben ser atacadas y, por tanto, se necesita realizar un amplio estudio de estas amenazas para asegurar adecuadamente las redes M2M. En este contexto, las principales amenazas están relacionadas con los ataques a la disponibilidad de servicios y con la privacidad de los datos tanto de los consumidores como de los proveedores de servicios. En el caso de M2M, estas amenazas son muchos mayores debido a las limitaciones de hardware presentes en muchos de los dispositivos, lo que conduce a un nuevo reto: garantizar los requisitos de disponibilidad de los servicios y de privacidad de los datos en el abanico de aplicaciones M2M minimizando el desperdicio innecesario de recursos.

Basándonos en las razones mencionadas anteriormente, esta tesis pretende proporcionar soluciones efectivas de seguridad para redes inalámbricas M2M que sean capaces de reducir el consumo de energía de la red sin prejuicio de una seguridad adecuada del sistema.

Con este objetivo, en primer lugar proponemos una taxonomía coherente de las redes M2M

que nos permita identificar los temas de seguridad que merecen especial atención y qué entidades o servicios específicos son particularmente amenazados. En segundo lugar, se define en modo eficiente, un esquema de agregación de datos seguro que es capaz de aumentar el tiempo de vida de la red mediante la optimización del consumo de energía de los dispositivos. En tercer lugar, se propone un nuevo test de autenticación a nivel físico que minimiza los costes de comunicación en canales inalámbricos y que se enfrenta con éxito a los ataques de agotamiento de las baterías. En cuarto lugar, se estudian los aspectos específicos de gestión de claves para proporcionar un nuevo protocolo que garantice la distribución de claves secretas para cada protocolo criptográfico utilizado en el sistema. En quinto lugar, se describe la colaboración con el estándar WAVE2M en la definición de un formato de trama adecuado y capaz de suportar los servicios de seguridad necesarios, incluidos los que se proponen en esta tesis; WAVE2M fue financiado para promover el uso global de una tecnología de comunicación inalámbrica emergente para servicios de ultra-bajo consumo y de largo alcance. Y finalmente sexto, proporcionamos un preciso análisis de las soluciones de privacidad que realmente se ajustan a los requisitos de las redes de servicios M2M. Todos los análisis a lo largo de esta tesis han sido corroborados por simulación confirmando mejoras significativas en términos de eficiencia. Este hecho, en conjunto con que se aportan niveles adecuados de seguridad, permite afirmar que los protocolos presentados en esta tesis son adecuados para las aplicaciones M2M.

Riassunto

Al giorno d'oggi, la grande diffusione di nuovi dispositivi, come gli smart-phone, dimostra che c'è una crescente tendenza ad affidarsi a nuove tecnologie per generare e/o sostenere il progresso; la società è chiaramente pronta ad affidarsi a sistemi di comunicazione di ultima generazione per affrontare le problematiche di oggi per quanto riguarda l'ambito economico e sociale. La ragione di questo cambiamento è rappresentato dal fatto che le tecnologie sono state aperte a tutti gli utenti, anche se quest'ultimi non hanno necessariamente un conoscimento specifico in questa materia, e quindi l'introduzione di nuove applicazioni di facile utilizzo può rappresentare un'opportunità di business e un fattore chiave per l'aumento della generale coesione fra tutti i cittadini.

Tra gli attori di questa evoluzione tecnologica, le reti wireless machine-to-machine (M2M) stanno diventando di grande importanza. Tali reti inalambriche sono costituite da dispositivi interconessi a basso consumo che sono in grado di fornire una grande varietà di servizi con poco o nessun intervento dell'utente. Esempi di questi servizi possono essere la gestione delle flotte/merci, la rilevazione d'incendi, i consumi dei servizi pubblici (distribuzione acqua, energia, ecc) o il monitoraggio dei pazienti.

Tuttavia, visto che qualsiasi nuova tecnologia emergente va di pari passo con le sue rispettive minacce alla sicurezza, che dovono essere affrontate, ulteriori studi sono necessari per assicurare le reti M2M. In questo contesto, le minacce principali sono legate agli attacchi alla disponibilità dei servizi e alla privacy dei dati sia per gli abbonati che per i distributori dei servizi. Queste minacce sono notevolmente più complicate a causa delle limitate risorse a livello hardware nella maggior parte delle implementazioni M2M, dando luogo ad una nuova sfida: garantire i requisiti di disponibilità e di privacy per tutte le applicazioni M2M riducendo al minimo l'inutile spreco di risorse preziose.

Basandoci su i motivi sopra elencati, questa tesi di dottorato è stata scritta per fornire soluzioni di sicurezza efficienti per reti wireless M2M che siano in grado di ridurre il consumo di energia della rete senza compromettere i servizi di sicurezza globali del sistema.

Con questo obiettivo in mente, per prima cosa proponiamo una tassonomia coerente delle

reti M2M che ci permette di identificare i temi di sicurezza che meritano particolare attenzione e quali sono gli enti o servizi specifici che risultano particolarmente minacciati. Secondo, definiamo uno schema efficiente per la aggregazione sicura dei dati che è in grado di aumentare la durata di vita della rete ottimizzando il consumo di energia dei dispositivi. In terzo luogo, vi proponiamo un nuovo test di autenticazione a livello fisico che riduce al minimo i costi di comunicazione nei canali wireless e che è capace di risolvere con successo gli attacchi di esaurimento delle batterie. In quarto luogo, studiamo aspetti specifici della gestione delle chiavi per fornire un nuovo protocollo che garantisca la distribuzione delle chiavi segrete per tutti gli schemi di crittografia utilizzati in questo sistema. Quinto, descriviamo la collaborazione con lo standard WAVE2M nella definizione di un formato di trama reale e in grado di supportare i servizi di sicurezza necessari, compresi quelle che abbiamo già proposto; WAVE2M è stato finanziato per promuovere l'uso globale di una tecnologia emergente per la comunicazione wireless utile per i servizi di tipo ultra-low long-range. E infine, sesto, forniamo un'accurata analisi delle soluzioni di privacy che effettivamente rispondono alle esigenze di reti M2M. Tutte le analisi in questa tesi sono giustificate da simulazioni che confermano significativi miglioramenti in termini di efficienza. Questa caratteristica, inisme con i requisiti di sicurezza proposti, ci permette di affermare che le soluzioni trovate sono adatte alle reti di basso consumo M2M.

Acknowledgements

Anno domini 2013, primavera ormai inoltrata, il sole è alto sul cielo di Barcelona, gli uccellini cinguettano una sinfonia che sembra improvvisata, ma che sicuramente parla di amore e di corteggiamento, anche il vento vuol dir la sua e sbuffa a intermittenza, il verde è il colore predominante nei miei occhi e nella mia testa. Un verde che vuol dire speranza, speranza per un futuro roseo e migliore. Migliore?! Ma migliore di cosa?

Migliore del mio passato e presente?! Impossibile! Cosa si vuol desiderare di più di tutto quello che ho, e che ho avuto e vissuto?

Famiglia, Amici, Amore, Lavoro, Viaggi...tutto quello che una giovane dottore potesse mai desiderare!

Queste brevi linee vanno dedicate a tutti coloro che mi hanno supportato, sopportato, amato, detestato e che hanno fatto parte della mia vita. Il primo prindisi va a voi e a...

> "... tutti le strade che mi hanno permesso giungere a questo momento."

Perché la vera felicità non sta nel risultato o nel punto d'arrivo, ma nel percorso, nel metterci alla prova ogni giorno e nel superare le difficoltà con costanza, cuore e cogl....Voi siete il mio percorso e la mia coppa di vino spumeggiante che ha inebriato i miei sensi, proiettandomi in un sogno che non voglio finisca mai.

Per la mia famiglia. Grazie per il supporto costante e per non avermi mai fatto mancare niente. Siete stati la mia guida e il mio faro nelle notti scure di tempesta. *Future lines of investigation*: il mondo è grande e ci sono tate persone che hanno bisogno della vostra sensibilità, disponibilità e coraggio. Aprite le braccia alle persone che hanno veramente bisogno, il mondo vi ripagherà con una moneta inestimabile: il sorriso nella bocca di un bambino.

Per i miei amici. Grazie per avermi insegnato la via della condivisione e del rispetto per ogni cultura/persona/essere vivente. Siete stati il mio supporto in tutti questi anni lontano dalla mia famiglia e senza di voi non sarei arrivato da nessuna parte. *Future lines of investigation*: abbiate il coraggio di mettervi in discussione. Non esiste solo il lavoro e i soldi non fanno la felicità. Rimboccatevi le maniche e cercate quello che è per voi il vero significato della felicità. Una volta trovato il vostro percorso: dateci dentro, non ve ne pentirete!!!

Per il mio amore. Grazie per avermi insegnato il significato di questa parola tanto speciale. Il dolce e l'amaro in un sol boccone. *Future lines of investigation*:

A- Toc Toc!

B- Chi è?

A- L'amore!

B- Non c'è nessuno, ripassate più tardi!

Per i miei colleghi. Grazie per avermi sopportato in questa prima esperienza di lavoro. Un ringraziamento speciale va dedicato a Mischa, Miquel, Juan, Apostolos, Ana Maria, France Telecom e il CTTC. Spero di aver ripagato le vostre aspettative e spero di continuare a collaborare con voi anche in futuro. *Future lines of investigation*: non ho niente da insegnarvi dal punto di vista professionale, ma ho solo una raccomandazione: come disse Charlie Chaplin "Un giorno senza un sorriso, è un giorno perso"!

Per i miei viaggi. Grazie a voi per avermi aperto gli occhi, per avermi fatto vivere la diversità e per avermi fatto apprezzare la bellezza di ogni singolo momento vissuto in un posto "lontano da casa". *Future lines of investigation*: Sud America prima o poi arriverò!

Infine, grazie a tutti!!! Vi ho tutti nel mio cuore e nella mia mente. Sappiate che potrete sempre contare su di me e chi mi conosce bene sa che mi farei in quattro per ognuno di voi!

Adesso, poggio la penna, spengo la luce e vi saluto sapendo che la mia sfida è appena cominciata...

> "È più facile essere eroe che dottore, perché eroe lo puoi essere una volta sola ma dottore lo si deve essere sempre..."

Andrea Bartoli
Barcelona, Spain
May, 2013

Contents

List of Figures

List of acronyms

3GPP	Third Generation Partnership Project
6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ACK	Acknowledge
AES	Advanced Encryption Standard
AP	Authentication Preamble
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CA	Certificate Authority
CBC	Cipher Block Chaining
CBKE	Certificate-Based Key Establishment
CCM	Counter with MAC
CDA	Concealed Data Aggregation
CIA	Confidentiality Integrity Authentication
CRC	Cyclic Redundancy Check
CRL	Certificate Revocation Lists
CSI	Channel State Information
CSMA	Carrier Sense Multiple Access
CTR	Counter Mode
CTS	Clear-to-Send
DH	Diffie-Hellman
DoD	Department of Defense
DoS	Denial of Service
DSA	Digital Signature Algorithm
DSSS	Direct Spread Spectrum Sequence
EAP	Extensible Authentication Protocol-
ECAES	Elliptic Curve Augmented Encryption Scheme
ECB	Electronic Codebook
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm

ECIES	Elliptic Curve Integrated Encryption Scheme
ECMQV	Elliptic Curve Menezes-Qu-Vanstone
ECMQV	Elliptic Curve Qu-Vanstone
EEPROM	Electrically Erasable Programmable Read-Only Memory
ETSI	European Telecommunications Standards Institute
FAST	Flexible Authentication via Secure Tunneling
FCS	Direct Frame Check Sequence
FoS	Falsification of Service
FT	France Telecom
GCM	Galois Counter Mode
GH	Group Head
GIT	Greedy Incremental Tree
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
GTK	Group Transient Key
H2H	Human-to-Human
H2M	Human-to-Machine
HMAC	Hash-based Message Authentication Code
ID	IDentification
IDS	Intrusion Detection Sytsem
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoIT	Internet of Intelligent Things
IoT	Internet of Things
IP	Internet Protocol
IPR	Intellectual Propriety Right
IT	Information Technology
IV	Initialization Vector
KEK	Key Encryption Key
KM	Key Management
KS	Key Distribution
LEAP	Lightweight Extensible Authentication Protocol
LEDS	Location aware End-to-end Data Security
LiSP	Lightweight Security Protocols
LLSP	Link-Layer Security Protocol
LoS	Leak of Service
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAC	Message Authentication Code
MANET	Mobile Ad-hoc NETwork
MDS	Maximum Distance Separable
MK	Master Key
MTBF	Mean-Time-Before-Failure

MTC	Machine-Type Communication
NGSCB	Next Generation Secure Computing Based
NSA	U.S. National Security Agency
ç **NWK**	NetWorK
OCB	Offset CodeBook
OFB	Output Feedback
OFDM	Orthogonal Frequency-Division Multiplexing
OSI	Open Systems Interconnaction
P2P	Peer-to-Peer
PAM	Priority Alarm Message
PC	Personal Computer
PDA	Personal Digital Assistant
PET	Privacy-Enhancing Technology
PFS	Perfect Forward Secrecy
PH	Privacy Homomorphism
PHY	PHYsical
PKC	Public Key Cryptography
PKE	Public Key Encryption
PKKE	Public-Key Key Exchange
PLC	Power Line Communication
PMK	Pairwise Master Key
PSK	Pre-Shared Key
PTK	Pairwise Transient Key
QoS	Quality of Service
RA	registration Authority
RFID	Radio Frequency IDentification
RL	Relying Party
ROLL	Routing Over Low power and Lossy networks
RTS	Request-to-Send
SA	Security Association
SAT	Secure Aggregation Tree
SDAP	Secure hop-by-hop Data Aggregation Protocol
SDC	Statistical Disclosure Control
SDO	Standards Developing Organization
SELDA	Secure and rELiable Data Aggregation protocol
SFD	Start of Frame Delimiter
SKC	Symmetric Key Cryptography
SKKE	Symmetric-Key Key Exchange
SNEP	Secure Network Encryption Protocol
SNR	Signal to Noise Ratio
SPINS	Security Protocol for Sensor Network
SRAM	Static Random Access Memory
SRDA	Secure Reference-Based Data Aggregation

SSL	Secure Sockets Layer
SUN	Smart Utility Network
TC	Trust Center
TCG	Trusted Computing Group
TDD	Time Division Duplexing
TKIP	Temporal Key Integrity Protocol
TPM	Tamper Proof Module
UNB	Ultra-Narrow-Band
UWB	Ultra-Wide-Band
WDA	Witness based Data Aggregation
WEP	Wired Equivalent Privacy
WiMAX	Worldwide Inter-operability for Microwave Access
WLAN	Wireless Local Access Network
WPA	Wi-Fi Protected Access
WPAN	Wireless Personal Area Network
WRAN	Wireless Regional Area Network
WSN	Wireless Sensor Network

List of Publications

Articles in JCR/Scopus SCI-indexed journals

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris and D. Barthel, *"Secure Lossless Aggregation Over Fading & Shadowing Channels For Smart Grid M2M Networks"*, IEEE Transactions on Smart Grids, Special Issue on Smart Grid Security, vol. 2, No. 4., pp. 844-864, June 2011.

- A. Bartoli, J. Hernández-Serrano, O. Leon, A. Kountouris and D. Barthel, *"Energy-Efficient PHY-Layer Packet Authenticator for Machine-to-Machine Networks"*, Transactions on Emerging Telecommunications Technologies (ETT), Special Issue Machine-to-Machine (MtM). Accepted for publication, April 2013.

Articles in non-indexed journals

- A. Bartoli, M. Navarro, J. Alonso-Zárate, M. Dohler, M. Lagunas, *"Ciutats intel·ligents: què ens cal per arribar-hi?"*, Revista Telecos.Cat, Col·legi Oficial - Associació Catalana d'Enginyers de Telecomunicacions (COETC/ACET) no. 57, pp. 10-13, September 2012.

Conference papers

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Secure Lossless Aggregation for Smart Grid M2M Networks"*, in Proceedings of IEEE First Int'l Conference on Smart grid communications, IEEE SmartGridComm 2010, 4-6 October 2010, Gaithersburg, Maryland (USA).

- A. Bartoli, J. Hernández-Serrano, M. Dohler, A. Kountouris, D. Barthel, *"Low-Power Low-Rate Goes Long-Range: The Case for Secure & Cooperative Machine-to-Machine Communications"*, in Proceedings of Workshop on Wireless Cooperative Network Security (WCNS), 13 May 2011, Valencia (Spain).

- A. Bartoli, M. Soriano, J. Hernández-Serrano, M. Dohler, A. Kountouris, D. Barthel, *"Security and Privacy in your Smart City"*, in Proceedings of Barcelona Smart Cities Congress 2011, 29-2 December 2011, Barcelona (Spain).

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Vulnerability Test Business Opportunity for Smart Grid Communication System Certification"*, in Proceedings of Madrid Smart Grids Congress 2012, 22-23 October 2012, Madrid (Spain).

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"On the Ineffectiveness of Today's Privacy Regulations for Secure Smart City Networks"*, in Proceedings of Barcelona Smart Cities Congress 2012, November 2012, Barcelona (Spain).

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Optimizing Energy-Efficiency of PHY-Layer Authentication in Machine-to-Machine Networks"*, in Proceedings of IEEE Global Communications conference exhibition & industry Forum (GLOBECOM 2012), 3-7 December 2012, Anaheim,California (USA).

Patents

- A. Kountouris, D. Barthel, M. Dohler, A. Bartoli, J. Hernández-Serrano, M. Soriano, *"Method Of Processing A Data Packet On Transmission, Method Of Processing A Data Packet On Reception, Device And Node Equipment Associated Therewith"*, PCT-FR2012-051352, priority date 17/06/2011.

Internal reports

- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *Taxonomy & State-of-the-Art*, First Deliverable MAESTRO Project in collaboration with ORANGE/-France Telecom, March 2010, Barcelona (Spain).

- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *Security Protocols & Cross-Layer Issues*, Second Deliverable MAESTRO Project in collaboration with ORANGE/France Telecom, September 2010, Barcelona (Spain).

- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *Analysis and Design of Secure Data Aggregation Schemes*, Third Deliverable MAESTRO Project in collaboration with ORANGE/France Telecom, March 2011, Barcelona (Spain).

- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *The Progress of Wavenis-OSA Standard Alliance Regarding Security*, Fourth Deliverable MAESTRO Project in collaboration with ORANGE/France Telecom, September 2011, Barcelona (Spain).

- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *The Progress of the Security WG of WAVE2M Standard Alliance Applied to a Smart Grid Scenario*, Fifth Deliverable MAESTRO Project in collaboration with ORANGE/France Telecom, March 2012, Barcelona (Spain).
- Andrea Bartoli, Juan Hernández-Serrano, Miquel Soriano, Mischa Dohler, *All Contributions to MAESTRO Project*, Sixth Deliverable MAESTRO Project in collaboration with ORANGE/France Telecom, September 2012, Barcelona (Spain).

Chapter 1

Introduction

"The logic takes you from A to B. Imagination will take you everywhere."
Albert Einstein.

Contents

Taking into account today's concept of communication, socialization and how information is exchanged, it's clear that at least in industrialization countries, there is a very powerful trend based on using innovative advanced technologies to decrease the distance and reduce the delay for fast reliable communications. This is the case of globalization, which connects people all around the world exploiting today's diffusion of smart phones or innovative devices mainly through social networks or text-messaging applications.

Although in the last decades communications have increasingly shifted from human-to-human (H2H) to human-to-machine (H2M), the tremendous number of applications that are arising today, which exploit the incredible advances made in the last years at a software/hardware level, are contributing to change the way we interact with objects as well. This technological evolution is driven by our economy needs, which are requiring to improve every communication systems and promote new ones to better exploit our resources.

In this sense, concepts such as Internet of Things (IoT) or Machine-to-Machine (M2M) are changing again our daily life by driving the technological market and by allowing the

development of automatic self-organized networks which aim to improve efficiency and reduce the wasting. In general, self-organized networks are those communication systems that are able to run without human intervention (the machines automatically interact with each other) using intelligent devices to control critical aspects of our society while incurring low impact on the environment.

However, not all the self-organized networks are the same. Notably, on the one hand, the IoT can be seen as a technology that virtually connects every object you can think of to the Internet, allowing devices to share important data, as well as controlling them. While, on the other hand, M2M can be defined as a technology that allows communication between peers to automatically exchange data; thus reducing time and increasing efficiency. In a few words, as M2M networks can work with every communication system and thus it does not need expensive infrastructure, IoT is exploiting M2M networks to enable new services through the Internet.

Figure 1.1 shows IoT and M2M typical services architectures. On the left side, we represent M2M typical low-power networks where devices exchange data using different short-range communication paradigms, such as wireless channel. On the right side, we represent the connection with the central gateway or service provider, which mainly makes use of two long-range communication systems: the first one exploits the Internet capabilities, thus allowing IoT applications; the second one exploits alternative communication systems, such as GSM, thus allowing M2M applications.

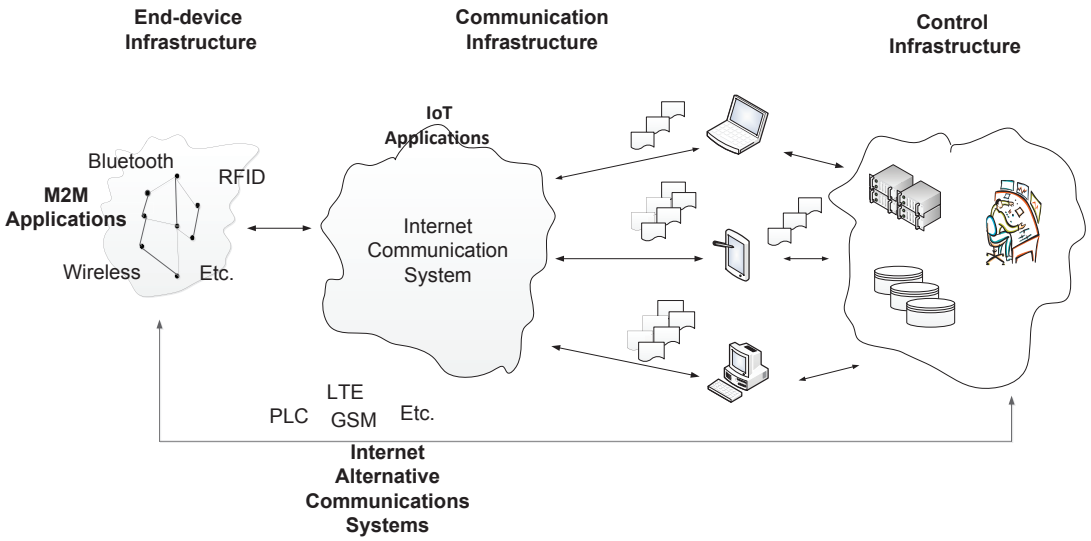

Figure 1.1: M2M and IoT services architectures.

Taking into account all the possible applications where these technologies can be implemented, Harbor Research [Harbor Research, 2011] has described the effects of M2M networking as “Pervasive Intelligence”, allowing businesses to make immediate decisions based on accurate, real-time data from near and far portions of critical infrastructures. For this reason, numerous

markets are including the use of wireless media to inter-connect specialized smart devices in a variety of new businesses: M2M is increasingly used for remote monitoring of environmental conditions (at landfills for example), for industrial monitoring of chemical containers, pipeline status and capacity management, alerting operators to dangerous conditions as well as reducing maintenance costs and increasing the reliability of systems and service delivery.

With that said, we can imagine the possible benefits of IoT services in future scenarios: connecting things such as blood pressure or heartbeat sensors to the Internet can enable third parties to create modern web or mobile applications using the collected data that, on consequence, can provide useful data for future health services (fast response to unexpected events). Considering M2M applications, wireless sensors networks (WSN) can generate a huge amount of information in real time that can be used to improve product performance and/or monitoring the machine status in remote fashion.

Nevertheless, after a complete analysis of these innovative technologies, we understand that IoT and M2M, besides providing with better efficiency, introduce new challenges in order to support reliable and secure services. Generally speaking, this problem is common to every innovation: novel communications technologies are able to provide many opportunities, but, at the same time, they can open the door to a wide new range of dangerous threats, which can jeopardize communications and the user's privacy as well. Preciously for this reason, the core of this thesis deals with securing M2M networks.

In general, M2M technology can be used for several applications in a variety of industry markets, but in this thesis we mainly treat wireless networks for system monitoring, such as utility metering services. In this scenario, M2M uses wireless networking to communicate in real time with embedded sensors to enable worldwide organizations to save energy, water and natural resources. On the one hand, from the service provider's perspective, the public sector can develop and implement profitable M2M green applications [Sallinen, 2010, Muhaidheen, 2007, Kim et al., 2010, Lu et al., 2011] to improve efficiency to boost revenues. On the other hand, from the end consumer perspective, M2M can improve the quality of life of every citizen by offering a new range of services. Nevertheless, it's clear that security it's a mandatory issue for the successful implementation of this technology. To this end, we first study the state of the art of security for wireless low-power communication systems or traditional communication system schemes, then we propose new solutions and several improvements to current protocols, and finally we validate the proposed solutions.

1.1 Problem Statement

The advent of innovative communication systems has emphasized the fact that the nowadays implemented wireless communications protocols, which are based on traditional techniques to allow secure communication, are not able to efficiently cope with the new networks features. Since every technology goes together with their security threats, these traditional solutions result

unsuitable for M2M networks: they are not able to properly challenge the arising threats and thus new solutions are necessary.

In this sense, after a first approach to the state of the art, we have identified that availability of the communications and thus protection against jamming, denial-of-service and exhaustion attacks have not been properly addressed from the community. In order to improve the already proposed security solutions or to propose new ones, several M2M networks features are analyzed in this section. Going more into details, the following risks, which have been recommended related to security for M2M systems, are those M2M networks features that should be taken into account to provide reliable security solutions in this context:

1. **Wireless medium**. M2M communication system implement wired and wireless channels. Wired systems are considered to be more secure than wireless ones. Instead, the wireless broadcast medium makes the physical layer very accessible for an attacker, which can jam, inject or modify link layer packets without difficulty and which can easily compromise and spoof a smart device.

2. **Supernode uses**. The availability of a powerful supernode and/or gateway, which can manage security issue in a centralized manner, varies with the application space; i.e., in some applications such powerful nodes are available and in some they are not. However, considering that a supernode is a single critical point of failure, defining a protocol which uses supernodes cannot be suitable for M2M applications. In a few worlds, if an attacker is able to block the supernode transmissions, a part of the network, such as a cluster, may be isolated and thus will not be able to communicate with the final gateway.

3. **Limited resources**. Nodes in M2M networks are often "low-end" devices with constrained resources and hence the use of well-known but expensive security algorithms (e.g. asymmetric cryptography) is often not feasible and may even be questionable.

4. **Inter-operability**. M2M networks take advantage of several communication paradigms to permit the capillary exchange of information from the source message to the final gateway. Since the system has to take into account several communication paradigms, and thus their typical threats, security must be designed evaluating where to place cut-points to control, and eventually block, traffic transmission. Particularly important are those critical points where different domains are crossed.

5. **Unattended devices**. Typical M2M application may implement thousands of devices which are left unattended for years of operation without the possibility of human intervention. For this reason, security issues must be guaranteed at the design stage with self-healing mechanisms.

6. **Support for multicast, groupcast, anycast and highly directional traffic**. Security must support several types of communication. These kinds of communications are often

implemented to improve efficiency. For example one multicast message is more efficient, in terms of energy consumption and time-operation, than multiple unicast messages. Thus, security must consider every kind of transmission.

7. **Automated operations (auto-configuration and auto-organization at fast convergence)**. M2M devices must be able to react when environment changes have been identified. In order to efficiently use the resources and to avoid the wasting, self-organized solutions are a key improvement for M2M applications. For example, since M2M uses wireless media, technologies that allow choosing the best spectrum opportunities based on the SNR are specifically appropriate. This optimization can exponentially increase the network lifetime.

8. **Large scale network roll-outs with widely distributed devices**. In order to connect also rural or remote locations to the principle backbone network, a capillary system of nodes must be deployed. Since centralized architectures are not recommended for this scenario, distributed secure protocols must be defined. These methods should be able to control the entire network with distributed solutions, such as distributed intrusion detection systems, to eventually isolate a node when it is identified as a fake entity.

Above mentioned issues require special attention during the system design process so as to facilitate a commercially attractive deployment of the proposed solutions. With security being a global issue spanning through all protocol layers and across all network elements, a chain is as strong as its weakest link. Securing an embedded M2M system thus generally pertains to its most important assets, notably the physical elements in the network, the communication settings and the end consumers.

As for the first, this not only requires providing adequate physical tamper resistance to ensure the integrity of data stored by the sensor nodes, including access to memory, power supply, etc. but also has to allow secure communication to safeguard the limited resources of the devices. As for the second, this requires to guarantee the reliability of the wireless channel for securing the logical data contents which is sensed and delivered over the network from the sensing nodes towards the final gateway(s). As for the last, these require studying the end consumers' requirements and offer to them a range of security services in-line with their services requirements.

1.2 Goals and work planning

Every communication network are exposed to security threats that, if are not properly addressed, exclude them to be deployed in a plethora of envisaged civilian and military scenarios. The often wireless and distributed nature of M2M networks drastically increases the spectrum of potential security threats. Securing these communication systems implies three important dimensions:

1. securing the service provider and consumer;
2. securing the communication and thus the data exchanged through the network; and
3. securing the infrastructure and the physical devices deployed in the scenario.

This thesis was conducted thus to address the above goals proposing and then analyzing several communication energy-efficient protocols for M2M systems. In order to follow an efficient design approach, this thesis also aims to involve various layers, i.e. provides a complete security suite involving the link, networking and application layers. In addition, to be commercially viable, sought solutions ought to be IETF ROLL/6LoWPAN and IEEE 802.15.4 agnostic.

Since the starting point of any research work is the preliminary study of the system structure, the first contribution should be the identification of assets and access points of the M2M networks. An asset implies important system components (e.g. routing information, process, physical device), the access, corruption or loss of which adversely affects the system. An access point refers to the point of entry facilitating communication with or otherwise interaction with a system in order to use system resources to either handle information or gain knowledge of the information the system contains. The main assets in M2M systems are the routing information, the data contents, the channel and energy at large. Access points are typically vulnerable during data delivery, route discovery, update and advertisement.

Another important issue is the considered attacker classes and types. As for the former, typically one would consider mote class attackers, which have access to few devices with similar capabilities, and laptop class attackers, which are powerful devices with more battery power, more capable CPU, high transmit power, etc. As for the latter, one would typically consider outside attackers, which are not part of the network infrastructure, and inside attackers, which have managed to gain access to the network.

Thereupon, the security reference model typically involved is referred to as the CIA model, i.e. the solution ought to guarantee Confidentiality, Integrity and Availability. Confidentiality involves the protection of data context, routing information, routing neighbor maintenance exchanges, etc., so that only authorized network entities may access it. Integrity concerns the protection of the sensed data, routing information and routing neighbor maintenance exchanges from unauthorized and improper modification. It essentially ensures that the both neighbor state information and the exchanged and derived information maintained in the routing database are authentic, authorized and ideally undeniable. Finally, availability ensures that data and routing information exchanges and forwarding services are available when required.

With this in mind, different threats and potential attacks are pertinent to these embedded systems and thus several opportunities have been identified to improve the prior art concerning security. In the following we outline the contribution map or planning for this thesis. All our contributions are later expanded and corroborated by formal analysis and simulations which confirm their goodness for a great variety of M2M scenarios.

1.2.1 Contribution map

In this section we describe the main contributions of this thesis.

A novel taxonomy of security in M2M networks

Due to the absence of a truly accepted taxonomy of security issues in the open literature, we have aimed at generating a coherent taxonomy that properly covers the needs of current and future researchers developers in M2M networks. We have thus commenced by identifying typical architectures, their valuable assets, the threats they are vulnerable to, the attackers which could exploit this, the type of attack per OSI layer, the thus needed security services to counter these attacks, the protocols carrying out these services and the algorithms which these protocols rely on. We consider that our contribution to this topic is over; details about our proposed taxonomy can be found in Chapter 2.

Part of the contributions of this chapter was used to provide an interesting survey regarding security and cooperation for M2M networks in the following international conference:

- A. Bartoli, J. Hernández-Serrano, M. Dohler, A. Kountouris, D. Barthel, *"Low-Power Low-Rate Goes Long-Range: The Case for Secure & Cooperative Machine-to-Machine Communications"*, in Proceedings of Workshop on Wireless Cooperative Network Security (WCNS), 13 May 2011, Valencia (Spain).

Secure lossless data aggregation for M2M networks

The main objective of data aggregation is to increase the network lifetime by reducing the overall consumption of resources in terms of computation and transmitted bits. Since aggregated messages can load much more sensitive data than traditional packets and aggregation as to be seen as a typical operation in M2M networks, security is an important characteristic for data aggregation communication protocols in this scenario. However, while increasing network lifetime, data aggregation protocols may degrade important quality of service metrics, therefore, the design of an efficient data aggregation protocol is an inherently challenging task that may allow finding trade-off between energy efficiency, data accuracy, latency, fault-tolerance, and security.

In order to achieve this, data aggregation techniques are tightly coupled with how packets are routed through the network. We have thus proposed a lossless data aggregation protocol that not only avoids an extra cost for security but also reduces the overall cost of the process of sending the data. This is due to aggregation savings making up for or even exceeding the computational cost of the security operations. In Chapter 3, we describe, analyze and evaluate the proposed secure lossless aggregation mechanism.

The contributions of this chapter are published in a part of a journal article and on proceeding in one international conference paper:

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris and D. Barthel, *"Secure Lossless Aggregation Over Fading & Shadowing Channels For Smart Grid M2M Networks"*, IEEE Transactions on Smart Grids, Special Issue on Smart Grid Security, vol. 2, No. 4., pp. 844-864, June 2011.

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Secure Lossless Aggregation for Smart Grid M2M Networks"*, in Proceedings of IEEE First Int'l Conference on Smart grid communications, IEEE SmartGridComm 2010, 4-6 October 2010, Gaithersburg, Maryland (USA).

PHY-layer authentication preamble

Taking into account wireless media, one of the most critical challenges is to achieve in a secure manner a desirable level of availability during the service offering. Generally speaking, availability is guaranteed when both devices and links are active to fulfill the network functionalities. Said availability can be jeopardized in the case of denial of service (DoS) attacks. Two prominent examples of DoS attacks are: 1) disturbing the communication link through jamming and thus forcing retransmissions; and 2) engaging a device into a meaningless packet exchange. Both of these attacks exhausts the intended device's battery and thus significantly shortens its lifetime. Given these DoS problems, several security solutions have been proposed by the community. Notably, frequency hopping, channel surfing techniques [Wang and Wyglinski, 2011], and similar methods have been proposed in order to prevent jamming. However, to prevent exhaustion attacks through meaningless packet exchanges, no energy efficient solution has been proposed to date since all of them require the entire packet to be received before deciding upon its authenticity [Bartoli et al., 2011]. In order to provide a solution for the presented issues, we have explored the possibility to use an effective authentication verification method at PHY-layer able to challenge exhaustion concerns and attacks in M2M networks. This method is populated in Chapter 4.

The contributions of this chapter are published in two journals article and on proceeding in an international conference paper. In addition this solution was patented form France Telecom:

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris and D. Barthel, *"Secure Lossless Aggregation Over Fading & Shadowing Channels For Smart Grid M2M Networks"*, IEEE Transactions on Smart Grids, Special Issue on Smart Grid Security, vol. 2, No. 4., pp. 844-864, June 2011.

- A. Bartoli, J. Hernández-Serrano, O. Leon, A. Kountouris and D. Barthel, *"Energy-Efficient PHY-Layer Packet Authenticator for Machine-to-Machine Networks"*, IEEE Transactions on Emerging Telecommunications Technologies (ETT), Special Issue Machine-to-Machine (MtM). Accepted with changes, October 2012.

- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Optimizing Energy-Efficiency of PHY-Layer Authentication in Machine-to-Machine Networks"*, in Proceedings of IEEE Global Communications conference exhibition & industry Forum (GLOBECOM 2012), 3-7 December 2012, Anaheim, California (USA).
- A. Kountouris, D. Barthel, M. Dohler, A. Bartoli, J. Hernández-Serrano, M. Soriano, *"Method Of Processing A Data Packet On Transmission, Method Of Processing A Data Packet On Reception, Device And Node Equipment Associated Therewith"*, PCT-FR2012-051352, priority date 17/06/2011.

Key Management in M2M networks

As cryptographic mechanisms are one of the strongest ways to provide security services for telecommunication applications, protocols and data storage, they will be widely implemented in M2M scenario. However, as cryptographic schemes depend on the keying material, the proper management of cryptographic secret keys is essential to the effective use of cryptographic algorithms for security. Keys are analogous to the combination of a safe. If the combination becomes known to an adversary, the strongest safe provides no security against penetration. Similarly, poor key management may easily compromise strong algorithms. Ultimately, the security of information protected by cryptography directly depends on the strength of the keys, the effectiveness of mechanisms and protocols associated with keys, and the protection afforded the keys.

In a few words, cryptography concerns itself with securing information so that unauthorized individuals cannot understand the messages sent by valid actors of a network and key management is the mechanism responsible for the proper and secure distribution, creation, and revocation of the keys used from cryptographic algorithms. With that said, it is clear that cryptography can be rendered ineffective by the use of inappropriate algorithm pairing, poor physical security and weak key management protocol. For these reasons, our future efforts will focus on study and propose an effective, secure and self-organized key management protocol. This solution has to be particularly suitable for M2M technologies, thus to improve the prior art and it is described in Chapter 5.

The contributions of this chapter are published in the following accepted journal article:

- A. Bartoli, J. Hernández-Serrano, O. Leon, A. Kountouris and D. Barthel, *"Energy-Efficient PHY-Layer Packet Authenticator for Machine-to-Machine Networks"*, IEEE Transactions on Emerging Telecommunications Technologies (ETT), Special Issue Machine-to-Machine (MtM). Accepted for publication, October 2012.

Security working group of the WAVE2M Community

WAVE2M Community is an independent standards alliance whose participants work together to define the WAVE2M technology roadmap and to deliver new WAVE2M features and capabilities.

WAVE2M community simplifies network installation, remote control, data monitoring, and automated two-way communications. This technology was developed by Coronis for its wireless platforms, finished products and customizable vertical solutions. In conjunction with other communication technologies, such as Bluetooth and cellular networks, WAVE2M enables Coronis to innovate in their markets with wireless devices that are not only cost-effective, but also offer significant wireless range and run on a single set of batteries for many years. Based on WAVE2M features and capabilities, similar to the ZigBee profiles, all new WAVE2M adopters can define their own profiles to meet specific application requirements: frequency bands, data rate, output power, channel bandwidth, network topology, self-routing and self-healing options, etc.

However, connecting objects, devices and things is clearly an opportunity but also poses serious challenges. The opportunity is in instrumenting and interconnecting the devices all around the world and thus allowing it to act intelligently to improve the efficiency of the processes. The main problem remains in a sustainable way to inter-connect this large amount of objects given their obvious constraints in power, processing capabilities, memory and size. In order to solve these issues and thus provide reliable communication for the WAVE2M networks, we have collaborated with its security working group, to propose and define secure, insecure and control packet formats. These specifications are described in Chapter 5. Finally we like to underline that the proposed solutions are able to meet the energy-efficiency and security level requested for M2M systems and thus can be implemented in future WAVE2M networks.

Analysis and recommendations of end-consumer privacy for wireless M2M services

Although today's connected world, with its increased availability of data, can improve customer experience, efficiency and business options in different areas, it does come with an unwanted side effect: the same data used to improve the consumers life can also be used for purposes that consumers have not approved. This applies equally M2M applications: the data that enables the efficient running of personalized services can also be used as a source of information about the personal behavior of consumers. Traditionally, this risk is mitigated by enforcing strict controls on the data and its usage, and by keeping data that can be used for personal identification separate from other data sets.

However, the customer has to trust the data-gathering organization to handle their information in an appropriate manner. This trust is hard to earn. Furthermore, data protection is a difficult, expensive and risky endeavor, and there have been large scale data losses from respectable companies and government organizations, despite the considerable resources and expertise applied to their protection. Additional trust and cost issues are imposed by regulations that surround data retention, which place legal requirements on organizations to keep and reveal data they generate.

In this thesis we have provide privacy-preserving solutions for efficient data management for utility companies that may approach end-consumers and service-provider. These solutions

are commonly related to PET (Privacy Enhancing-Technology) protocols and will describe how to avoid the consumer excessive monitoring from internal and external possible attackers. In Chapter 6 we analyze such solutions and provide overall recommendations for M2M networks targeted to utilities.

The contributions of this chapter were published on proceeding in three international and one national conferences:

- A. Bartoli, M. Soriano, J. Hernández-Serrano, M. Dohler, A. Kountouris, D. Barthel, *"Security and Privacy in your Smart City"*, in Proceedings of Barcelona Smart Cities Congress 2011, 29-2 December 2011, Barcelona (Spain).
- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"Vulnerability Test Business Opportunity for Smart Grid Communication System Certification"*, in Proceedings of Madrid Smart Grids Congress 2012, 22-23 October 2012, Madrid (Spain).
- A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, D. Barthel, *"On the Ineffectiveness of Today's Privacy Regulations for Secure Smart City Networks"*, in Proceedings of Barcelona Smart Cities Congress 2012, November 2012, Barcelona (Spain).
- A. Bartoli, M. Navarro, J. Alonso-Zárate, M. Dohler, M. Lagunas, *"Ciutats intel·ligents: què ens cal per arribar-hi?"*, Revista Telecos.Cat, Col·legi Oficial - Associació Catalana d'Enginyers de Telecomunicacions (COETC/ACET) no. 57, pp. 10-13, Setembre 2012.

1.2.2 Working plan

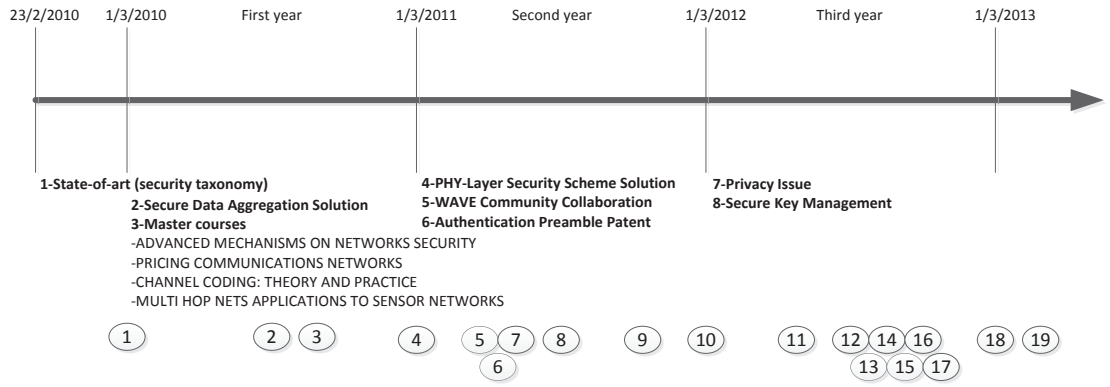

Figure 1.2: Timeline.

In this section we present the working plan, as it has been followed to this date, designed to accomplish the tasks outlined in Section 1.2.1. Figure 1.2 clearly presents a timeline of tasks in

which the milestones of this thesis are clearly highlighted.

The first year of this research work was devoted to get more research and educational training, attending to several master courses in Telematics at the *Universitat Politècnica de Catalunya*, and to start my own research in the field of wireless M2M security. Two contributions have resulted of this first year of work: the taxonomy of security in M2M networks and a secure lossless data aggregation protocol for M2M networks. Both have been previously summarized in Section 1.2.1 and extended in Chapter 2 and 3.

The second year was dedicated to study concerns specific to the wireless channel. In this context, we tried to find a solution to two main challenges: 1) the unnecessary resource consumptions due to nodes receiving packets non-intended to them, since wireless is broadcast by nature; and 2) the need of protecting the network against exhaustion attacks originated by the injection or replay or fake packets, which are at least received by the in-range network nodes. With such a goal, we proposed to use of a PHY-layer authentication preamble that allow verifying or authenticating the proper source and destination of a packet with just the reception of the PHY preamble. Thanks to the promising preliminary simulation results, this security scheme was patented in collaboration with ORANGE/FT. More details of this contribution can be found in Chapter 4.

In the third year, we focused on improving the implementability of our PHY-layer authentication preamble. With such an aim, we defined a cross-layer interface and we proposed a protocol that addresses the risk of losing synchronization of our first proposal. Besides that, in this same year, we also addressed privacy and key management issues regarding security for M2M applications.

For the remaining next months, we are focusing on proposing novel key management schemes specifically targeted to wireless M2M network. The proposed schemes should guarantee the provisioning of keying material required by our previous proposals.

In the following we enumerate the list of publications that results from this ongoing work. These publications are depicted in the figure with its number within a circle at the time that they were released.

1. MAESTRO project: Deliverable #1 (March 2010)
2. MAESTRO project: Deliverable #2 (September 2010)
3. Conference paper: Smart Grid Communications (October 2010)
4. MAESTRO project: Deliverable #3 (March 2011)
5. Conference paper: Wireless Cooperative Network Security (May 2011)
6. Patent: Authentication Preamble Security Scheme (May 2011)
7. Journal paper: Transactions on Smart Grids, special issue Smart Grid security (June 2011)

8. MAESTRO project: Deliverable #4 (September 2011)
9. Conference paper: Smart City Congress (December 2011)
10. MAESTRO project: Deliverable #5 (March 2012)
11. Magazine: IEEE Communication Magazine (August 2012)
12. MAESTRO project: Deliverable #6 (September 2012)
13. Publication: official Catalan telecommunication magazine (September 2012)
14. Conference paper: Smart Grids Congress (October 2012)
15. Conference paper: Smart City Congress (November 2012)
16. Conference paper: Glob Communications (December 2012)
17. Journal paper: Transactions on Emerging Telecommunications Technologies, special issue Machine-to-Machine (December 2012)
18. MAESTRO project: Deliverable #7 (March 2013)
19. Final Thesis Defense

Chapter 2

A novel taxonomy of security in M2M networks

"There is a driving force more powerful than steam, electricity and atomic energy: the human will."
Albert Einstein.

Contents

Despite security being the center of numerous investigations on both academic as well as industrial sides, there is no unique and universally accepted taxonomy available to date. We thus aimed to compile a coherent taxonomy composed of the most important elements needed

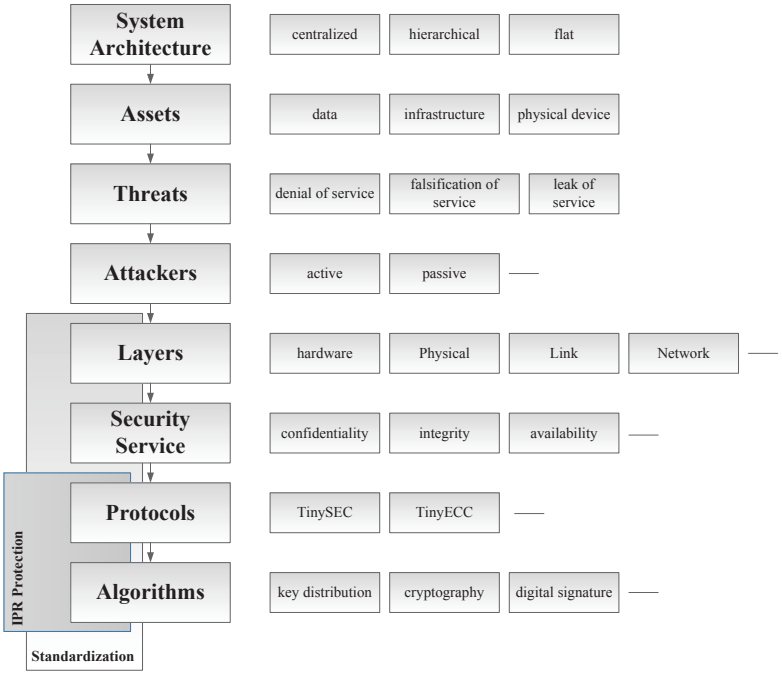

Figure 2.1: Taxonomy of security mechanisms pertinent to embedded system designs.

for security studies on embedded systems. Said taxonomy is summarized in Figure 2.1 and then treated in greater details in subsequent sections.

Visually summarized in Figure 2.1, our taxonomy commences with the availability of a system architecture, which generally can be centralized, clustered or flat. This architecture is reliant on assets, such as data (measured bits, etc.), logical infrastructure (routing paths, control information, etc.), physical devices (nodes, power supply, etc). These assets in turn can be jeopardized by possible threats, typically embodied by denial of service, falsification of service, leak of service. The malicious attacks can be carried out by attackers, which generally enjoy different degrees of attacking capabilities and can be active or passive. Attacks are typically perceived and counteracted differently at different OSI layers, such as L0 subsystem/hardware, L1 Physical, L2 Link, L3 Network, etc. This therefore requires suitable security services to be offered, such as confidentiality, integrity, availability, etc. These services are executed by suitable security protocols, such as TinySEC, etc. These protocols in turn rely on security algorithms, such as cryptography techniques, hash, digital signatures, etc. Some of these protocols and/or algorithms are part of standards and/or protected by Intellectual Property Right (IPR). Each of these elements will be subsequently discussed in more details.

2.1 System Architecture

The starting point is a suitable system architecture of choice. Typically, a design engineer has the choice between the following architectures:

- Centralized architecture.
- Hierarchical Architecture.
- Flat architecture.

A specific choice will heavily impact key performance metrics, such as support of different types of control and data flow, end-to-end performance, security, etc. Each of the above architectures has its pros and cons, which are briefly discussed below.

Centralized Architecture

A centralized architecture implies the availability of a single (or a few) entities which have control over the entire network. A typical embodiment of a centralized approach is shown in Figure 2.2. Note that a centralized approach typically means one-hop connectivity to all network members, but in the context of short-range embedded systems is typically realized via a multi-hop network.

From a performance point of view, centralized approaches are known to be superior to other approaches, given that a) optimum algorithms are used and; b) the central entity has complete knowledge of the entire system. The latter, usually comes along with a large overhead which needs to be gauged against the performance gains.

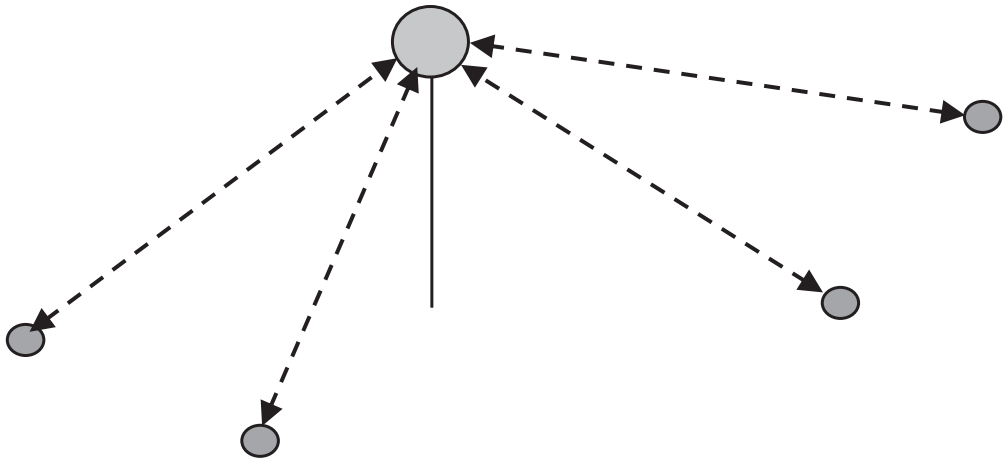

Figure 2.2: Embodiment of a centralized embedded architecture.

From a security point of view, this single centralized entity needs to monitor the safety of the entire network. The utmost important task here is, if needed, to generate and to distribute security keys to all the members via a pair-wise secure channel established with each member. This requirement might generally be too stringent for embedded multi-hop systems since a central key server must be continuously available and present in every possible subset of a group in order to support continued operation in the event of arbitrary network partitions. Continuous availability can be addressed by using fault-tolerance and replication techniques; unfortunately, the omnipresence issue is difficult to solve in a scalable and efficient manner.

Hierarchical Architecture

A hierarchical architecture implies the availability of clusters, which is controlled by a cluster head which communicates to its associated node members. A simple node is typically but not necessarily associated to a single cluster head. Communication between node and cluster head is typically but necessarily done in a single hop. Cluster heads typically communicate with each other by means of a flat architecture, or via another hierarchical tier, or via a central entity. Hierarchical networks can be heterogeneous, i.e. cluster heads (super nodes) are more powerful than simple nodes, or homogenous, i.e. cluster heads and associated nodes have a complexity of approximately the same order of magnitude. A typical embodiment of a hierarchical approach is shown in Figure 2.3. Note that a hierarchical approach is essentially a hybrid between a centralized and flat architecture.

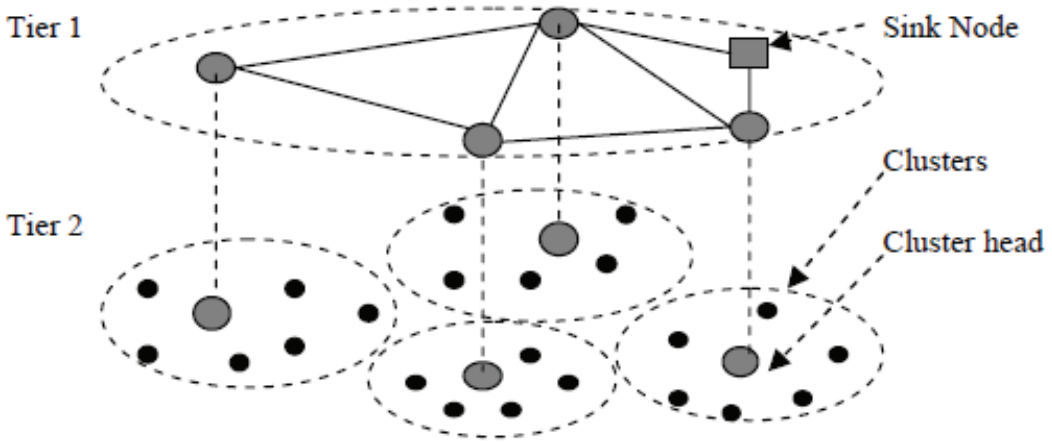

Figure 2.3: Embodiment of a hierarchical embedded architecture.

From a performance point of view, hierarchical approaches offer the advantage of trading the pros and cons of the centralized and flat architectural approaches. Notably, the overhead related to the knowledge of the cluster state is diminished, at the expense of a sub-optimum performance

when compared to centralized approaches.

From a security point of view, the load of key management is now distributed among cluster heads. Typically, each cluster head would now generate and distribute keys only to its associated members. The obvious advantage is that no single point of failure is present and the problem of omnipresence as well as scalability hence diminished. The disadvantage is that more points of attack and failure are created since cluster heads are easier compromised than some centralized security entity. When a cluster head is under attack, the entire cluster can suffer the consequences.

Flat Architecture

A flat architecture implies that all nodes of the network are equal from a networking point of view and also typically but not necessarily have the same processing capabilities. Short-range embedded systems typically require the presence of multiple hops over such a flat architecture until the sink node or gateway is reached. A typical embodiment of a flat approach is shown in Figure 2.4.

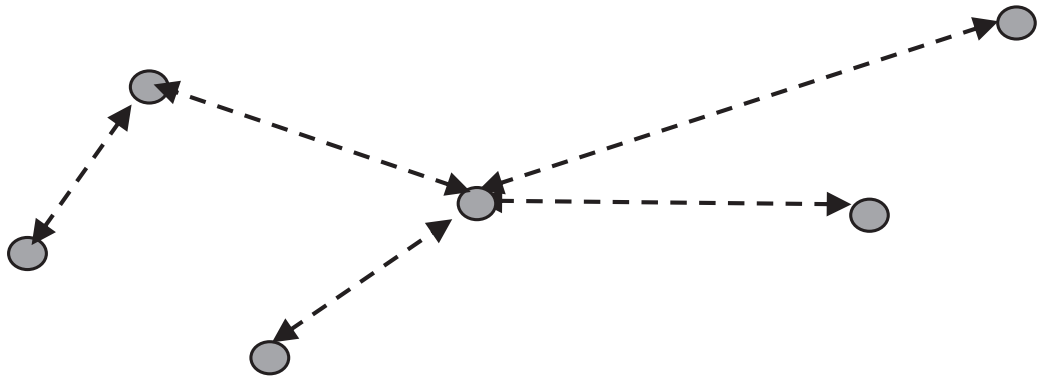

Figure 2.4: Embodiment of a flat embedded architecture.

From a performance point of view, flat architectures are known to be fairly poor. Notably, the entire MANET (Mobile Ad-hoc NETwork) community has been designing protocols for these types of architectures without any viable commercial solution being on the table today. However, when the network is not very large, then flat approaches were shown to perform sufficiently well.

From a security point of view, the flat peer-to-peer architecture requires that every member contributes an equal share to the common group secret, computed as a function of all members' contributions. This is particularly appropriate for dynamic networks since it avoids the problems with the single point(s) of trust and failure.

Architecture & M2M Networks

It has been agreed that a first focuses of this embedded security thesis are urban and utility sensing scenarios, such as metering in Smart Grid. The vast majority of applications for these scenarios are typically exhibiting little dynamics. It has further been agreed that the primarily traffic flow will be converge-cast with some peer-to-peer (P2P) traffic present. This has the following implications:

- **Control Traffic**. From a performance point of view, it is thus advantageous to keep the control centralized as the control communication overhead as much smaller as possible; optimized multi-hop networks with central final gateway are possible only whether continuous unicast control traffic flows are not used. In this sense, self-organized protocols must be defined and implemented. Only broadcast messages should be allowed from the central final gateway to every end-points.
- **Data Traffic**. From a performance point of view, the converge-cast traffic naturally passes in both up and down-link through the central entity. The emerging P2P traffic, however, can be channeled via a flat architecture (whilst the terminals involved in this transmission are instructed by the centralized control signals).
- **Security Mechanisms**. A purely centralized approach would yield the aforementioned omnipresence problem. We will thus concentrate on some hierarchical approaches to find a suitable tradeoff. Since embedded networks typically cover medium/large area using capillary devices, a centralize presence is not a suitable solution considering security. Self-organized communication protocols should be implemented to provide security in every corner of the network and for the entire network's lifetime.

In summary, a viable embedded system design is likely to require different architectural approaches for control, data and security mechanisms. Control traffic should be centralized or partially centralized. Data traffic should go through the whole multi-hop network, from the end-points to reach the final gateway. Security mechanism should be self-organized or partially self-organized.

2.2 System Assets

Above architectures are reliant on the following assets which deserve protection:

- Binary data.
- Logical infrastructure.
- Physical device components.

Each of the above has its peculiarities, which are briefly discussed below.

Binary Data

The data circulating in the embedded network is arguably the most valuable asset as it essentially delivers the required data and actuation instructions, as well as controls the data flow in the nodes and network. It is thus composed of uplink sensed data, downlink actuation data, (typically) downlink control data and node-internal software binaries:

- **Uplink Sensed Data**. As for the sensed data, one key advantage of embedded networks is the fine grain sensing that large and dense sets of devices can provide. The readings are typically acquired by the sensing unit of the sensor at given physical location at given time. The majority of nodes strictly speaking performs spot readings, such as the water consumption, temperature at the sensor, etc. However, most of the readings are good representatives of an entire sensing area; for instance, the temperature taken a given location is likely to be highly correlated to the temperature around this location. Some more sophisticated sensors are also capable of performing true area sensing, such as directional light sensors, etc. These reading locations ought generally (but not always) be secured to void unauthorized modification as well as unauthorized readings of the physical event. The sensing rate does not necessarily need to coincide with the transmission rate, where a node may choose to defer the forwarding of the sensed data until some threshold criteria is met. Typically, this would mean that an alarming situation arises, or the sensed value deviates from the previously sent value by a given delta, etc. The transmitted data ought to be sufficiently secured such that the gateway is able to retrieve information about the sensed event with sufficient reliability. The sensed values generally ought to be aggregated to avoid overwhelming amounts of traffic back to the gateway. For example, the system may average the temperature of a geographic region, combine sensor values to compute the location and velocity of a moving object, or aggregate data to avoid false alarms in real-world event detection. Depending on the architecture of the wireless sensor network, aggregation may take place in many places in the network. All these aggregation locations must be secured.

- **Downlink Actuation Data**. As for the actuation data, this typically comprises instructions by the gateway or some P2P nodes to perform some form of actuation. Examples of such actuation are the closure of a water valve, after being instructed by the utility company (because the bill has not been paid) or a nearby humidity sensor (because a leak has likely occurred). In contracts to most uplink sensed data, the downlink actuation data is typically very important and exhibits little redundancy. This requires the actuation data to be particularly secure to various forms of attacks.

- **Downlink Control Data**. As for the largely downlink control data, its role is to ensure a proper functioning of the networking infrastructure. It includes discovery, synchronization, acknowledgements, etc. Often neglected in security designs, also this data requires highest protection to void the infrastructure to be comprised.

- **Node-Internal Software Binaries**. The node-internal program instructions, etc, also constitute an important set of binary data which requires protection. Unprotected code may result in malicious alternations with severe implications.

Whilst most contributions today have focused on security mechanisms for sensed data, it is important to realize that actuation instructions, control signals as well as software binaries require equal, if not higher, degrees of protection.

Logical Infrastructure

The infrastructure supporting the embedded network is physically composed of the device components discussed below and carrying the data discussed above. Without claim for an exhaustive list, it is generally composed of the following elements:

- **Communication Channel**. The infrastructure is realized by means of communication channels which require a given bandwidth. If this element is compromised, then network connectivity is jeopardized and the network essentially fails. Protection of this resource is hence of importance.

- **Neighborhood**. The underlying element of any infrastructure is to know who is able and authorized to talk to whom in its vicinity via the given communication channel. This is typically determined by neighborhood discovery protocols. Clearly, this information ought to be protected as an attacker could aim to erase or create a false neighborhood.

- **Network Topology**. Based on the neighborhood information, a suitable network topology can be built. Said topology should be kept secret so that intruders are not able to change the topology or use it to their benefit.

- **Routing Paths**. With an established network topology, routing paths can be established. Routing and data forwarding is a crucial service for enabling communication in embedded systems. Unfortunately, many of the current routing protocols suffer from many security vulnerabilities. Therefore, utmost care ought to be taken when protecting the routing paths themselves as well as knowledge thereof.

- **Key Network Attributes**. In the context of embedded systems, there exists a large variety of key network attributes which are worth protecting. Examples are the duty cycle of the nodes and the allowed distance between nodes. Another example is the physical location of nodes, since the utility of a embedded network is its ability to accurately and automatically locate each entity in the network. A network designed to locate faults will need accurate location information in order to pinpoint the location of a fault. Unfortunately, an attacker can easily manipulate unsecured location information by reporting false signal strengths, replaying signals. A proper protection of these

(often application specific) attributes is hence also important so that the integrity of the infrastructure can be guaranteed.

As of today, most security mechanisms are able to sufficiently secure the neighborhood, network topology as well as routing paths. Techniques related to securing the communication channel as well as key network attributes, however, are scarce.

Physical Device Components

The embedded network is essentially supported by two types of physical devices, i.e. the sensor nodes and the gateway node(s):

- **Sensor/Devices Node**. The sensor/device nodes are assembled from various components, such as memory, power supply, etc, which will be discussed below. Since these components in embedded sensor nodes are generally severely constrained, in these networks a reliable security system has to be based on energy-efficiently mechanisms, such as optimized communication protocols.

- **Gateway Node**. The gateway, whilst assembled from similar components as the nodes, is generally not so heavily constrained. However, the fact that it acts essentially as the data bottleneck between embedded network and the core network implies that this component as to work as secure fire-wall respectively between unsecured and secured area. the gateway node needs to be carefully protected since otherwise full unauthorized access to the embedded network could be established.

As said above, the components of the device require special attention. Typically, an embedded node is assembled from the following components:

- **Actuator**. A sensing unit consists of an array of sensors that can measure the physical characteristics of its environment, like temperature, light, vibration, among others. Replacing a sensor by a false sensor or modifying its calibration has a serious impact on the integrity of the sensed data and hence requires appropriate protection. An actuator unit typically consists of an active component which can perform certain actions upon receiving instructions. Maliciously making an actuator perform actions is not only undesirable but also very dangerous. An actuator unit hence requires special attention.

- **Processing Unit**. The processing unit is, in most cases, a microcontroller which can be considered as a highly constrained computer that contains the memory and interfaces required to create simple applications. A microcontroller used in an embedded network's device has typically just enough computational capabilities and memory for executing simple tasks. The program instructions, sensed data as well as all acquired control information contained in the processing unit must be protected to prevent malicious attacks of various forms. This is complicated by the fact that the available power, memory

and processing capabilities are severely limited, thus preventing typically invoked security mechanisms to be used.

- **Transceiver**. The transceiver is able to send and receive messages through the wired or wireless channel. Disrupting the operation of the transceiver essentially disables the system's networking capability and thus its proper functioning. Clearly, the transceiver must be protected from tampering.

- **Power Supply**. The power unit provides the energy required by all components, and such energy may come from either a battery or from renewable sources. Since it is the lifeline of the embedded node, it needs to be efficiently protected from malicious interventions. In general, any security designs also ought to ensure that the additional energy required supporting the security algorithms can be supported by the power supply.

- **Physical Shielding**. Sensor nodes are easy to access, since they have to be physically near of the event they monitor. As a result, an adversary can try to compromise a sensor node. Once the attacker obtains, subverts, and takes control over a sensor node, it can access to its internal information, and also use it for malicious purposes by launching complex or stealthy attacks. Therefore, there should be some kind of protection to avoid such attacks, like a Tamper Proof Module (TPM).

Assets & M2M Networks

The major assets to protect in M2M networks are the binary data, notably the data and control information, and the infrastructure at large, notably the network topology and routing paths. The protection of physical devices is for the time being a secondary goal considering this thesis contributions.

2.3 Security Threats

Above assets are typically jeopardized by two fundamental threat classes that are related to system failures and malicious attacks. As for the threat due to system failure, this might be caused by the following:

- **Sensing Operations**. Nodes with little power left may perform erroneous sensing, simply because the sensing unit cannot be furnished with sufficient power. This may lead to the propagation of false data and thus jeopardizes the integrity of the system. Nodes with depleted batteries simply cease delivering the requested data and thus jeopardize the reliability of the system at large.

- **Networking Operations**. Drained nodes cause connectivity holes and, if the area grows too large, leads to networking failure which this jeopardizes the reliability of the system.

- **Hardware Failures**. Failures in the nodes' hardware can constitute a serious threat. Examples are battery, processor and memory read/write or transceiver failures. These are often impacted by climatic conditions.

- **Human Failures**. Human programming errors (leading to software security issues) and human assembling components errors (leading to hardware security issues) also jeopardize the reliability of the system.

Whilst below security mechanisms could likely cater for countermeasures against a node reaching the state of delivering erroneous sensor readings, above threats are not within the scope of this study.

As for the threat due to malicious attacks, the following three (not completely orthogonal) threat class can be distinguished:

- **Denial of Service (DoS)**. It essentially implies that any data and control services are rendered useless by the attack. It prevents the gateway and actuators to receive meaningful data or control signals. It mainly jeopardizes availability.

- **Falsification of Service (FoS)**. It essentially implies that data and control services are falsified by the attack. It does not prevent the gateway and actuators to receive meaningful data or control signals, but it may be falsified. It mainly jeopardizes integrity.

- **Leak of Service (LoS)**. It essentially implies the exposure of data and control services to the attacker. It does not prevent the gateway and actuators to receive data or control signals, but it leads of a leak of information. It mainly jeopardizes confidentiality.

Whether a particular threat requires attention in the protocol and system design depends on a suitable threat and risk analysis, which will be different for each application under consideration. We now briefly summarize the malicious attack threat classes.

Denial of Service

Generally jeopardizing availability, a vast gamut of threats triggering denial of service is known which is summarized in below (non-exhaustive and partially overlapping) list:

- **Destruction**. Nodes are vulnerable to physical harm, such as destruction, which allows the attacker to put the device out of service altogether.

- **Jamming**. Jamming of a node or set of nodes is typically achieved by transmitting a radio signal that interferes with the radio frequencies being used by the sensor network. This process is able to isolate a node or to disturb their communications.

- **Exhaustion**. The lifespan of the end-devices in a wireless low-power network is limited by the power of the battery. When the power is exhausted, the nodes cannot operate

further. For example, the attacker can fake a message asking the devices to continuously retransmit messages to exhaust its energy and eventually cause an out-of-service situation.

- **Hello Flood Attack**. A malicious node can send or replay HELLO-messages with high transmission power. It creates an illusion of being a neighbor to many nodes in the networks and can confuse the network routing badly, thus the data not reaching its destination or the network being depleted prematurely.

- **Spoofed Routing**. More in general, corruption of the routing tables being internal control information, leads to spoofed routing, which again may result in data not reaching its destination or the network being depleted prematurely.

- **Sinkhole Attack**. Sinkhole attacks typically work by making a compromised node look especially attractive to surrounding nodes with respect to the routing algorithm and lure nearly all the traffic from a particular area through a compromised node, creating a metaphorical sinkhole with the adversary at the center. Because nodes on, or near, the path that packets follow have many opportunities to tamper with application data, sinkhole attacks can enable many other attacks (selective forwarding, for example).

- **Selective Forwarding**. In a selective forwarding attack, malicious nodes behaves like a black hole and may refuse to forward certain messages and simply drop them, ensuring that they are not propagated any further. However, such an attacker runs the risks that neighboring nodes will conclude that he has failed and decide to seek another route. A more subtle form of this attack is when an adversary selectively forwards packets. An adversary interested in suppressing or modifying packets originating from a select few nodes can reliably forward the remaining traffic and limit suspicion of its wrongdoing.

- **Wormhole Attack**. In the wormhole attack [Hu et al., 2002], an adversary tunnels messages received in one part of the network over a low latency link and replays them in a different part. An adversary situated close to the final gateway may be able to completely disrupt routing by creating a well-placed wormhole. An adversary could convince nodes who would normally be multiple hops from a base station that they are only one or two hops away via the wormhole. This can create a sinkhole. Since the adversary on the other side of the wormhole can artificially provide a high-quality route to the base station, potentially all traffic in the surrounding area will be drawn through its if alternate routes are significantly less attractive.

Embedded networks without sufficient protection from DoS attacks are clearly not viable. Apart from special cases whereby an a priori trust exists in all nodes, the nodes of a M2M network cannot be trusted for the correct execution of critical network functions, notably actuation. Essential network operations that assure basic connectivity can be heavily jeopardized by nodes that do not properly execute their share of the network operations like routing,

packet forwarding, name-to-address mapping, and so on. Node misbehavior that affects these operations may range from simple selfishness or lack of collaboration due to the need for power saving to active attacks aiming at e.g. subversion of traffic.

Falsification of Service

Generally jeopardizing integrity, a vast gamut of threats triggering falsification of service is known which is summarized in below (non-exhaustive and partially overlapping) list:

- **Event Modification**. An adversary may simply alter the event being monitored. For instance, if sensors monitor the outbreak of a fire, the attacker may simply get spatially close to a given sensor or set of sensors and put a lighter close to the sensor.

- **Camouflage**. Adversaries can insert their nodes or compromise the nodes to hide in the sensor network. After that these nodes can masquerade as a normal node to attract the packets, then misroute the packets, e.g. forward the packets to the nodes conducting the privacy analysis.

- **Replay**. As the medium is wireless, the attacker can intercept the message flows easily and replays those to start a new session. Such attacks can also be related to DoS as useless receptions can jeopardize the limited resources of end-devices.

- **Injection/Modification**. Here, the content of a relayed packet is changed such that it is not or little correlated to its original content.

- **Sybil (multiple identities)**. A sybil attack is defined as a "malicious device illegitimately taking on multiple identities". Using the Sybil attack [Douceur, 2002], an adversary can “be in more than one place at once” as a single node presents multiple identities to other nodes in the network which can significantly reduce the effectiveness of fault tolerant schemes such as distributed storage [Castro and Liskov, 1999], dispersity and multipath. It may be extremely difficult for an adversary to launch such an attack in a network where every pair of neighboring nodes uses a unique pair-wise key to initialize frequency hopping or spread spectrum communication. Sybil attacks also pose a significant threat to geographic routing protocols.

- **Node Replication (duplication)**. Often referred to as impersonation, an attacker here seeks to add a node to an existing sensor network by copying (replicating) the node ID of an existing sensor node. Node replication attacks can occur if an adversary can copy the node identification of a network node. In this manner, packets could be corrupted, misrouted or deleted, and if this adversary could perform this replication it is possible that cryptographic keys could be disclosed.

- **Acknowledgement Spoofing**. Several sensor network routing algorithms rely on implicit or explicit link layer acknowledgements. Due to the inherent broadcast medium, an adversary can spoof link layer acknowledgments for overheard packets addressed to neighboring nodes. The aim is thus to convince the sender that a weak link is strong or that a dead or disabled node is alive.

Generally, an insider cannot be prevented from participating in the network, but it should only be able to do so using the identities of the nodes compromised. Using a globally shared key allows an insider to masquerade as any (possibly even nonexistent) node. Identities must therefore be verified. In the traditional setting, this might be done using public key cryptography, but generating and verifying digital signatures is beyond the capabilities of M2M nodes. One solution is to have every node share a unique symmetric pair-wise key with a trusted base station. Two nodes can then use a secure protocols to verify each other's identity and establish a shared key. A pair of neighboring nodes can use the resulting key to implement an authenticated, encrypted link between them. In order to prevent an insider from wandering around a stationary network and establishing shared keys with every node in the network, the final gateway can reasonably limit the number of neighbors per node. Thus, when a node is compromised, it is restricted to (meaningfully) communicating only with its verified neighbors. This is not to say that nodes are forbidden from sending messages to gateway nodes or aggregation points multiple hops away, but they are restricted from using any nodes except their verified neighbors to do so. In addition, an adversary can still use a wormhole to create an artificial link between two nodes to convince them they are neighbors, but the adversary will not be able to eavesdrop on or modify any future communications between them.

Leak of Service

Generally jeopardizing confidentiality, a vast gamut of threats triggering leak of service is known which is summarized in below (non-exhaustive and partially overlapping) list:

- **Tampering**. Nodes are vulnerable to physical access, such as tampering, which allows the attacker to gain access to the node and thus network.

- **Eavesdropping**. By listening to the data, the adversary could easily discover the communication contents. Network traffic is also susceptible to monitoring and eavesdropping. This should be no cause for concern given a robust security protocol, but monitoring could lead to attacks. It could also lead to wormhole or black hole attacks.

- **Traffic Analysis**. Traffic analysis typically combined with monitoring and eavesdropping. An increase in the number of transmitted packets between certain nodes could signal that a specific sensor has registered activity. Through the analysis of the traffic, some sensors with special roles or activities can be effectively identified and possibly attacked

Approaches counteracting leak of services are very similar to the ones described above in the context of falsification of service.

Security Threats & M2M Networks

The majority of outsider attacks can be prevented by simple link layer encryption and authentication using a globally shared key. Major classes of attacks not countered by link layer encryption and authentication mechanisms are tampering, jamming, exhaustion and complex routing attacks, such as Hello and wormhole attacks. As for the latter this is because, although an adversary is prevented from joining the network, nothing prevents it from using a wormhole to tunnel packets sent by legitimate nodes in one part of the network to legitimate nodes in another part to convince them they are neighbors or by amplifying an overheard broadcast packet with sufficient power to be received by every node in the network.

The simplest defense against HELLO flood attacks is to verify the bi-directionality of a link before taking meaningful action based on a message received over that link. The identity verification protocol is sufficient to prevent HELLO flood attacks. Not only does it verify the bidirectional link between two nodes, but even if a well-funded adversary had a highly sensitive receiver or had wormholes to a multiple locations in the network, a trusted final gateway that limits the number of verified neighbors for each node will still prevent HELLO flood attacks on large segments of the network when a small number of nodes have been compromised.

Wormhole and sinkhole attacks are very difficult to defend against, especially when the two are used in combination. Wormholes are hard to detect because they use a private, out-of-band channel invisible to the underlying device. Sinkholes are difficult to defend against in protocols that use advertised information such as remaining energy or an estimate of end-to-end reliability to construct a routing topology because this information is hard to verify. Routes that minimize the hop-count to a final destination are easier to verify, however hop-count can be completely misrepresented through a wormhole. When routes are established simply based on the reception of a packet as in TinyOS beaconing or directed diffusion, sinkholes are easy to create because there is no information for a defender to verify. A technique for detecting wormhole attacks is presented in [Song et al., 2005], but it requires extremely tight time synchronization and is thus infeasible for most embedded networks. Because it is extremely difficult to retrofit existing protocols with defenses against these attacks, the best solution is to carefully design routing protocols in which wormholes and sinkholes are meaningless.

Even in protocols completely resistant to sinkholes and wormholes, a compromised node has a significant probability of including itself on a data flow to launch a selective forwarding attack if it is strategically located near the source or a base station. Multipath routing can be used to counter these types of selective forwarding attacks. Messages routed over paths whose nodes are completely disjoint are completely protected against selective forwarding attacks involving at most compromised nodes and still offer some probabilistic protection whenever nodes are compromised. However, completely disjoint paths may be difficult to create. Nevertheless, in

this thesis we have focused on exhaustion attacks as main contribution.

2.4 Types of Attacks

Above security threats are potentially realized by attackers which generally differ in:

- Ability.
- Activity.
- Class.

We will briefly summarize the properties pertaining to attackers' abilities and activities, and the thus resulting class scheme.

Ability

The ability of an attacker is typically determined by the following:

- **Cost**. It relates to the cost an attacker is required to spend in terms of equipment to carry out an attack successfully. This can range from extremely cheap, where only a soldering iron and some cables are required, to prohibitively high, where top-of-the-line semiconductor test equipment is needed.
- **Skills**. It generally relates to the skills and knowledge that an attacker has to possess for a successful attack. Some attacks might be carried out by a kid after proper instruction, while others might require extensive knowledge of the particular application of the network, or a person trained in the use of special equipment. (This property can also be modeled as cost.)
- **Traces**. This relates to the traces left behind by the attack. If after the attack the node is left in the same state as before the attack, including unaltered memory contents, then this is harder to notice than an attack which causes physical destruction of the node.

Activity

Attacking activities can generally be classified as passive versus active:

- **Passive Attacks**. They extract information from the device merely by observing physical properties of the devices.
- **Active Attacks**. They involve the manipulation (tampering) of the device itself.

And also as non-invasive versus semi-invasive versus invasive:

- **Non-Invasive Attacks**. They do not manipulate the device.
- **Semi-Invasive Attacks**. Tamper with package of the device but do not make direct electrical contact with the chip's surface.
- **Invasive Attacks**. They have practically no limits to the measures which can be taken to extract the information of the device (e.g. probing station).

Note that not all semi-invasive or invasive attacks are active attacks. For instance, passive semi-invasive attacks may try to just read sensitive data from memory components, and passive invasive attacks can use a probe station to sense valuable data signals.

Examples of passive attacks are traffic analysis and camouflaging. The majority of attacks, however, are active attacks, such as routing attacks (spoofed and replayed routing information, selective forwarding, sinkhole, sybil, wormhole, HELLO flood, etc); denial of service; node malfunctions; eavesdropping; node replication; physical attacks and message corruption.

Class

To grasp both ability and activity, IBM has introduced the following taxonomy on the class of attackers:

- **Class I (clever outsiders)**. They are often very intelligent but may have insufficient knowledge of the system. They may have access to only moderately sophisticated equipment. They often try to take advantage of an existing weakness in the system, rather than try to create one.
- **Class II (knowledgeable insiders)**. They have substantial specialized technical education and experience. They have varying degrees of understanding of parts of the system but potential access to most of it. They often have highly sophisticated tools and instruments for analysis.
- **Class III (funded organizations)**. They are able to assemble teams of specialists with related and complementary skills backed by great funding resources. They are capable of in-depth analysis of the system, designing sophisticated attacks, and using the most advanced analysis tools. They may use Class II adversaries as part of the attack team.

Types of Attacks & M2M Networks

In this study, we will general aim to provide countermeasures to Class I and II adversaries.

2.5 Layers Under Attack

Capitalizing on the availability of above security threats, the adversaries' attacks have a different impact at different OSI communication layers, where we will restrict ourselves to:

- L0 Subsystem/Hardware.
- L1 Physical-Layer.
- L2 Link-Layer.
- L3 Network-Layer.

Using above-described threats, we will now briefly summarize on a per-layer basis the possible attacks and typical countermeasures taken.

L0 Subsystem/Hardware

Typical attacks carried out at subsystem level and respective countermeasures are:

- **Destruction**. It typically implies that nodes are physically destroyed, removed or stolen. Within the price limit of a sensor node, one suitable countermeasure is to design a very robust case which protects the entire node and also allows mounting it to some unmovable structure; note however that e.g. the typical glass casing used with road sensors costs around 120 euros as of Q1 2010. Another option is to have a dense enough topology so that some fallout can be tolerated. Note that another way to destroy a node or even large fields of sensor nodes is to use Taser-like devices which create a highly directive EM-beam rendering the circuitry useless.
- **Tampering**. It implies that physical access to the node is established without destroying it. It allows for various personification attacks, such as sibyl, wormhole, etc. To counteract this, a similar measure as with destruction can be taken. In addition, self-destructing mechanisms could be deployed once physical intrusion is detected. Furthermore, to protect the binary code contents present on the node, encryption with changing keys is recommended. To counter a physical attack on the microcontroller, code attestation is recommended. Furthermore, in the case of a physical attack on the external memory (EEPROM), code obfuscation can be used. If the actual sensor is attacked, various trust management schemes can be of use.
- **Jamming**. At hardware level this implies that an adversary is transmitting at such a high power that the low power amplifiers of the embedded nodes are saturated, thus rendering the intended signal useless. Two countermeasures are generally available, which are described in a little more details below. First, an ultra-narrowband (UNB) emergency channel can be maintained with a well designed filter which keeps the residual interference at bay. Second, frequency hopping may be deployed which essentially avoids the jammed bands.
- **Exhaustion**. Various higher layer attacks may lead to an exhaustion of the battery. Possible countermeasures are to allow for provision of natural energy resources which

recharge quicker, such as solar, vibration, magnetic induction, etc. Another possible countermeasure is to duty-cycle the battery in that power is only provided at pre-defined times which cannot be overridden by higher layer requirements.

L1 Physical-Layer

Typical attacks carried out at L1 PHY layer and respective countermeasures are:

- **Jamming**. At PHY layer, this implies that a point-to-point link is disturbed by an adversary transmitting at high power and the same frequency band used by the nodes. Several countermeasures are available. First, an UNB emergency channel can be maintained which usually costs little extra bandwidth and little extra hardware requirements; narrowband channels are known to have a significantly large susceptibility which potentially allows the nodes to communicate "through" the interference. Second, an ultra-wideband (UWB) radio can be used for communications which is usually resistant to interference of less bandwidth. Notwithstanding this, if an adversary has a powerful wideband jammer, then UWB will not help. Third, for embedded systems currently on the market, such as IEEE 802.15.4, "frequency hopping" or "surfing channel techniques" may help as long as not all hopping bands are jammed and/or interfered. Fourth, a very strong link layer channel code can be used which, together with suitable link layer retransmission schemes, may just be enough to facilitate communication.

- **Eavesdropping**. Since the wireless medium is essentially a broadcast medium, any adversary may just be able to eavesdrop on ongoing transmissions, given that she can decipher its contents. An interesting and emerging countermeasure currently being investigated is the use of physical layer security. Here, a time-division-duplexing (TDD) link is presumed allowing a legitimate transmitter and receiver to exchange the channel state information (CSI). This CSI is unique and only pertinent to the established physical layer link. It can thus be used to encrypt the contents of the physical layer data stream, making it difficult for an adversary to decipher the stream already at physical layer. Another interesting countermeasure is to use the physical layer to estimate some node inherent properties, such as position and distance (which can be obtained from the time and/or angle of arrivals), which allows excluding nodes which are not within predefined limits.

L2 Link-Layer

Typical attacks carried out at L2 link-layer and respective countermeasures are:

- **Exhaustion**. A typical attack launched at Link-layer is to exhaust the radio's power supply. Exhaustion typically happens due to collisions, i.e. transmitting a malicious packet with the aim to make it collide with a useful packet; overhearing, i.e. force nodes

to listen to packets which are advertised as being of interest but actually being malicious; idle listening, i.e. force nodes to wait for a packet which was promised to be transmitted but which is not; retransmissions, i.e. force nodes to retransmit a packet continuously even though it has been well received; interrogation, i.e. force nodes to issue clear-to-send (CTS) messages by continuously broadcasting a request-to-send (RTS) message. A typical countermeasure would be to design a suitable link-layer, which prevents most of above exhaustion mechanisms. Additive measures are node authorization (e.g. by extra protection of network ID), node authentication, message verification (CRC) and message encryption.

- **Acknowledgement Spoofing**. This attack convinces the sender that a weak link is strong or that a dead or disabled node is alive. Countermeasures are the same as the ones typically taken for exhaustion.

- **Sybil**. Once an adversary has access to the network, it can take multiple identities and thus cause significant harm at link-layer. Notably, these identities can occupy the channel and thus prevent legitimate nodes to communicate meaningfully. Second, it can influence data aggregation mechanisms employed at link-layer where entirely uncorrelated data is provided hence only allowing for packet "addition" but no compression. Another attack can be carried out in the context of voting, since, in the presence of a high network connectivity, some voting mechanisms are sometimes used to establish the next-hop forwarder of choice; the false identities thus can influence or even stuff the voting ballot. Again, countermeasures must be taken which essentially prevent false identities to be taken, such as re-keying, etc.

- **De-Synchronization**. An embedded multi-hop network typically relies on strong synchronization between nodes. An adversary may trigger signals, typically embedded into the link-layer control information, which cause nodes to desynchronize and thus disconnect the network. An example is the Dust Networks industrial monitoring platform where synchronization signals are inserted by the transmitter in the data packet and by the receiver piggybacked in the ACK packet. If the classical IEEE 802.15.4 link-layer was to be used, where the ACKs are not secured, an adversary node may simply modify this piggybacked synchronization signal. In general, a suitable link-layer design with some basic security mechanisms may prevent this, such as for example not using all and not always the same nodes to synchronize.

- **Traffic Analysis**. An adversary could monitor the activity of the network by simply analyzing the occupancy of the channel. Whilst the information gained from this is minimal, a possible countermeasure is to dummy packets in quieter hours of the system.

- **Eavesdropping**. Once access is gained to the network at link-level, an adversary may simply eavesdrop on the ongoing transmissions and extract required information. To

counteract this attack, suitable encryption schemes need to be used.

L3 Network-Layer

Typical attacks carried out at L3 NTW layer and respective countermeasures are:

- **Hello Flood Attack / Sybil / Spoofed Routing / Camouflage / Modification**. A prerequisite to countering this type of attacks is to ensure that the communicating nodes are authenticated prior to data encryption applied in the networking operation. Authentication ensures that the nodes are who they claim to be even though it does not provide an indication of whether the node has been compromised.

- **Sinkhole Attack / Selective Forwarding /Wormhole Attack**. These can typically be countered by performing some regular monitoring of the network using source routing protocols. Furthermore, if permissible, some physical monitoring of the field devices can be of advantage

- **Traffic Analysis / Sniffing Attacks**. An adversary could monitor the activity of links and conclude on the choice of routes and thus networking topology. Countermeasure could pertain to the insertion of dummy packets into unused routes.

Layer Under Attacks & M2M Networks

In this study, we will focus on security threats, attacks and countermeasures pertaining mainly to L2 link layer. We will aim to provide suitable authentication, authorization and encryption mechanisms to counteract the majority of above highlighted attacks.

2.6 Security Services

Above-described attacks on the OSI communication layers need to be counteracted by suitable security services. Invoking the often-used CIA (Confidentiality, Integrity and Availability) security model, the following high-level services can be identified:

- Confidentiality.

- Integrity.

- Availability.

We will now detail these services and briefly summarize their properties.

Confidentiality

Confidentiality essentially means keeping information secret from unauthorized parties. A embedded network should not leak data readings to neighboring networks or adversaries. The confidentiality objective is required in sensors' environment to protect information within the nodes as well as traveling between the nodes of the network or between the devices and the final gateway from disclosure, since an adversary having the appropriate equipment may eavesdrop on the communication. By eavesdropping, the adversary could overhear critical information such as sensitive data and routing information. Providing confidentiality at large typically implies the following security services:

- **Data Secrecy**. The data content should generally be kept secret. This prevents unauthorized nodes to obtain access to the information. It typically involves encryption algorithms.

- **Forward and Backward Secrecy**. There are two properties that need to be considered: forward secrecy, where a sensor should not be able to read any future messages after it leaves the network, and backward secrecy, where a joining entity should not be able to read any previously transmitted message. These properties may not be important in certain scenarios, where there is no need to hide the contents of the network from old nodes and new nodes authorized to perform the same tasks as their partners. However, there are other scenarios where these properties must be taken into account, such as in networks with nodes that must be authorized to perform certain tasks.

- **Code Obfuscation**. It is a mechanism that allows the protection of a valuable piece of information (e.g. the security credentials) contained inside the node. By obfuscating the code and data, the amount of time needed by the attacker to analyze the compromised nodes will increase, thus it will be more difficult to deduce the secrets from the extracted contents of program hash, the EEPROM (memory transistor) or the SRAM (memory transistor). The obfuscation methods must not be equal for all the nodes. This is to prevent the attacker from using the same method to retrieve the secrets once he/she is successful in compromising one node.

- **Key Management**. To facilitate above security services for maintaining confidentiality, some key management schemes are typically needed. It is used in symmetric encryption schemes, etc.

Integrity

Integrity means that the data produced and consumed by the network must not be maliciously altered. Unlike confidentiality, integrity is, in most cases, a mandatory property. The wireless channel can be accessed by anyone, thus any peer (outsiders and insiders) can manipulate the

contents of the messages that traverse the network. Even more, data loss or damage may occur due to the harsh communication environment, and in the worst case the network will accept corrupted data. As the main objective of an embedded network is to provide services to its users, the network will fail in its purpose if the reliability of those services cannot assured due to inconsistencies in the information. Providing integrity at large typically implies the following security services:

- **Authentication**. An adversary can easily inject messages, so the receiver needs to make sure that the data used in any decision-making process originates from the correct source. As in conventional systems, authentication techniques verify the identity of the participants in a communication, distinguishing in this way legitimate users from intruders. In the case of embedded networks, it is essential for each node and final gateway to have the ability to verify that the data received was really sent by a trusted sender and not by an adversary that tricked legitimate nodes into accepting false data. If such a case happens and false data are supplied into the network, then its behavior could not be predicted, and most of times the mission of the network will not be accomplished as expected. However, authentication for broadcast messages requires stronger trust assumptions on the network nodes.

- **Authorization**. It implies that only authorized entities can be able to perform certain operations in the network (e.g. information providing, controlling the system). Since an embedded network can be considered as one single entity, where all nodes perform the same tasks and acknowledge the role of the final gateway as manager and supervisor, it could be supposed that any authenticated device is inherently authorized to perform its tasks. Nevertheless, there might be situations (e.g. when nodes actuate over physical systems) where some members of the network need to have a proper authorization in order to perform certain tasks. Is in these situations where authorization must be taken into account.

- **Freshness**. It requires the data to be recent. This is an important security requirement to ensure that no message has been replayed meaning that the messages are in an ordering and they cannot be reused. This prevents the adversaries from confusing the network by replaying the captured messages exchanged between sensor nodes. To achieve freshness, security protocols must be designed in such a way that they can identify duplicate packets and discard them preventing replay attack.

- **Code Attestation**. Software based attestation enables a third party to verify the code running on the system to detect any maliciously altered code. Usually code attestation is done through the use of special hardware mechanisms proposed by the Trusted Computing Group (TCG) and Next Generation Secure Computing Based (NGSCB). However, these hardware mechanisms are costly and are not implemented in the current version of the

nodes. Thus this kind of software attestation is designed to provide the detection of malicious code alteration and verify that the nodes are using the correct codes.

- **Key Management**. Above integrity services typically require the availability of keys, which can but not necessarily have to be different from those used for confidentiality. Therefore, a proper key management service needs to be in place to guarantee a proper functioning of the security services.

- **Trust Management**. The basic tasks of a trust management system are determining initial trust, observing the trustee's actual behavior and updating trust accordingly. Usually, trust management systems can be classified into two categories: credential-based trust management systems and behavior-based trust management systems. This classification is based upon the approach used in order to establish trust among the peers of a system.

Availability

Availability implies that the users of an embedded network must be capable of accessing its services when they need them. The importance of availability of nodes when they are needed cannot be ignored. For example, when the network is used for monitoring purpose in manufacturing system, unavailability of nodes may fail to detect possible accidents. Availability ensures that sensor nodes are active in the network to fulfill the functionality of the network. It should be ensured that security mechanisms imposed for data confidentiality and authentication are allowing the authorized nodes to participate in the processing of data or communication when their services are needed. As nodes have limited battery power, unnecessary computations may exhaust them before their normal lifetime and make them unavailable. Sometimes, deployed security protocols or mechanisms in embedded networks are exploited by the adversaries to exhaust the nodes by its resources and makes them unavailable for the network. Providing availability at large typically implies the following security services:

- **Security Policies**. Such policies should be in place so that sensor nodes do not need to spend extra computation or do not try to allocate extra resources for security purposes.

- **Self-Organization**. There is no fixed network infrastructure as embedded networks are typically forming in an ad-hoc fashion. Therefore, nodes must have the self-organizing and self-healing capability to support multi hop-routing, frequency hopping, etc. The self-organizing security protocols should e.g. support efficient key management schemes so that nodes are able to organize themselves according to the key distribution and can build trust relations with the neighbor nodes and secure virtual infrastructure as well.

- **Non-Repudiation**. It involves providing some ability to allow traceability or network management review of participants of the routing process including the ability to determine the events and actions leading to a particular routing state. Non-repudiation

relies on the logging or other capture of on-going routing exchanges. Given the limited resources of a node and potentially the communication channel, and considering the operating mode associated with embedded systems, transaction logging or auditing process communication overhead is not be practical; as such, non-repudiation is not further considered in this study.

Security Services Time-line

Above services are not always offered but rather occur at very specific periods during the network lifetime:

- **Pre-Deployment Phase**. Prior to deployment, the services of code obfuscation and some pre-programmed key management can be provided. This, however, will not be further considered here as it depends on the manufacturing process.
- **Start-Up Phase**. During this time, all security services are invoked which guarantee a safe ramp up of all networking activities. Typically, the following services are invoked: initial key management; authentication; authorization; establishment of security policies; suitable self-organization; etc.
- **Run-Time Phase**. This is the ideal state the network ought to be in as long as possible where only the following security services need to be provided: forward, instantaneous and backward secrecy; freshness; authentication; code attestation; and trust management.
- **Adaptation Phase**. This phase is entered if there are detected changes in the network, such as compromising of nodes, malfunctioning, regularly triggered overhaul, etc. It essentially involves a re-initialization of a subset of the services provided at start-up, such as local re-keying; authorization to new nodes; self-organization in the areas where problems have been detected; etc.

Security Services & M2M Networks

We will first focus on issues pertaining to integrity and confidentiality (in order to explore cryptography tools) and then also tackle issues related to availability.

2.7 Security Protocols

Above-described services are generally offered by means of protocols, typically LEAP, TinySEC, SenSec, TinyECC, MiniSEC, SPINS, LLSP, LiSP, AMSecure, LEADS. These are the most frequently protocols used in order to guarantee security at L2 Link Layer. We also underline that some of them are only a part of security suite, for example, LEAP is a key management protocol and thus naturally needs complementing protocols during run time to guarantee tight security.

These security protocols are defined for WSN services but can be adapted to M2M applications with appropriate mortifications.

LEAP

LEAP [Zhu et al., 2003] allows multiple keying mechanisms for providing confidentiality and authentication in sensor networks. The packets exchanged by nodes in a sensor network can be classified into several categories based on different criteria, e.g. control packets versus data packets, broadcast packets versus unicast packets, queries or commands versus sensor readings, etc. The security requirements for a packet will typically depend on the category it falls in. Authentication is required for all types of packets, whereas confidentiality may only be required for some types of packets. For example, routing control information usually does not require confidentiality, whereas (aggregated) readings reported by a sensor node and the queries sent by the base station may require confidentiality. We argue that no single keying mechanism is appropriate for all the secure communications that are needed in sensor networks. As such, LEAP supports the establishment of four types of keys for each sensor node: an individual key shared with the base station, a pairwise key shared with another sensor node, a cluster key shared with multiple neighboring nodes, and a group key that is shared by all the nodes in the network:

- **Individual Key**. Every node has a unique key that it shares with the base station. This key is used for secure communication between the node and the base station. For example, a node can use its individual key to compute message authentication codes for its sensed readings if the readings are to be verified by the base station. A node may also send an alert to the base station if it observes any abnormal or unexpected behavior of a neighboring node. Similarly, the base station can use this key to encrypt any sensitive information, e.g. keying material or special instruction that it sends to an individual node.

- **Group Key**. It is a globally shared key that is used by the base station for encrypting messages that are broadcast to the whole group. For example, to comply the base station mission, such as sends queries used from the nodes of a same area. Note that from the security point of view, the use of this key has to be monitored because the compromission of this key affects every node of the same group. Therefore, an efficient rekeying mechanism is necessary for updating this key to provide continuous backward and forward secrecy.

- **Cluster Key**. It is a key shared by a node and all its neighbors, and it is mainly used for securing locally broadcast messages, e.g., routing control information, or securing sensor messages which can benefit from passive participation. Researchers have shown that in-network processing techniques, including data aggregation and passive participation are very important for saving energy consumption in sensor networks. For example, a node that overhears a neighboring sensor node transmitting the same reading as its own current

reading can elect to not transmit the same. In responding to aggregation operations such as MAX, a node can also suppress its own reading if its reading is not larger than an overheard one. Clearly, for passive participation to be feasible, sensor nodes should be able to decrypt or verify some classes of messages, e.g., sensor readings, transmitted by their neighbors. This requires that such messages be encrypted or authenticated by a locally shared key. As such, LEAP provides each node a unique cluster key shared with all its neighbors for securing its messages. Its neighbors use the same key for decrypting or verifying its messages.

- **Pairwise Shared Key**. Every node shares a pairwise key with each of its immediate neighbors. In LEAP, pairwise keys are used for securing hop-by-hop authentication. For example, a node can use its pairwise keys to secure the distribution of its cluster key to its neighbors, or to secure the transmission of its sensor readings to an aggregation node. Note that the use of pairwise keys precludes passive participation.

LEAP also includes an efficient protocol for local broadcast authentication based on the use of one-way key chains. A distinguishing feature of LEAP is that its key sharing approach supports in-network processing, while restricting the security impact of a node compromise to the immediate network neighborhood of the compromised node. The key establishment and key updating procedures used by LEAP are efficient and the storage requirements per node are small.

Cisco LEAP has had well-known security weaknesses since 2003 involving offline password cracking. LEAP uses a modified version of MS-CHAP, an authentication protocol in which user credentials are not strongly protected. Cisco suggests to the network administrators either force users to have stronger, more complicated passwords or move to another authentication protocol also developed by Cisco, EAP-FAST, to ensure security. Nevertheless, automated tool like ASLEAP demonstrates the simplicity of getting unauthorized access in networks protected by LEAP implementations.

TinySec

TinySec [Karlof et al., 2004a] is a link layer security architecture for wireless sensor networks, implemented for the TinyOS operating system. In order to overcome the processor, memory and energy constraints of sensor nodes TinySec leverages the inherent sensor network limitations, such as low bandwidth and relatively short lifetime for which the messages need to remain secure, to choose the parameters of the cryptographic primitives used. TinySec has two modes of operation: authenticated encryption (TinySec-AE) and authentication only (TinySec-Auth). With authenticated encryption, TinySec encrypts the data payload and authenticates the packet with a Message Authentication Code. The MAC is computed over the encrypted data and the packet header. In authentication only mode, TinySec authenticates the entire packet with a MAC, but the data payload is not encrypted. An important feature of TinySec is its ease of use and transparency, as many application developers will either implement the security features

incorrectly or leave out any security entirely if the security API is difficult to use. TinySec solves this problem by integrating into TinyOS at a low level.

TinySec adds 3.03% to the overall energy consumption when using TinySec-Auth mode, and 9.1% when using TinySec-AE mode. This extra energy consumption can be explained by the increased packet length imposed by TinySec along with the cryptographic computations that are required. While TinySec has been well designed for its target application domain it still suffers from a number of limitations. The primary limitation is that no key exchange mechanisms are included. TinySec uses a single network wide shared key and it is left up to the application developer to change this key as appropriate and distribute new keys to all nodes. This is not a simple task and is odds with one of the major features of TinySec; its ease of use and transparency. If an application developer ignores this issue, which is likely, the use of TinySec will give a false sense of security as any such shared key cannot be expected to remain secret for very long in sensor networks where an adversary has physical access to sensor nodes. The other major limitation of TinySec is its limited platform support; the official release of TinySec as included in version 1 of TinyOS only works on the MICA2 mote, a old device which is no longer being manufactured.

SenSec

SenSec [Krontiris et al., 2008] is another link-layer security protocol which is heavily based on TinySec but still has a number of significant differences. It is worth noting that the design of SenSec was motivated by the problems encountered while trying to use TinySec for a specific WSN deployment. However a number of the design choices were driven by the needs of this particular deployment and may not be generally applicable. SenSec has currently only been implemented for the MICA2 platform.

SenSec can be seen as an evolution of TinySec with marginally lower power consumption, due to one pass encryption and authentication, and a higher level of security, again only slightly. However evolution is not always necessarily a good thing. SenSec is not as flexible as TinySec as: it does not offer an authentication only mode and it is not as easy to use either as it is not integrated into the TinyOS distribution. SenSec also fails to address most of the major limitations of TinySec mentioned above, such as the lack of freshness checks and replay protection, no built in key exchange mechanisms and a lack of support for a variety of platforms.

TinyECC

The primary objective of TinyECC [Liu and Ning, 2008] is to provide a ready-to-use, publicly available software package for ECC-based PKC operations that can be flexibly configured and integrated into sensor network applications. TinyECC is written in nesC, with occasional in-line assembly code to achieve further speedup for popular sensor platforms including MICAz, TelosB, Tmote Sky and Imote2. A unique feature of TinyECC is its configurability. TinyECC

includes almost all known optimizations for ECC operations. Each optimization is controlled by a software switch, which can turn the optimization on or off based on the developers' needs. Different combinations of optimizations have different ROM/RAM consumption, execution time, and energy consumption. This gives the developers great flexibility in integrating TinyECC in their applications. TinyECC only includes support for the well-studied elliptic cryptography ECC schemes, such as the Elliptic Curve Diffie-Hellman (ECDH) key agreement scheme, the Elliptic Curve Digital Signature Algorithm (ECDSA) and the Elliptic Curve Integrated Encryption Scheme (ECIES). Notably, ECDH is a variant of the Diffie-Hellman key agreement protocol on elliptic curve groups, ECDSA is a variant of the Digital Signature Algorithm (DSA) that operates on elliptic curve groups and finally ECIES is a public-key encryption scheme which provides semantic security against an adversary who is allowed to use chosen-plaintext and chosen-ciphertext attacks. ECIES is also known as the Elliptic Curve Augmented Encryption Scheme (ECAES) or simply the Elliptic Curve Encryption Scheme.

These ECC schemes allow smaller key sizes for similar security levels to the alternatives such as the original DH and DSA schemes. For each of the schemes, a party that would like to use the scheme needs to agree on some domain parameters such as the elliptic curve and a point G on the curve, and must have a key pair consisting of a private key d and a public key Q = dG.

Even though elliptic curve cryptography is feasible on sensor nodes, its energy requirements are still orders of magnitude higher compared to that of symmetric crypto-systems. Therefore, elliptic curve cryptography would make more sense to be used only for infrequent but security-critical operations, like key establishment during the initial configuration of the sensor network, or code updates.

MiniSec

MiniSec [Luk et al., 2007] is a secure network layer protocol that claims to have lower energy consumption than TinySec while achieving a level of security which matches that of Zigbee. A major feature of MiniSec is that it uses offset codebook (OCB) mode as its block cipher mode of operation, which offers authenticated encryption with only one pass over the message data. Normally two passes are required for both confidentiality and authentication. Another major benefit of using OCB mode is that the cipher text is the same length as the plaintext, disregarding the additional fixed length tag, four bytes in MiniSec's case, so padding or ciphertext stealing is not necessary. Another primary feature that MiniSec has over the other security suites mentioned here it is a strong replay protection method that is able to solve this challenge without implement traditional solutions; it works without transmit overhead, without the need of sending a large counter with each packet and without the problems associated with synchronized counters if packets are dropped. To achieve this MiniSec has two modes of operation, one for unicast packets MiniSec-U, and one for broadcast packets, MiniSec-B.

Authors of this schema have claimed that MiniSec has about one third the energy consumption of TinySec, but as MiniSec and TinySec are only available for different mote

platforms it has been difficult to confirm this claim. The tradeoff required to achieve the high level of security with such low energy overhead is to increase memory requirements, which for many WSN applications will not pose a problem. The current release of MiniSec also suffers from some implementation flaws which artificially limit the length of the 802.15.4 packets by overloading the length field, a mistake TinySec's designers also made. Furthermore MiniSec overloads a bit in the packet header that CC2420 data sheet specifies as reserved and should always be set to zero. This can lead to unpredictable operation if MiniSec requires this field to be set to 1.

As with TinySec, MiniSec does not include any key exchange and key management mechanisms as part of its current implementation despite the fact that each node requires two unique keys for every node it participates in unicast communication with. As a result applying appropriate key management protocols is by far the most difficult part of using MiniSec for any application. Also, key management aside, as MiniSec is not integrated into the TinyOS build environment it is currently not as easy to use as TinySec.

In conclusion, while MiniSec brings some interesting ideas to providing energy efficient security to wireless sensor networks, it still requires some work, especially in terms of implementation details, before it can be considered viable for practical deployments.

SPINS

SPINS [Perrig et al., 2002] is the oldest proposed security suite for wireless sensor networks and consists of two building blocks: secure network encryption protocol (SNEP) and micro, timed, streaming, loss-tolerant authentication protocol (μ-TESLA). SNEP provides data confidentiality, two parties' authentication and data freshness. μ-TESLA provides efficient broadcast authentication.

As SPINS was never fully implemented, or even fully specified, it is difficult to perform an evaluation or accurate comparison with the other security suites mentioned here. While never actually specified SPINS appears to be designed as a network level, end-to-end security protocol, but most of the design, particularly SNEP, is applicable to link layer, hop-by-hop protocol.

SNEP has a number of limitations compared with some of the other protocols aimed at the same application space, TinySec and MiniSec in particular. SNEP adds 8 bytes of overhead to each message sent, which is more than TinySec or MiniSec. SNEP includes replay protection, which TinySec lacks but it is much more vulnerable to dropped packets than MiniSec. While in theory μ-TESLA is a very suitable scheme for broadcast authentication in WSNs, in practice it presents a number of serious complications due to the need for tight time synchronization between nodes and also the problem of choosing the time period between key disclosures. As with every other scheme already mentioned, SPINS not include any key management mechanisms for exchanging, updating or revoking keys.

LLSP

LLSP [Ren et al., 2005] is a link layer security protocol which ensures message authentication, access control, message confidentiality, and replay protection. It is based on the idea of TinySec. However, it uses different packet format and crypto structure. LLSP supports early rejection capability. It has also low performance overhead. However, it supports low scalability.

LEDS

LEDS [Ren et al., 2006] provides location aware end-to-end security. LEDS also provides end-to-end authentication and en-route filtering. It provides location aware key management. LEDS can be used in small as well as large networks. LEDS divides the network in cell regions, however, number of keys increases with cell size. In addition, LEDS does not support dynamic topology.

LEDS is thus a location aware scheme that provides many security services such as data confidentiality, availability, and authenticity. In LEDS, the data confidentiality is achieved by using symmetric cryptography and linear secret sharing. To check the authenticity of the data, a legitimate report carries many MACs that are verified by the nodes in the intermediate cells. For the data availability, the overhearing nodes in every forwarding cell collaborate to inform the next cell in case a legitimate report is dropped by a malicious node.

Although overhearing nodes theoretically provide data availability, a practical method to implement this technique seems not feasible. The most logical realization is a voting system that has a high communication overhead and its management introduces a high computational complexity.

AMSecure

AMSecure [Wood and Stankovic, 2006] is a link layer security suite which provides message confidentiality, authentication, integrity, replay protection and semantic security. AMSecure was designed for use on MICAz and Telos motes, both of which use the CC2420 radio chip, which is IEEE 802.15.4 compatible. AMSecure uses the security features of the Texas Instruments CC2420 radio chip in order to provide all of its security services. An interface is provided to allow security aware applications to manage the keys being used. AMSecure support should be very easy to add existing TinyOS applications as it fits into the TinyOS active messaging stack without any need to modify the higher layer protocols or applications.

From the limited information that is available AMSecure can be considered to have a high level of security as it uses the AES cipher, which is generally held to be much more secure than RC5 or Skipjack, the ciphers almost every other suite described here uses. Also AMSecure should be competitive in terms of power consumption and speed because of it uses an hardware accelerated AES cryptography scheme. The arguably biggest weakness of AMSecure lies in its

lack of portability. Due to the fact that it relies so heavily on the CC2420 security operations it would prove difficult to port to a platform that uses a different radio chip.

LiSP

LiSP [Park and Shin, 2004] is a lightweight security mechanism, which is based on an efficient rekeying technique. LiSP can be used for key management of large as well as small networks. The main features of LiSP includes efficient key broadcast without retransmission/ACK, authentication of key disclosure without incurring additional cost, ability to detect and recover lost keys, key refreshment without disrupting ongoing data encryption/decryption. LiSP uses novel rekeying mechanism to periodically renew the shared key to solve the key stream reuse problem and maximize scalability/energy efficiency. It uses stream cipher for its cheap and fast processing, and use inexpensive crypto algorithms for key renewal operation. LiSP is very flexible: it requires very loose time synchronization which means it can operate in less reliable broadcast media. LiSP decomposes the entire network into clusters/sensing groups. Each group selects a node as group head (GH). One of the GH in the group acts as key Distribution (KS) which controls the security of the group. In addition, LiSP uses intrusion detection system (IDS) to find malicious activities within the network.

LiSP addresses the vulnerability of key stream ciphers caused by key reuse with frequently and synchronously updating of the group key. LiSP utilizes broadcast transmission to distribute the group keys and uses one-way key chains to recover from lost keys. LiSP thus requires the use of static administration keys to perform periodic administrative functions. This leaves those keys vulnerable to disclosure.

Security Protocols & M2M Networks

We will first focus on low-complexity algorithms which facilitate a secure encryption of binary data, such as data and ACK packets. It is reasonable to assume that the starting point is symmetric cryptography which rely on symmetric keys. Once a node is compromised, a re-keying is thus required for which we are likely to follow a group-key strategy to keep overhead at bay for an acceptable security level. At a later stage, we aim to explore key management efficient solutions.

2.8 Security Algorithms

Above-described protocols are heavily supported by security algorithms, such as:

- Symmetric cryptography.
- Asymmetric cryptography.
- Hash functions.

- Digital signatures.

We will now detail these algorithms and briefly summarize their properties.

Symmetric cryptography

Symmetric Key Cryptography (SKC) primitives are able to offer the necessary mechanisms for protecting the confidentiality of the information flow in a communication channel. For achieving this confidentiality level is necessary that both the origin and destination share the same security credential (i.e. secret key), which is utilized for both encryption and decryption. As a result, any third-party that does not have such secret key cannot access the information exchange. The security properties of these primitives mainly reside on the complexity of their internal operations and the length of the keys. There are two types of SKC primitives: Block Ciphers and Stream Ciphers. Block Ciphers are more flexible and powerful, while Stream Ciphers are simpler and faster.

Symmetric Block Cipher Algorithms

A symmetric block cipher operates on a group of bits with axed-size (usually 64 or 128 bits), known as blocks. A block of plaintext is transformed into blocks of cipher text by using several rounds of mathematical operations. In all these rounds most block ciphers also have to transform the original key, using a usually complicated process named key schedule. In cases where more than one block has to be encrypted (e.g. a plaintext that is larger than the block size), it is necessary to use a mode of operation (mode, for short). Moreover, if the plaintext is not a multiple of the block size, some modes requires padding that plaintext. Some of them provide confidentiality, and a few of them provide integrity. Concretely, the earliest modes, as ECB (Electronic Codebook), OFB (Output Feedback) or CBC (Cipher Block Chaining), provide only confidentiality. However, modes such as CCM (Counter with MAC), GCM (Galois Counter Mode), OCB (Offset Codebook Mode) and others modes guarantee both security properties by including a Message Authentication Code (MAC). In addition, in order to generate some randomization in the process, these modes, except ECB, needs of an initialization vector (IV) that has to be combined with the original key. A summary of canonical block ciphers follows:

- **Skipjack**. Skipjack [Chung-Wei Phan, 2002] is considered one of the simplest and fastest block ciphers, in comparison with other ciphers such as RC5, RC6 and AES/Rijndael. It was designed by the NSA (U.S. National Security Agency), and declassified in 1998. The primitive uses building blocks of 64 bits and a cryptographic key of 80 bits for encrypting data blocks of 16 bits. Therefore, it is especially suitable for constrained sensor nodes with 16-bit microcontrollers. This primitive needs of 32 rounds to encrypt the plaintext, alternating two stepping rules known as A and B. Both rules can describe as a linear feedback shift register with additional non-linear keyed G permutation (Feistel cipher of

4 rounds). Also, the key schedule of Skipjack is straightforward, i.e. the key is cyclically repeated enough times to fill the key schedule buffer. Besides these obvious benefits regarding the simplicity of the algorithm, this primitive does not provide the same security than others, due to its small key size.

- **RC5**. In 1994, Ron Rivest designed a simple symmetric algorithm named RC5 [Fan et al., 2009]. This primitive has variable parameters both in block size (32, 64, 128 bits), in key size (0 to 2040 bits) and in number of rounds (between 0 and 255), although it is recommended to use a block size of 64 bits, a key size of 128 bits and 20 rounds. On the other hand, its internal operations are quite simple: integer additions, bitwise XOR, and variable rotations, over two b/2 bits registers. In spite of using simple building blocks, resulting on a small code size in their implementation, algorithm is complex and slow, thus not all the highly-constrained devices could support them.

- **AES**. The Advanced Encryption Standard (AES) was designed in 1998 by Joan Daemen and Vicent Rijmen [Sanchez-Avila and Sanchez-Reillol, 2001]. It is a derivative of the Rijndael algorithm, which supports a larger range of block and key sizes ranging between 128 bits and 256 bits. On the other hand, AES works with a fixed block size of 128 bits and a key size of 128, 192 or 256 bits. Depending on the size of the key, the number of rounds is 10, 12 and 14 respectively. For encrypting, AES needs a 4 X 4 array of bytes (known as the state) over a finite field. In each round (except the last round) four distinct stages are executed: AddRoundKey, where the sub key is combined (XOR operation) with the state, Sub-Bytes, where each byte is replaced with its entry in a fixed 8 bits lookup table, ShiftRows, where bytes in each row of the state are shifted cyclically toward the left, and MixColumns, where each column is multiplied with a fixed polynomial p(x).

- **Twofish**. Other block cipher algorithm similar to AES is Twofish [Lai et al., 2002], designed in 1998 by Bruce Schneier, John Kelsev, Doug Whiting, David Wagner, Chris Hall and Niels Ferguson. Twofish uses a block size of 128 bits with large key sizes (up to 256 bits). Internally, it features a complex key schedule, and four different key-dependent 8x8 bits S-boxes. Besides, it uses other elements belonging to other cipher families, concretely the pseudo-Hadamard transform and the maximum distance separable (MDS) matrix 4x4 over GF(2^8).

Symmetric Stream Cipher Algorithms.

A stream cipher is a method of symmetric cryptography that takes as an input a flow of bits of variable size and transforms it into a cipher text. For that purpose, these cryptographic algorithms combine, mainly by using simple operators such as XOR, those input bits with a pseudorandom key flow obtained from a cryptographic key and an initialization vector (IV). It is essential to use such IV in stream ciphers for avoiding situations where one plaintext produces

the same cipher text when encrypted with the same key. A summary of a canonical stream cipher follows:

- **RC4**. One of the better known stream cipher algorithm, which is used by several protocols like Secure Sockets Layer (SSL) or Wi-Fi Protected Access (WPA), is RC4 [Xie and Pan, 2010]. This simple algorithm was designed by Ron Rivest in 1987, and it is also known as ARC4 or ARCFOUR. There are two design decisions that explain its simplicity. Firstly, it minimizes the execution time by using simple operations as XOR, AND, addition and shifting. And secondly, it uses a block size of 8-bit, consuming less memory space than others cryptosystems. Therefore, RC4 is highly suitable for those architectures with extremely limited microcontrollers. However, RC4 must be implemented carefully for avoiding any attacks as the detected in WEP (Wired Equivalent Privacy implemented in Wi-Fi), where the primitive was not properly initialized.

Symmetric-key based schemes are widely used as they are relatively less computation complexity, which are suitable for the limited resource characteristics of the wireless sensor network. However, the shortages of the symmetric key schemes are also obvious. Different schemes may have different weakness such as security strength (resilience), scalability and connection probability.

Asymmetric cryptography

Public key cryptography (PKC), also known as asymmetric cryptography, is a form of cryptography that uses two keys: a key called secret key, which has to be kept private, and another key named public key, which is publicly known. Any operation done with the private key can only be reversed with the public key, and vice versa. This property makes all PKC-based algorithms useful for secure broadcasting and authentication purposes. It is also an invaluable tool for allowing the secure exchange of secret keys between previously unknown partners. The computational cost of calculating the underlying primitive operations had hindered its application in highly-constrained devices, such as sensor nodes. However, at present there are several PKC primitives that are being considered for a sensor network environment. Obviously, it is important to know their properties to determinate if they could be suitable in a constrained architecture. For example, the algorithm RSA is known by offering good cryptographic operations (encrypting and signing), but limited microprocessors must pay a relatively high computational cost. For all these reasons, is necessary to analyze what kind of PKC primitives are more suitable for the sensor nodes.

Hash function

Cryptographic hash functions or hash primitives are utilized in order to compress a set of data of variable length into a set of bits of fixed length. The result is a “digital fingerprint” of the data,

guaranteeing integrity, identified as a hash value. A cryptographic hash function must satisfy two properties: i) Given a hash value h, it should be hard to find a message m such that hash (m) = h; ii) It should be hard to find two different messages m1 and m2 such that hash (m1) = hash (m2). This kind of hash primitives are usually used to build other cryptographic primitives. For example, the Message Authentication Code primitive provides authenticity and integrity in the messages, either by using the CBC mode of operation in a process named CBC-MAC, or by taking advantage of the existence of hash primitives.

A summary of a canonical hash function follows:

- **SHA-1**. An example of hash function is SHA-1 [Docherty and Koelmans, 2011] algorithm which takes 512 bits and returns 160 bits, and operates with operations of additions, rotations, XOR, AND, OR and NOT. Nevertheless, the complexity required for finding a collision is only 2^{63}, hence it is recommended to use other stronger algorithms such as SHA-256. The algorithm SHA-256 takes 512 bits and returns 256 bits.

The Message Authentication Code approach is very important in low-power networks because with only a few bytes of overhead (4 bytes, 8 bytes or sometimes 16 bytes) we are able to verify the authenticity and the integrity of a message. If we also encrypt both message and MAC of the message, we are able to guarantee confidentiality, integrity and authentication that are the cornerstones in embedded networks security services.

Digital signature

Digital signatures [Lee et al., 1995] facilitate the authenticity of the messages. The device in a network may be communicating with the unknown or less familiar device located everywhere. The communication may also require routing through many intermediate points. During the key agreement process, for establishing a secret key, any middlemen can substitute a devices public-key to its public-key and thus results in establishing a shared secret with the device. Therefore, for establishing shared secret using the key agreement algorithm, it is important for device to receive an authenticated public-key from the peer. For authenticated exchange of public-key, Digital Signature and Digital Certificates are used.

Digital signature is a public-key method to verify the authenticity of a received data from the peer. In digital signature, like the key agreement algorithm, a device uses a pair of keys, “sign private-key” and “sign public-key”. Only the device knows its sign private-key whereas the sign public-key is distributed to all the communicating devices. A device signs the message using a signatures algorithm with its sign private-key to generate a signature and any device that has got the access to the sign public-key of the signed device can verify the data with the signature using the signature verification algorithm. If any third party modifies the data or signature, the verification fails. Since only the signed device knows its sign private-key, it will be impossible for any other device to forge the signature.

Security Algorithms & M2M Networks

In order to guarantee CIA we recommend implementing different security solutions. With the use of symmetric cryptography we are able to guarantee confidentiality through encryption, and integrity and authentication through a message authentication code that will be added in the packet frame. A hash function like CBC-MAC used by TinySec is a good solution to guarantee integrity and authentication. Also Zigbee uses link-layer techniques by means of the AES-CCM* algorithm. We are not focus on digital signatures because it is a public-key method, which is an energy-expensive process to guarantee authenticity, thus it is not recommended for low-power networks.

2.9 Security in Industrial Solutions

Taking into account embedded systems, connecting objects, devices and things is clearly an opportunity but also poses serious challenges. The opportunity is in instrumenting and interconnecting the physical world around us and thus allowing it to act intelligently; this is essentially the vision of IBM's Smarter Planet initiative [IBM, 2010]. The main challenges remain in viably networking this large amount of objects given their obvious constraints in power, processing capabilities, memory and size.

Related to "how" connectivity is facilitated, the era of wireless sensor networks (WSNs) had been born and occupied mainly academic circles for more than a decade. Having gathered a great expertise in the area and published hundreds of articles, the academic community has essentially proven that, from a technology point of view, WSNs can be used to connect objects over short distances.

Naturally, various pioneering companies dedicated to WSNs emerged trying to capitalize on the commercial value of the emerging technology. Among the pioneers, were companies like Cross Bow, Dust Networks, Arch Rock and Coronis. They played a central role in the development of the Internet of Things and acted as a first bridge between academic findings and industrial needs. An example of a pertinent academic finding is that cooperation and relaying are great tools to save energy when covering larger geographical areas; an example of industrial need is that security is a must for any of the real-world deployments.

With more and more companies emerging, and thus proving the viability of an IoT, the community realized quickly that a plethora of propitiatory technologies is counterproductive to the vision of a quickly scaling IoT. The emergence of standards developing organizations (SDOs) in the area of short-range low-power low-rate wireless systems has hence been a natural development. The various standardization bodies aimed to creating a common understanding of the architecture, protocols and functionality of the IoT. Developments in SDOs typically reflected industrial needs whilst incorporating findings of the academic community. Examples of said bodies are the IEEE, IETF, HART, ISA, DASH7, among others.

On the longer term, however, we will likely experience another shift in designing the

IoT and/or IoIT (Internet of Intelligent Things). Notably, to be able to truly cover large geographic areas (with often heterogenous devices) is either not possible with known short-range technologies or would simply require too much investment in multihop infrastructure. In addition, applications with mobility, roaming and alike cannot be supported, thus short-cutting large markets, such as car and logistics telemetry. The vision of M2M communication enabled by cellular network connectivity has hence been taking shape in past years, ignited by pioneering developments of Swedish company Maingate in 1998 as well as European manufacturing giants Ericsson and Nokia shortly after. The SDOs dominant in this area are ETSI M2M and 3GPP LTE.

Both "short range systems", in M2M language referred to as *Capillary M2M*, and "long range system", in M2M referred to as *Cellular M2M*, will likely co-exist until (almost) full migration to cellular system will have been achieved. All these issues aforementioned are following discussed.

2.9.1 Capillary M2M Solutions

The short range capillary M2M solutions are being standardized by various SDOs, notably the IEEE, IETF and interest groups relying thereupon.

IEEE Standards Solutions

The IEEE is standardizing the physical and medium access control layers. There are three families facilitating low-power short-range IoT operation, i.e. IEEE 802.15.4 (as used by ZigBee); IEEE 802.15.1 (as used by Bluetooth); and IEEE 802.15.11 (as used by Wifi). We subsequently briefly discuss their role in the capillary M2M ecosystem.

IEEE 802.15.4.

IEEE 802.15.4 [802, 2010, Lopez et al., 2009] is maintained by the IEEE 802.15 working group. IEEE standard 802.15.4 intends to offer the fundamental lower network layers of a type of wireless personal area network (WPAN) which focuses on low-cost, low-speed ubiquitous communication between devices. The emphasis is on very low cost communication of nearby devices with little to no underlying infrastructure, intending to exploit this to lower power consumption even more. Important features include real-time suitability by reservation of guaranteed time slots, collision avoidance through CSMA/CA and integrated support for secure communications. Devices also include power management functions such as link quality and energy detection. IEEE 802.15.4 is the basis for the ZigBee, WirelessHART, and ISA 100.11a specification, each of which further attempts to offer a complete networking solution by developing the upper layers which are not covered by the standard. The following list summarizes the stable and currently evolving versions:

- **IEEE 802.15.4-2006**. It consists today of several different physical layers, all tailored to low power operation. There are two different link layers, i.e. the non-beacon-enabled mode for low traffic and the beacon-enabled mode for medium and high traffic. The former is a traditional multiple access approach used in simple peer networks; it uses standard CSMA for conflict resolution and positive acknowledgements for successfully received packets. The latter is a flexible approach able to mimic the behavior of a large set of previously published WSN link layers, such as framed link layers, contention-based link layers with common active periods, sampling protocols with low duty cycles, and hybrids thereof; it follows a flexible super-frame structure where the network coordinator transmits beacons at predetermined intervals. The link layer is generally very secure, accept that the acknowledgements are sent in clear thus constituting a very serious security hole which has been greatly underestimated by many real-world deployments, including those using ZigBee.

- **IEEE 802.15.4e**. The IEEE 802.15.4e task group is in charge to modify the link layer sub-layer of IEEE 802.15.4 to meet the requirements of various industrial applications overcoming limitations of the current link layers. The application includes factory automation, process automation, intelligent building, asset tracking, and smart grid. This task group has emphasized three major elements: media management to minimize listening costs, improved security mechanisms, and increased link level reliability through the use of multiple channels, especially in the narrow, lower frequency bands. Now, with the 4e standard approaching ratification, IP networks will be able to improve their performance. Security has been taken very seriously, where the loophole of the unsecured acknowledgement has been rectified.

- **IEEE 802.15.4f**. It has been chartered to define new wireless physicals and link layers enhancements required to support active RFID system for bi-directional and location determination applications. An active RFID tag is a device which is typically attached to an asset or person with a unique identification and the ability to produce its own radio signal not derived from an external radio signal. Currently, three PHY layers are under discussion.

- **IEEE 802.15.4g**. The role of IEEE 802.15 Smart Utility Networks (SUN) Task Group 4g is to create a PHY amendment to 802.15.4 to provide a global standard that facilitates very large scale process control applications such as the utility smart-grid network capable of supporting large, geographically diverse networks with minimal infrastructure, with potentially millions of fixed endpoints. It is currently under development.

- **IEEE 802.15.4k**. It addresses applications such as critical infrastructure monitoring. It defines an alternate physical and only those link layer modifications needed to support its implementation. It is fully concentrated on ultra-low power operation, thus allowing for

connectivity where no permanent energy sources are available.

IEEE 802.15.1.

Bluetooth has originally been a proprietary wireless technology developed by Ericsson in 1994 as a wireless alternative to RS-232 data cables. Today Bluetooth is managed by the Bluetooth Special Interest Group with the aim to guarantee true interoperability between Bluetooth-enabled devices; a goal it has not fully lived up to. The power consumption of the various Bluetooth evolutions is too large to consider it as a serious contender for the IoT; however, latest developments into low-power designs may still yield some surprises.

- **IEEE 802.15.1**. Bluetooth Low Energy. Bluetooth low energy is an alternative to the Bluetooth standard that was introduced in Bluetooth v4.0, and is aimed at very low power applications running off a coin cell. It has a communication range of a few dozen meters, also operates in the 2.4GHz ISM band, supports data rates of around 200kbps, and draws less than 15mA in transmission. First chips have appeared in late 2010, such as the TI CC2540.
- **Security Issues**. Bluetooth has some serious security concerns not all of which have been addressed in recent standards revisions. Some of these issues are summarized in [Bouhenguel et al., 2008].

IEEE 802.11.

In 1997 the IEEE adopted IEEE Standard 802.11-1997 [802, 2010], the first wireless LAN (WLAN) standard. This technology is promoted from WiFi Alliance that is a trade association in charge of certifies products if they conform to certain standards of interpretability. Wifi has had a tremendous success in recent years and has also technically been advanced through various amendments. As such, IEEE 802.11 networks are not suitable to low-power networking designs; however, latest developments into low-power solutions may yield some surprises. Notably, if low-power Wifi really takes off, the problem of coverage which IEEE 802.15.4 networks try to overcome by means of multihop will automatically be eliminated.

- **IEEE 802.11 Low Power**. With the growing market for smart objects and wireless sensors, several companies have developed application specific integrated circuits that are optimized for sensing applications. These products achieve a similar power profile as above low power architectures whilst leveraging the huge installed base of over 2 billion Wifi certified devices; a vibrant standard and industry alliance of close to 300 members; well proven encryption, authentication and end to end network security; mature network management systems; etc. Among one of the first companies promoting the concept of low power Wifi was Ozmo Devices. They tune the .11 protocol stack as well as introduce aggressive power saving operations.

- **Security Issues**. The Wifi Protected Access (WPA) security protocol has become the industry standard for securing .11 networks. Using a pre-shared encryption key (PSK) or digital certificates, the WPA algorithm Temporal Key Integrity Protocol (TKIP) securely encrypts data and provides authentication to said networks. TKIP was designed to be a transition between old hardware and new encryption models. The IEEE 802.11i protocol improved upon the WPA algorithm (TKIP) to the new WPA2 [Jyh-Cheng Chen et al., 2005] that uses a better encryption algorithm: Advanced Encryption Standard (AES). As a major step forward, the protocol also specifies more advanced key distribution techniques, which result in better session security to prevent eavesdropping.

IETF Standards Solutions

The Internet Engineering Task Force (IETF) is actually not an SDO since not approved by the US government. It is composed of individuals, not companies. It meets about three times a year, and gathers an average of 1,300 individuals. It enjoys more than 120 active working groups organized into various areas. The general scope of the IETF is *above the wire/link and below the application*. However, layers are getting fuzzy (link & application layers influence routing) and we lately hence experience a constant exploration of edges. There are three working groups pertinent to capillary M2M where we will concentrate on two, i.e. IETF 6LoWPAN (establishing gateway to Internet); and IETF ROLL (facilitating routing in low-power network). We subsequently briefly discuss their role in the capillary M2M ecosystem.

IETF 6LoWPAN.

IPv6 over Low power WPAN (6LoWPAN) acts as a simplified gateway between the low power embedded network and the Internet. It facilitates neighborhood discovery, header compression with up to 80% compression rate, packet fragmentation (1260 byte IPv6 frames $\rightarrow$ 127 byte IEEE 802.15.4 frames), and thus direct end-to-end Internet integration. However, it does not provide routing. Security is also catered for [Barker, 2010].

IETF ROLL.

Routing Over Low power and Lossy networks (ROLL) deals with the design of a routing protocol for wireless low power mesh networks. It is in its final stage of standardization. It is based on a gradient routing protocol where nodes acquire a rank based on the distance to the collecting node and the messages follow the gradient of ranks to reach the destination. Again, security is currently being catered for [Tsao et al., 2012].

WirlessHART Standard Solution

WirelessHART (Highway Addressable Remote Transducer) [WirelessHART, 2010, Lopez et al., 2009] is an open-standard wireless networking technology developed by HART Communication Foundation. It is the wireless version of the HART protocol, which is the most used in the automation and industrial applications which require real time responses. The protocol utilizes a time synchronized, self-organizing, and self-healing mesh architecture. The protocol currently supports operation in the 2.4 GHz ISM Band using IEEE 802.15.4 standard radios. With respect to the stack of WirelessHart, the physical layer is based on the IEEE 802.15.4-2006 whereas the link layer has been modified to meet the industrial needs. Its link layer is based on TSMP and similar to IEEE 802.15.4e with the only difference that a set of time/frequency hopping patterns are fixed. The frequency hopping approach allows to mitigate fading and interferences in the communication channel.

In WirelessHART, the Security Manager is responsible for the generation, storage, and management of the keys that are used for device authentication and encryption of data. In order to provide authentication WirelessHart provides the MAC that is generated with CCM* (counter with CBC-MAC) using the AES-128 algorithm. For its generation it is necessary to include a 128-bit key, a nonce of 13 bytes and the message header without encryption. Public, Join, Network and Session Keys must be provided from the WirelessHART Network Manager:

- **Network key**: it is shared by all network devices and is used to generate the MAC on the link layer.
- **Public key**: it is pre-configured in every node and is used to provide authentication during joining process; in this case network key cannot be used because it is delivered from the security manager after the first authentication.
- **Join key**: it is pre-configured in every new node and whenever a new node joins in the network it will be authenticated by the network manager that will send to it the network and the session keys.
- **Session key**: it is the unique key between two network devices. It is used to provide confidentiality and integrity to any interchanged messages in order to ensure privacy to end-to-end communication. The delivery of this key is managed by the security manager.

which is able to provide secure links core to the WirelessHART design.

ISA 100.11a Standard Solution

ISA100.11a [100.11a, 2010, Lopez et al., 2009] is a wireless communication standard aiming to provide reliable and secure operation for non-critical monitoring, alerting, and control applications specifically focused to meet the needs of industrial users.

ISA100.11a defines a subset of the OSI stack and an organization structure of permitted networks, system management, gateway, and security specifications for low-data-rate wireless connectivity with fixed, portable, and moving devices, including support for very limited power consumption.

ISA100.11a utilizes the 802.15.4 physical layer, provides extensions to the 802.15.4 link layer and defines network layer through application layer functions and services. The medium and part of the data link layer is based on IEEE 802.15.4 2.4GHz DSS physical and extends the 802.15.4 MAC layer including methods for channel hopping, TDM based bandwidth management, mesh networking (forming, routing and discovery support). The network layer is based on IETF RFC 4944 [Network layer, 2007] (transport of IPv6 packets over IEEE 802.15.4) with constraints to focus on security and low power; network layer services include address translation (and compression), fragmentation and routing. The Transport layer is based on UDP per RFC4944, and includes security services. The Application layer provides objects model and object-to-object communication services. Key goals are robustness in harsh industrial applications, coexistence in the presence of other wireless services, and low cost/low complexity deployment. Security services are extended throughout the entire stack and are based on the security offered by IEEE 802.15.4-2006 with symmetrical and asymmetrical keys, configuration, operation and maintenance.

ZigBee Alliance

ZigBee, created by the ZigBee Alliance, is a set of recommendations to facilitate interoperability between wireless low power devices. The relationship between IEEE 802.15.4 and ZigBee is similar to that between IEEE 802.11 and the Wifi Alliance. ZigBee relies today on the physical and link layers of IEEE 802.15.4, will shortly rely on the networking layer of the IETF 6LoWPAN (and likely ROLL), and then builds its industrial profiles on top. We shall briefly focus on the security services offered for its most popular modes, i.e. standard security mode for ZigBee stack 1; and high security mode for ZigBee stack 2 (ZigBee PRO). However, no matter how secure the system is, the reliance on the insecure IEEE 802.15.4 link layer makes it a vulnerable design choice.

- **Standard security mode in Zigbee**. The ZigBee standard security mode, which is based on a 128-bit AES algorithm, adds to the security model provided by IEEE 802.15.4 three security services methods: frame protection, key establishment and transport and also device management. Two types of frame protection can be applied, i.e. network and application level security. Once network-level security has been set up, application-level security can be set up for an individual pair of nodes. Application-level security is used when the communications between the two nodes must remain private from the rest of the network. This requires their communications to be encrypted/decrypted using an application link key which is unique to the pair. In order to set up application-level security

between two nodes, a specific function must be called on one of the nodes to request an application link key from the Trust Center. The Trust Center responds to this request by sending the same application link key to both nodes.

- **High security mode in ZigBee PRO**. In the "High" security mode of ZigBee PRO, the Trust Center maintains a list of devices, link keys and network keys that it needs to control and enforce the policies of network key updates and network admittance. In addition this security mode provides also the use of master keys: these optional keys are not used to encrypt frames while they are used as an initial shared secret between two devices when they perform the Key Establishment Procedure (SKKE) to generate Link Keys. Thus with these kinds of keys, a couple of node are able to deal with a secret key (link key) without the supervision of a Trust Center and consequently saving energy. Furthermore, protections for device authentication and key management and distribution, including the use of the SKKE (Symmetric-Key Key Exchange) and PKKE (Public-Key Key Exchange) are provided, among other security features. However, no matter how secure the system is, the reliance on the insecure IEEE 802.15.4 link layer makes it a vulnerable design choice.

DASH7 Alliance

DASH7 is the name of a technology promoted by the DASH7 Alliance. It is an emerging embedded low power networking technology using the ISO/IEC 18000-7 standard for active RFID, operating at in the 433MHz unlicensed spectrum. DASH7 provides multi-year battery life, range of up to 2km, low latency for tracking moving objects, small protocol stack, sensor and security support, and data transfer up to 200 kbit/s. It has found interest in military circles too where the US DoD (Department of Defence) awarded a $429 million contract for DASH7 devices, making it one of the largest wireless sensor networking deployments in the world.

ANT+ Alliance

The ANT+ Alliance has now more than 300 industrial members. It promotes an a priori proprietary protocol for ultra-low power networking applications in sport, wellness, home and industrial automation. It operates for up to three years on a coin cell battery. It handles peer-to-peer, star, tree and practical mesh topologies. It operates at 2.4GHz, provides data security through a 64bit network key with additional application layer security, supports up to 2^{32} addressable devices, among others. It is manufactured by giants like Texas Instruments through its CC257x family.

WAVE2M Community

WAVE2M technology relies on ultra-low power RF components allowing for decade long battery-driven operation with applications. This community results on an independent standards

alliance whose participants work together to define the WAVE2M technology roadmap and to deliver new WAVE2M features and capabilities, as determined by member interests and market requirements. Based on WAVE2M features and capabilities, similar to the ZigBee profiles, all new WAVE2M adopters can define their own WAVE2M profiles to meet specific application requirements: frequency bands, data rate, output power, channel bandwidth, network topology, self-routing and self-healing options, etc. The work is driven by the Technical Committee, composed of the four Physical/Link layers, Network layer, Application and security working groups.

2.9.2 Cellular M2M Solutions

Cellular M2M technology developments are commencing to take momentum, with many companies and various SDOs envisioning future IoT applications to run over such networks. From a rate and range point of view, current cellular systems already meet the M2M requirements; however, from a power consumption point of view, many issues remain open. We will thus briefly discuss various cellular M2M initiatives.

ETSI M2M

ETSI M2M is composed by various manufacturers, operators and service providers, among others. ETSI typically provides the framework, requirements and architecture, whereupon technologies such as 3GPP or IEEE can be used to populate the developed architecture. The work is organized in stages:

- **Stage 0**: Use cases documents. Several use case documents have been developed in parallel, such as M2M requirements for smart metering, health applications, etc.
- **Stage 1**: Services requirements. The thus resulting service requirements have then been developed which aims to unify the requirements of the different use cases documents.
- **Stage 2**: Architecture. Here, capabilities and interfaces are developed, as well as message flows, etc.
- **Stage 3**: Refinement. In this stage, the architecture is refined to meet the prior outlined user requirements.

ETSI M2M currently (Q1 2011) also works on security requirements which influence the entire M2M architectural design.

3GPP LTE-M

The concept of M2M has been born out from 2G cellular systems and, early adopters of GSM/GPRS data plans, clearly demonstrated the its value. 3GPP thus naturally issued in January

2007 a technical report TR 22.868 “Study on Facilitating Machine to Machine Communication in 3GPP Systems” which identified that a huge market potential for M2M beyond the current market segment. However, due to CDMA-based 3G systems not being suitable to low power operations, there have been little developments until recently. With OFDM-based LTE on the horizon, cellular M2M has suddenly become of interest again and a set of further documents has been issued lately, e.g. TS 22.368 “Service Requirements for Machine-Type Communications (MTC)" and TR 23.888 “System Improvements for MTC”.

Not all MTC applications have the same characteristics and not every optimization is suitable to all applications; therefore, features are defined to provide some structure to the customer and the network is then tuned accordingly to needs. These features are offered on a per subscription basis and include items such as Low Mobility, Time Controlled, Time Tolerant, Packet Switched only, Small Data Transmissions, Mobile originated only, Infrequent Mobile Terminated, MTC Monitoring, Priority Alarm Message (PAM), Secure Connection, Location Specific Trigger, Network Provided destination for Uplink Data, Infrequent transmission, Group Based Policing, Group Based Addressing, etc.

Whilst the potential and market value are clear, technical problems, mainly in the area of low-power consumption, support of large amount of nodes and low delays, still remain. These and other problems are currently being addressed by the 3GPP as well as one of the largest EU projects EXALTED [M2M communication, 2010].

Chapter 3

Secure lossless data aggregation for M2M networks

"Poor is not the one who has little, but those who want more"
Seneca.

Contents

In many embedded M2M applications, the data collected from individual meters is aggregated at the final gateway to perform the requested service. In order to not overload the network with unnecessary messages, and thus not waste energy and facilitate wireless communications, many systems also perform in-network aggregation of sensor data at intermediate nodes en route to the final gateway. Most existing aggregation algorithms and systems do not include any provisions for security, and consequently these systems are vulnerable to a wide variety of attacks. In particular, compromised nodes can be used to inject false data that leads to incorrect aggregates being computed at the final gateway. In this Chapter we present the proposed secure lossless data aggregation protocol that not only provides an efficient tool to save M2M network's energy but guarantees security services as well.

3.1 Introduction to Data Aggregation Issue in M2M Networks

The main objective of data aggregation techniques is to increase the network lifetime and to improve the effectiveness of wireless channel communications. Network lifetime is increased by reducing the energy consumption of the entities involved in the communication system by decreasing the number of bytes for sending/receiving, while the effectiveness of the wireless communications can be improved by decreing the bandwidth usage; less messages exchanged thus more likely to find the wireless channel free. However, increasing network lifetime, data aggregation protocols may degrade important quality of service metrics in wireless networks, such as data accuracy, latency, fault-tolerance, and security. Therefore, the design of an efficient data aggregation protocol is an inherently challenging task because the protocol designer must trade between energy efficiency, data accuracy, latency, fault-tolerance, and security. In order to achieve this trade-off, data aggregation techniques are tightly coupled with how packets are routed through the network. Redundant packets should be avoided and useless overheard should be decrease. This Section explores data aggregation concept presenting topologies and general taxonomy, analyzes security in this context as well as presents the general features of the most used data aggregation solutions studied on the literature.

3.1.1 Data Aggregation Topology

Hence, the application and the architecture of the wireless network plays a vital role in the performance of different data aggregation protocols; data aggregation should be selected after have define network deployment. There are several protocols that allow routing and aggregation of data packets simultaneously. These protocols can be categorized into two categories: tree-based data aggregation protocols and cluster-based data aggregation protocols. Earlier work

on data aggregation focused on improving the existing routing algorithms so as to make data aggregation possible. As a result, many data aggregation protocols based on shortest path tree structure are following discussed. To reduce the latency due to tree-based data aggregation, recent work on data aggregation tends to group network nodes into clusters so that data are aggregated in each group for improved efficiency.

- **Tree-based data aggregation**. These techniques aim to provide an energy efficient data aggregation tree. Figure 3.1 illustrates an example of tree-based data aggregation. Greedy Incremental Tree (GIT) is a data-centric routing protocol that allows data aggregation in conjunction with Directed Diffusion routing protocol.

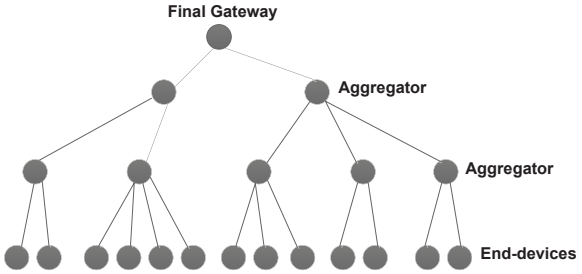

Figure 3.1: Tree-based data aggregation scheme.

- **Cluster-based data aggregation**. In these methods, nodes are subdivided into clusters. In each cluster, a cluster head is elected in order to aggregate data locally and transmit the aggregation result to the final gateway. Cluster heads can communicate with the final gateway directly via long range radio transmission or taking advantage of other communication systems, such as Internet or WiMAX. However, this is quite inefficient considering the limited resources of the nodes. Thus, cluster heads usually form a tree structure to transmit aggregated data by multi-hopping through other cluster heads which results in significant energy savings. Despite these savings, selecting a unique node as cluster head may provide vulnerability to the network by giving a single point of attack for malicious activities and thus facilitating the potential isolation of part of the network. Figure 3.2 presents an example of cluster-based data aggregation.

In large low-power networks, computing aggregates in-network, i.e., combining partial results at intermediate nodes during message routing, significantly reduces the amount of communication and hence the energy consumed. An approach used by several data acquisition systems is to construct a spanning tree rooted at the final gateway, and then perform in-network aggregation along the tree. Partial results propagate level-by-level up the tree, with each node awaiting messages from all its children before sending a new partial result to its parent. Researchers have designed several energy-efficient algorithms that can be used for computing aggregates using the tree-based approach.

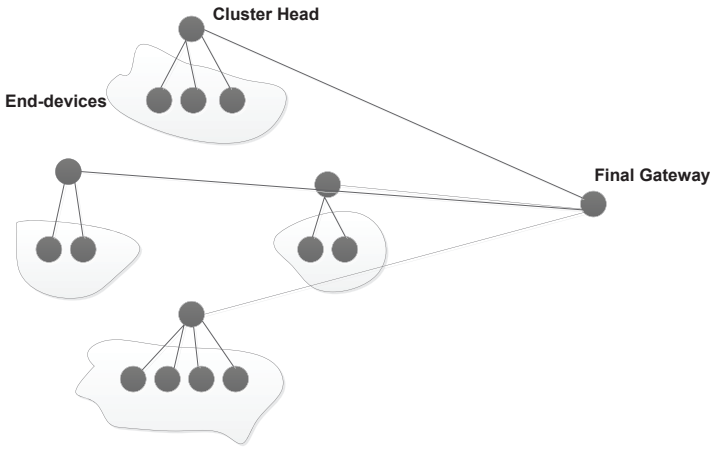

Figure 3.2: Cluster-based data aggregation scheme.

Tree-based aggregation approaches, however, are not robust to communication losses which result from node and transmission failures that are relatively common in wireless networks. Because each communication failure loses an entire sub tree of readings, a large fraction of sensor readings are potentially unaccounted for at the final gateway, leading to a significant error in the aggregate computed. To address this problem, researchers have proposed novel algorithms that work in conjunction with multipath routing for computing aggregates in lossy networks.

3.1.2 Taxonomy of Data-Aggregation Schemes

Data acquisition system for low-power networks can be classified into two broad categories on the basis of the data collection methodology employed for the application in place:

- **Query-Based Systems**. In query-based systems, the gateway broadcasts a query to the network and the nodes respond with the relevant information. Messages from individual nodes are potentially aggregated en route to the final destination. Finally, the gateway computes one or more aggregate values based on the received messages.

- **Event-Based Systems**. In event-based applications, such as monitoring and surveillance, nodes send a message to the gateway only when the target event occurs in the area of interest. If multiple reports being relayed correspond to the same event, they can be combined by an intermediate node on the route to the final destination.

In addition to data collection methods, data aggregation systems can be classified on how data is aggregated as well. In single-aggregator approaches, aggregation is performed only at the final destination point. On the other hand, hierarchical or multi-aggregator aggregation approaches make use of in-network aggregation; hierarchical aggregation schemes can be further

classified into tree-based schemes and ring-based schemes on the basis of the topology into which nodes are organized:

- **Single Aggregator Approach**. In this first model, the aggregation process takes place once between the devices and the final destination point. In other words, all individual collected data travels to only one aggregator point in the network before reaching the querier. This aggregator node should be powerful enough to perform the expected high computation and communication processes. The main role of the data aggregation might not be satisfied fully since redundant data will still travel in the network for a while until they reach the aggregator. This model is useful when the network is small or when the querier is not in the same network. However, large networks are not suitable places to implement this model especially when data redundancy at the lower levels is high. The data aggregation schemes that fit in this model can be divided into two categories: whether they have a verification phase or not; integrity verification test.
  - **Verification Phase**. This phase enhances the querier's ability to distinguish between the valid and invalid aggregated readings.
  - **No Verification Phase**. Data integrity has not been considered by the scheme's designers.
- **Multiple Aggregator Approach**. In this second model, collected data in the embedded network are aggregated more than one time before reaching the last destination. This model achieves greater reduction in the number of bits transmitted within the network especially in the large communication systems. The importance of this model appears as the network size is getting bigger especially when data redundancy at the lower levels is high. Again, the data aggregation schemes that fit in this model can be divided into two categories: whether they have a verification phase or not.
  - **Verification Phase**. Secure data aggregation scheme that contains a verification phase to enhance the querier ability in distinguishing between the valid and invalid aggregated readings. This phase is more complicated than the same phase in the single aggregator model since the data is aggregated many times at different aggregation points. The querier is interested to know whether the final aggregated result is altered or not and, in case it was modified, in which network point; complex Message Authentication Code or Signature use.
  - **No Verification Phase**. Again, data integrity has not been considered by the scheme's designers.

3.1.3 Security in Data Aggregation Schemes

As discussed above, most existing data management and acquisition systems for low-power communication systems are vulnerable to security attacks launched by malicious parties. Smart

devices are often deployed in unattended environments and leaved there for the entire network lifetime; hence they are vulnerable to physical tampering. Since current hardware solutions are not supported by secure tamper resistance mechanisms, it is relatively easy for an adversary to compromise a node without being detected. The adversary can obtain confidential information (e.g., key materials) from the compromised sensor and reprogram it with malicious code. Moreover, the attacker can replicate the compromised node and deploy the replicas at various strategic locations in the network.

To prevent unauthorized nodes from eavesdropping on or participating in communications between legitimate nodes, one can augment the aggregation and data collection systems with any one of several recently proposed authentication and encryption protocols. However, securing aggregation systems against attacks launched by compromised nodes is a much more challenging problem since standard authentication mechanisms cannot prevent insider attacks launched by a compromised node; when key materials is no longer a common secret among valid parties, cryptographic techniques are useless.

A compromised node can be used to launch a variety of security attacks. These attacks include jamming at the physical or link layer as well as other resource consumption attacks at higher layers of the network software. Compromised nodes can also be used to disrupt routing protocols and topology maintenance protocols that are critical to the operation of the network. By using a few compromised nodes to render suspect the data collected at the final destination, an adversary can effectively compromise the integrity and trustworthiness of the entire network.

Taking into account event-based systems, compromised nodes can be used to send false event reports to the final gateway with the goal of raising false alarms and depleting the energy resources of the nodes in the network; this is related to the false data injection attack. Similarly, in query-based systems, compromised nodes can be used to inject false data into the network with the goal of introducing a large error in the aggregate value computed at the data sink. The aggregate computed is erroneous in the sense that it differs from the true value that would have been computed if there were no false data values included in the computation.

In an aggregation-system, a compromised node can corrupt the aggregate value computed at the final destination in four ways. First, it can simply drop aggregation messages that it is supposed to relay towards the sink. This has the effect of omitting a large fraction of sensor readings being aggregated; second, it can alter a message that it is relaying to the data sink; third, it can falsify its own sensor reading with the goal of influencing the aggregate value; fourth, in systems that use in-network aggregation, it can falsify the sub-aggregate which it is supposed to compute based on the messages received from its child nodes.

The first attack in which a compromised node intentionally drops aggregation messages can substantially change the final estimate of the aggregate if tree-based aggregation algorithms are used. The deviation will be large if the compromised node is located near the root of the aggregation hierarchy because a large fraction of data will be omitted from being aggregated. Countermeasures against this attack include the use of multi-path routing, ring-based topologies,

the use of probabilistic techniques in the formation of aggregation hierarchies as well as watchdog techniques; observation of the compromised node behavior by neighboring valid nodes. To prevent the second attack in which a compromised node alters a message being relayed to the final destination, it is necessary for each message to include a Message Authentication Code generated using a key shared exclusively between the originating node and the gateway. This MAC enables the gateway to prove the integrity of a message, and filter out messages that have been altered. Hence, the end-effect of altering a message is no different from dropping it, and countermeasures such as multi-path routing are needed to mitigate the effect of this attack. The third attack in which an entity intentionally falsities its own reading is referred to as the falsified focal value attack. This attack is similar to the false data injection attack in event-based systems. Potential countermeasures to this attack include approaches used for fault tolerance such as majority voting and reputation-based frameworks. The three attacks discussed above apply to both single-aggregator and hierarchical aggregation systems, whereas the fourth attack applies only to hierarchical aggregation systems. This attack in which a node falsifies the aggregate value it is relaying to its parent(s) in the hierarchy is much more difficult to address. This attack is referred to as the falsified sub-aggregate attack. Watchdog techniques or complex verification tests can mitigate the effectiveness of this fourth attack.

3.1.4 Data Aggregation State-Of-Art

The interaction of data aggregation and cryptographic methods has to be coordinated. This subsection thus attempts to compare the secure data aggregation even if comparisons of security schemes can be difficult since the designers solve secure aggregation from different angles. Therefore, these schemes are compared in a number of different ways: security services provided, cryptographic primitives used, and resilience against attacks.

SDA is the first data aggregation that was proposed by Hu & Evans [Hu and Evans, 2003]. This protocol was proposed to study the problem of data aggregation once one node is compromised. This protocol achieves resilience against a node compromise by delaying the aggregation and authentication at the upper levels; authentication with μ-TESLA solution. Therefore, sensors measurements are forwarded unchanged and then aggregated at the second hop instead of aggregating them at the immediate next hop. Thus, the sensor needs to buffer the data to authenticate it once the shared key is revealed by the network coordinator. Moreover, the proposed scheme only offers data integrity, freshness and authentication. Even though it increases the confidence in the sensor readings integrity the data can be altered once a parent and child in the hierarchy are compromised. Once a compromised node is detected, no practical action is taken to reduce the damage caused by this compromise which affects the data availability in the network. Much worse, once a grandfather node detects a node compromise, it could not decide whether the cheating node is the child or the grandchild.

In order to improve SDA scheme, ESA was proposed by Jadia & Mathuria [Jadia and Mathuria, 2005]. ESA scheme, instead of using μ-TESLA to authenticate the base stations

broadcast in the validation process to reveal the shared key with sensors, it uses one-hop pair-wise keys (to encrypt data between a node and its parent) and two-hop pair-wise keys (to encrypt data between a node and its grandparent). This will improve the secure aggregation scheme by adding data confidentiality and reducing the memory overhead since data does not need to be stored until the key is revealed. However, the system will still be broken as soon as two consecutive nodes in the hierarchy are compromised.

Przydatek et al. [Chan et al., 2007] proposed a secure information aggregation (SIA) framework for WSNs called aggregate-commit-prove. This framework provides resistance against a special type of attack called stealthy attacks aggregate manipulation where the attacker's goal is to make the user accept false aggregation results without revealing its presence to the user. It consists of three node categories: a home server, a base station, and sensor nodes. SIA assumes that each sensor has a unique identifier and shares a separate secret cryptographic key with both the home server and the aggregator. The keys enable message authentication and encryption if data confidentiality is required. Moreover, it assumes that the home server and base station can use a mechanism, such as μ-TESLA, to broadcast authentic messages. SIA consists of three parts: collecting data from sensors and locally computing the aggregation result, committing to the collected data, and reporting the aggregation result while proving the correctness of the result. SIA offers data integrity, authentication, data freshness, and confidentiality (if required).

A witness based data aggregation (WDA) scheme for the WSN is being proposed by Du et al. [Du et al., 2003] to assure the validation of the data sent from an aggregator node to the base station. In order to prove the validity of the aggregated result, the aggregator node has to provide proofs from several witnesses. A witness is one who also performs data aggregation like the aggregator node, but does not forward its result to the base station. Instead, each witness computes the message authentication code of the result and then sends it to the aggregator node which must forward the proofs to the base station. WDA offers only integrity property to the data aggregation security and this is required to send multiple copies similar to the original aggregated result, to the aggregator point. Thus, the aggregator point must forward these reports as well as the aggregated result to the base station. Since the aggregator point is fixed and responsible to handle so much traffic, the aggregator resources will not last long.

Wagner [Wagner, 2004] proposed a mathematical framework (RA) for evaluating the security of several resilient aggregation techniques. The study measures how much damage an adversary can cause by compromising a number of nodes and then using them to inject erroneous data. Wagner has described a number of better methods for securing the data aggregation such as how the median function is a good way to summaries statistics. Furthermore, Wagner claimed that trimming and truncation can be used to strengthen the security of many aggregation primitives by eliminating possible outliers. However, this work only focused on examining the received aggregated data (at the base station) without studying how these data are aggregated. Thus, when the network size increases, the communication cost will be very high for the

transmission of all the sensor readings to the base station. Moreover, eliminating abnormal data with no further reasoning is impractical especially for applications such as monitoring bush-fire.

Moreover, Mahimkar & Rappaport [Mahimkar and Rappaport, 2004] have improved the data integrity vulnerability in SDA and ESA by signing the aggregated data in SecureDAV. In this schem, each sensor within a cluster will have its share of its secret cluster key to generate a partial signature on the aggregated data. Once an aggregator receives sensor readings in the same cluster, it aggregates them and broadcasts the average value of the readings. Each sensor in the cluster compares its reading with the average value received from the aggregator. Then, it partially signs the average value only and only if the difference between the received average value and its reading is less than a certain value (threshold). Then, the aggregator (cluster-head) combines partial signatures to form a full signature of the aggregated results and sends it to the base station. SecureDAV provides data confidentiality, data integrity, and authentication. The drawbacks of this scheme are: it requires high communication costs on data validation, and supports only the AVG aggregation function.

Yang et al. [Yang et al., 2006] proposed a secure hop-by-hop data aggregation protocol (SDAP) that can tolerate more than one compromised node. SDAP is based on two principles: divide-and-conquer and commit-and-attest. In order to reduce the damage caused by compromising an aggregator at a high level in the per-hop aggregation scheme, SDAP uses the divide-and-conquer principle to divide the network tree into multiple logical sub-trees which increases the number of aggregators and reduces the number of nodes in each sub-tree. Consequently, the damage caused by compromising an aggregator of a sub-tree is reduced. The other principle, that is commit-and-attest, enhances the ordinary hop-by-hop aggregation scheme by adding a commitment property, and helps the base station to prove the correctness of the aggregated data. Once an aggregator of a logical sub-tree commits its aggregation result, it cannot deny it later on. This scheme needs to send much data to ensure reasonable level of security.

Furthermore, Chan et al. [Chan et al., 2006] extended the work in SIA by applying the aggregate-commit-prove framework in a fully distributed network instead of single aggregator model. In general, this scheme (SHDA) offers exactly what the SIA does data integrity, authentication, and confidentiality. Each parent sensor performs an aggregation function whenever it has heard from its child nodes. In addition, it has to create a commitment to the set of the input used to compute the aggregated result by using a Merkle hash tree. Then, it forwards the aggregated data and the commitment to its parent until it reaches the base station. Once the base station received the final commitment values, it rebroadcasts them into the rest of the network in an authenticated broadcast. Each node is responsible for checking whether its contribution was added to the aggregated data or not. Once its readings are added, it sends an authentication code to the base station. For communication efficiency, the authentication codes are aggregated along the way to the base station. However, missing one authentication code for any reason leads the base station to reject the aggregated result. Furthermore, noticeable delay, too many transmissions and computation costs in terms of energy consumption will be added as

consequences of adding security to the scheme.

Sanli et al. [Sanli et al., 2004] developed a new data aggregation technique called the Secure Reference-Based Data Aggregation scheme (SRDA) that sends only the difference between sensed data and the reference value (called differential value) instead of raw data. Deference value is taken as the average value of previous sensor readings. In SRDA scheme, each sensor computes the differential data (sensed data reference value), encrypts it, and then sends it to the cluster-head. The authors claim that the security level of the network should be gradually increased as the data is traveled to higher level cluster-heads. Therefore, they suggest using a cryptographic algorithm (RC6) with adjustable parameters such as the number of rounds, to achieve different level of security in the WSN. Increasing or decreasing the number of rounds changes the security strength of the RC6 that can be measured by the security margin. The security margin is the deviation of the actual number of rounds from the minimum number of rounds for which the algorithm is considered to be secured. The SRDA uses a higher security margin at higher level cluster-heads compared to low level cluster-heads.

Moreover, the problem of aggregating encrypted data in the WSN is being addressed in Westhoff et al. [Westhoff et al., 2006]. The proposed protocol, called Concealed Data Aggregation (CDA), uses an additive and multiplicative homomorphic encryption scheme that allows the aggregator to aggregate encrypted data. In their studies, the authors argued that the security level it's still reasonable and the privacy homomorphism (PH) method helps to implement encryption in the WSN, although it can prove insecurity against chosen plain text attacks. However, they admitted that the encryption in CDA is very expensive and adds between 0%-22% additional data overhead compared to RC5 which increases the power consumption of the sending node. Generally speaking, CDA ensures only data confidentiality.

Furthermore, a new secure data aggregation scheme based on homomorphic encryption (EDA) is proposed by Castelluccia et al. [Castelluccia et al., 2005]. This scheme allows an aggregator to execute the aggregation function and aggregate the encrypted data that are received from its children with no need for decryption and to recover the original messages. It uses a modular addition instead of the xor (Exclusive-OR) operation that is found in the stream ciphers. Thus, even if an aggregator is being compromised, original messages cannot be revealed by an attacker. The authors claimed that the provided privacy protection by this scheme is comparable to the privacy protection that is provided by a scheme that performs end-to-end encryption with no aggregation. However, they admit that their proposed scheme generates significant overhead if the network is unreliable since sensors' identities of non-responding nodes must be sent together with the aggregated result to the base station. More importantly, this scheme concerns only one security property which is data confidentiality.

Wu [Wu et al., 2006] uses the cryptographic algorithms only when a cheating activity is detected. Topological constraints are introduced to build a secure aggregation tree (SAT) that facilitates the monitoring of data aggregators. In SAT, any child node is able to listen to the incoming data of its parent node. When the aggregated data of a data aggregator are

questionable, a weighted voting scheme is employed to decide whether the data aggregator is properly behaving or is cheating. If the data aggregator is a misbehaving node, then SAT is rebuilt locally so that the misbehaving data aggregator is excluded from the aggregation tree.

The authors of SELDA, Ozdemir et al. [Ozdemir, 2007], argue that compromised nodes have access to cryptographic keys that are used to secure the aggregation process and therefore cryptographic primitives alone cannot provide a sufficient enough solution to secure data aggregation problem. Based on this observation, the authors propose a Secure and rELiable Data Aggregation protocol, called SELDA which makes use of a web of trust. The basic idea behind SELDA is that sensor nodes observe actions of their neighboring nodes to develop trust levels (trustworthiness) for both the environment and the neighboring nodes. These misbehaviors are quantified as trust levels using Beta distribution function. Sensor nodes exchange their trust levels with neighboring nodes to form a web of trust that allows them to determine secure and reliable paths to data aggregators. Moreover, to improve the reliability of the aggregated data, data aggregators weigh the received sensors' data by using the web of trust concept. One important property of SELDA is that, due to the monitoring mechanisms in use, it is able to detect if a data aggregator is under DoS attack. The simulation results show that SELDA increases the reliability of the aggregated data at the expense of a tolerable communication overhead.

Ozdemir in [Ozdemir, 2008] presents a study where the authors improved the main idea of SELDA by introducing functional reputation concept where each functional reputation value is computed over sensor node actions with respect to that function. Hence, security of data aggregation process is ensured by selecting trusted data aggregators using aggregation functional reputation and by weighting sensor data using sensing functional reputation. The simulation results show that using functional reputation is more effective than using general reputation when evaluating the trustworthiness of a sensor node.

Zhang [Zhang et al., 2007] in his paper comments the fact that existing privacy homomorphism based in-network processing protocols can only work for some specific query-based aggregation functions, e.g., sum, average, etc. Hence, instead of privacy homomorphism, the authors take advantage of digital watermarking and propose an end-to-end statistical approach for data authentication that provides inherent support for data aggregation. The novel idea of this work is to modulate authentication information as watermark and superpose this information on the sensory data at the sensor nodes. The watermarked data can be aggregated by the intermediate nodes without incurring any en route checking. In order to check whether the data has been altered by the compromised nodes, upon reception of the sensory data, the data sink is able to authenticate the data by validating the watermark. More specifically, the proposed technique visualizes the sensory data gathered from the whole network at a certain time snapshot as an image, in which every sensor node is viewed as a pixel with its sensory reading representing the pixels intensity. Since senor data is represented as an "image" digital watermarking can be applied to this image. In order to balance the energy consumption among sensor nodes, a direct

spread spectrum sequence (DSSS) based watermarking technique is used. While each sensor node appends a part of the whole watermark into its sensory data, verification of watermark which requires an extensive computational resource is left to the sink. The proposed scheme adopts the existing image compression schemes as the aggregation functions to reduce network load while retaining the desired details of the data. Moreover, using a DSSS based watermarking scheme, the proposed technique is enabled to survive a certain degree of distortion and therefore naturally support data aggregation.

Wang P. [Wang et al., 2007] proposes a joint data aggregation and encryption scheme using Slepian-Wolf coding for efficient and secured data transmission in clustered wireless sensor networks. He first considers the optimal intra-cluster rate allocation problem using Slepian-Wolf coding for data aggregation, which aims at finding a rate allocation subject to Slepian-Wolf theorem such that the total energy consumed by all sensor nodes in a cluster for sending encoded data is minimized. Based on the properties of Slepian-Wolf coding with optimal intra-cluster rate allocation, a novel encryption mechanism, called spatially selective encryption, is then proposed for data encryption within a single cluster. This encryption mechanism only requires a cluster head to encrypt its data while allowing all its cluster members to send their data without performing any encryption. In this way, the data from all cluster members can be protected as long as the data of the cluster head (called virtual key) is protected. This can significantly reduce the energy consumption for performing data encryption. Furthermore, an energy-efficient key establishment protocol is also proposed to securely and efficiently establish the key used for encrypting the data of a cluster head. Simulation results show that the joint data aggregation and encryption scheme can significantly improve energy efficiency in data transmission while providing a high level of data security. However the drawbacks of this scheme are: use of cluster-head, with a jamming attack to this node is possible to block the traffic to the sink node, and establishes a key distribution mechanism that has to guarantee the exchange of a secret key, a master key, between a cluster-head and the sink node in a hostile communication environment.

Castelluccia in [Castelluccia et al., 2009] proposes a new homomorphic encryption scheme that allows intermediate sensors to aggregate encrypted data of their children without having to decrypt. As a result, even if an aggregator is compromised, it cannot learn the data of its children, resulting in much stronger privacy than a simple aggregation scheme using hop-by-hop encryption. He shows that, if key streams are properly derived, the scheme can achieve semantic security against any node collusion of size less than the total number of nodes. They also evaluate the performance of their scheme and show, as expected, that it is slightly less bandwidth-efficient than the naive hop-by-hop scheme. However, it provides a much stronger level of privacy comparable to that provided by end-to-end encryption with no aggregation. It also shows that the scheme distributes the communication load quite evenly among all nodes, resulting in better network longevity. Finally, they augmented their scheme to provide end-to-end aggregate authentication. Without knowledge of a group key, an external attacker cannot tamper with any aggregate, without being detected. In conclusion, they offer efficient and provably secure

techniques for end-to-end privacy and authenticity, with reasonably good security assurances, in WSNs. The limitation of this scheme is that aggregating data is not quite equivalent to collecting individual sensor readings so in some applications, for example, perimeter control, aggregation is not applicable, since only individual sensor readings are of interest. However, many WSN scenarios that monitor an entire microenvironment (e.g., temperature or seismic activity) do not require information from individual sensors; instead, they put more emphasis on statistical quantities, such as mean, median and variance.

Wei in [Wei et al., 2009] presents a secure data aggregation for wireless sensor networks. Here he explores a homomorphic aggregation system based on a public key encryption (PKE) scheme, which can do both additive and multiplicative aggregation as well as mixed the two operations. By far, it is the first data aggregation scheme able to mix aggregation homomorphic operations based on PKE scheme. Author assumes that the networks are hierarchical for end-to-end encryption, just like a tree. Moreover, he also assumes that all nodes involved in the aggregation are credible. Author's goal is to provide a data aggregation able to prevent a passive attacker (eavesdropper) from gaining any property of the message (with a high enough probability): it is called semantical security. A system is said to be semantically secure if it is secure against eavesdroppers. In this paper, Wei describes this security service in the form of a game played between a challenger (an authorized user of the cryptosystem) and an attacker (someone who wishes to learn more about the plaintext and/or key).

Nath in [Nath et al., 2009] describes the Outsourced Aggregation model, where sensing services outsource their sensor data collection and aggregation tasks to third-party service providers, which are related to aggregators. As aggregators can be untrusted or compromised, is essential for a sensing service to be able to verify the correctness of aggregation results. The author presents SECOA, a framework with a family of novel and optimally-secure protocols for secure outsourced aggregation. This framework is based on a unified use of one-way chains. It supports a large and diverse set of aggregate functions, multiple hierarchically organized aggregators, deterministic methods to detect any malicious aggregation behavior without communication with sensors, and incurs a small and workload-independent communication load on sensors. Motivated by Microsoft's SenseWeb, the author proposes a outsourced aggregation model where users query the web-portal for various aggregates on the data submitted by sensors. The queries may be qualified with predicates on static attributes of the sensors (e.g., sensors within a given geographic region). The portal forwards user queries to the root aggregator. The root aggregator computes query answer (potentially after communicating with other aggregators in the system), and then sends the answer back to the portal. The portal verifies the correctness of the answer and returns the answer to the user if it passes verification. It is possible to have multiple independent portals, e.g., for load balancing.

Papadopoulos in [Papadopoulos et al., 2012] presents a novel secure data aggregation. This scheme is the first work that provides both integrity and confidentiality covering a wide range of aggregates and returning exact results. The author's contribution consists into provide secure

aggregation for WSN where: (i) the sensors are deployed in open and unsafe environments, and (ii) the aggregation process is outsourced to an untrustworthy service. This data aggregation is related to SIES, a scheme that solves exact SUM queries through a combination of homomorphic encryption and secret sharing. Summarizing, the author shows how to adapt SIES in order to support exact aggregate queries (such as MAX, MEDIAN, etc.). Finally, he augments its schemes with a functionality that identifies malicious sensors, preventing denial of service (DoS) attacks and attributing robustness to the system. These techniques are lightweight and induce very small bandwidth consumption; therefore, they constitute ideal solutions for resource-constrained sensors. In their studies, the author explains the security of SIES against various attacks in terms of data confidentiality, integrity, authentication, and freshness as well.

3.2 Proposed Data Aggregation: Secure Lossless Aggregation

In this section we present a protocol for M2M networks that secures communications between a set of collector nodes or meters and their application server in an efficient and secure manner. The protocol is designed for a typical scenario depicted in Figure 3.3 where some metering nodes collect data which is reported to a gateway or application server through an operator multi-hop network. As a result, four types of nodes compound the network: 1) metering nodes, that actually infer the data; 2) aggregator nodes, that collect data sensed by a set of metering nodes; 3) routers that provide the necessary infrastructure to facilitate communication between involved nodes (notice that aggregator nodes are also routers); and 4) the gateway itself.

The aim of the protocol is to provide end-to-end (between meters and gateway or beyond) and hop-by-hop (within every link) whilst minimizing the traffic in the network and thus maximizing the overall life of involved nodes.

End-to-end security is achieved by means of a shared secret between every meter and the gateway; hop-by-hop security is done at link layer by means of pairwise keys between every network node and its one-hop neighbors. Adding lossless aggregation to this extra security, allows further optimization of network resources and minimization of energy expenditures. As we will show later in Section 3.3, the proposed protocol not only avoids an extra cost for security but also reduces the overall cost of the process of sending the data. This is due to aggregation savings making up for or even exceeding the computational cost of the security operations.

Subsequently, we outline the proposal of a complete security suite in form of end-to-end and hop-by-hop security services for the lossless aggregation. Consequently, we will quantify the benefits of the proposed security suite in the context of the popular IEEE 802.15.4 standard.

3.2.1 End-to-end

The aim of end-to-end security is to protect the data from unauthorized eavesdropping (confidentiality); to allow the destination to check the integrity of the received data and its

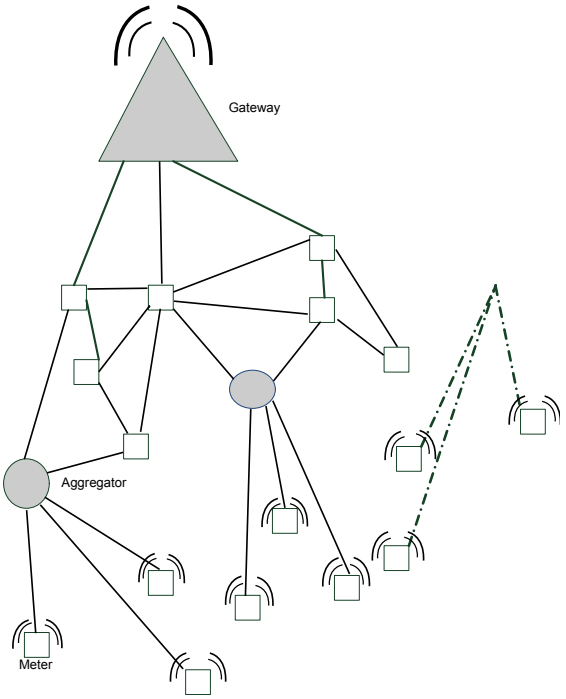

Figure 3.3: Abstraction of the M2M application scenario.

freshness; and to unequivocally identify the source of such data (authentication). End-to-end security is achieved here as follows.

The metering node creates a packet with the sensed data as shown in Figure 3.4. The headers include: the source of data (addressing field), destination (gateway), a timestamp, a key identifier, a security control and the data length; the data is encrypted with the key shared with the gateway; and a MAC is appended. Consequently, end-to-end security, which is to say confidentiality, integrity and authentication and freshness (because of the timestamp), is provided between the meter and the gateway.

Compared to non-secure protocols, the use of end-to-end security introduces some overhead (see OH_N in the implementation example in Figure 3.5); however, on the other hand, it allows the gateway to unequivocally and securely identify the source of the data and to detect any modification of this data along its path to the gateway or beyond.

In a typical implementation, the overhead OH_N related to achieving end-to-end security is at least:

- An identifier of the key/s used for encrypting the data and creating the MAC. This identifier allows the gateway to find or derivate the keys for checking the MAC and decrypting the data. Once again, 1 byte should be enough in most cases.

- A security control that contains the security level and the key identifier mode.

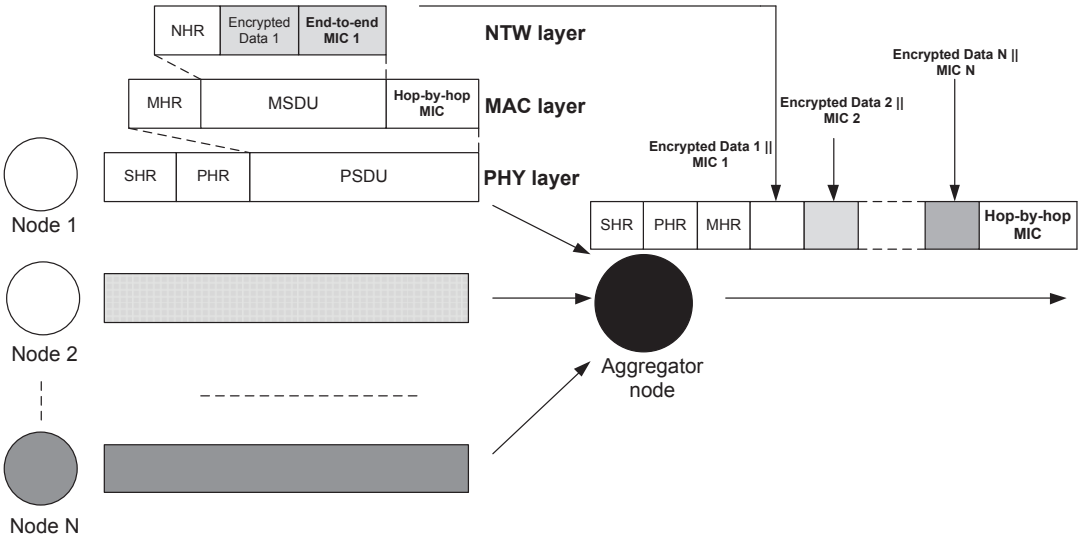

Figure 3.4: The secure lossless aggregation process.

- A timestamp in order to guarantee freshness of collected data (which will also be usually present in any non-secure scenario). Its length will be related to the amount of sent/received packets per time interval. Usually the timestamp is just related to the frame counter, the key counter or both.
- The length of the encrypted data. The gateway will need this length in order to know how many bits to decrypt after this header. Typically the length is part of the frame control field.
- A MAC of the packet header and data. The MAC typically is 32, 64 or 128 bits long.

Emphasis of end-to-end security is on ensuring confidentiality whilst being able to confirm source authenticity and data integrity.

3.2.2 Hop-by-hop

End-to-end security is checked at the final destination; however, before reaching the gateway or Internet destination, the packets must go through one or more wireless links that are by nature exposed to attackers. As a result, if no security is provided in order to restrict the access to the media, only the destination point will be able to detect altered, dropped or fake packets. This fact exposes the network to exhaustion attacks since those packets will waste precious energy at the intermediate nodes (routers). Consequently, hop-by-hop integrity, authentication and freshness should also be provided at link layer.

From above reasoning, the protocol requires the use of timestamps and MACs also at link layer. Compared to non-secure protocols, the use of hop-by-hop security introduces at least the

following overhead:

- A timestamp (it is not related to the timestamp at network layer) in order to guarantee freshness. Once again, the timestamp is often a frame counter, a key counter or a combination of both.
- An identifier of the key used for creating the MAC. This identifier allows the next hop to find or derive the keys for checking the MAC. Once again, 1 byte is enough in most cases.
- A MAC of the frame header and payload. The MAC typically is 32, 64 or 128 bits long. Strictly speaking the frame integrity check sequence (commonly referred as frame check sequence - FCS) can be replaced by this MAC (for example when using TinyOS [Karlof et al., 2004b]), and thus the real overhead would be just the difference (if there is any) in size between the MAC and the check sentence.

Emphasis of hop-per-hop security is hence on source authenticity and data integrity.

3.2.3 Lossless Aggregation

Since the collected/sensed data normally contains just a few bits of metering information, the payload of packets between meters and their aggregator node is usually far from its maximum allowed or its optimal size. As a result, within the security framework described above, we propose to concatenate several legitimate packets into a single one at the aggregator node. This concatenation or lossless aggregation reduces unnecessary overhead (headers and MACs), and thus aids in compensating for the overhead induced by hop-by-hop and end-to-end security. The proposed aggregation process is illustrated in Figure 3.4 and its execution is detailed in Algorithm 1.

From the aggregator node to the gateway, the intermediate nodes have only to check the MAC integrity/authentication of every received packet, and forward the packet with a new MAC and updated headers. Integrity, authentication and freshness at link layers are therefore checked at every hop. The resulting packets at the aggregator will be made of the following fields:

- header: that also includes the key identifier used for hop-by-hop security and timestamp.
- for $i = 1$ until $i = n$ with n the number of aggregated input packets at every output packet.
  - Network header of the ith meter's packet.
  - Encrypted data of the ith meter's packet.
  - MAC of the ith meter's packet.
- Link layer MAC.

Algorithm 1 Secure lossless aggregation (at every aggregator node).

```
osize = 0
opacket_id = 0
createOutputPacket( opacket_id )
for every input packet do
  if checkMIC() == TRUE then
    mac_data = getPacketMacData()
    if osize + sizeOf( mac_data ) > P'_a then
      createMIC( opacket_id )
      sendPacket( opacket_id )
      opacket_id = opacket_id + 1;
      createOutputPacket( opacket_id )
      osize = 0
    end if
    aggregateInputPacketPayloadIntoOutputPacket( mac_data, opacket_id )
    osize = osize + sizeOf( mac_data )
    if last received packet OR timeout then
      createMIC( opacket_id )
      sendPacket( opacket_id )
    end if
  end if
end for
```

Summarizing, the aggregator receives the packets, checks their link layer MACs, concatenates the payloads of as many link layer received packets as it can into one link layer payload of every output packet, calculates the link layer MAC of the output packet and sends the resulting packet to the next hop. However, to evaluate the effectiveness of this suite, we first need to understand the impact of the wireless communication channel, which is dealt in Annex 1.

We use AES-CCM with 128 bit shared key between the source (meter) and the base station for end-to-end security and for this reason the minimum payload length permitted is a multiple of the key length (128 bit = 16 bytes). If the payload is shorter than 16 bytes, the protocol will fill it with useless bytes only to guarantee a smooth encryption. Our protocol, obeying IEEE 802.15.4 using AES-CCM, permits at most to concatenate 3 nodes' packets with 16 bytes of data payload or 2 nodes' packets with 32 bytes.

3.2.4 Comparative Table Among Aggregation Techniques

In this sub-section, we compare the proposed lossless protocol with the state-of-art. Summarizing, our protocol offers a high level of security with end-to-end and hop-by-hop security services and in addition it is perfect for those applications, such as metering services, where data aggregation process has to save energy without modify any collecting individual sensor readings.

On the other hand, Castelluccia [Castelluccia et al., 2009] and Papadopoulos [Papadopoulos et al., 2012] for example, present interesting solution: they work with secure homomorphic data aggregation schemes that use mathematical function to collect and aggregate individual sensor readings. However with these solutions, any specific individual input sensed data cannot be recognized from the final result of the aggregation process. This is mainly why concatenation and homomorphic are not alternative for the same application but rather they should be the selected data aggregation solutions depend on the service requirements. All prior secure aggregation schemes, which were aforementioned, have some problems of different nature: security, energy consumption, etc. Our solution is a good trade-off because it is able to save energy with a simple mechanism of concatenation and also it is not jeopardized by traditional attacks.

Table 3.1: Comparing of previous secure aggregation schemes with the proposed one.

Scheme	*Conf*	*Integrity*	*Freshness*	*Auth*
CDA	X			
SDA		X	X	X
SIA	X	X	X	X
SHDA		X	X	X
WDA		X		X
SecureDAV	X	X		X
SRDA	X		X	
SDAP	X	X	X	X
ESA	X	X	X	X
EDA	X			
Wu(2007)	X		X	
SELDA	X		X	
Ozdemir (2007)	X		X	
Zhang (2008)		X		
Wang (2009)	X			
Castelluccia (2009)	X	X		X
Wei (2009)	X			
SECOA	X	X		X
SIES	X	X	X	X
Our Protocol	X	X	X	X

3.3 Protocol Analysis & Optimization

In the following we present a thorough evaluation of our introduced protocol suite. The aim is to show its goodness in terms of energy consumption apart from its previously stated security characteristics. To be more realistic, we have applied it to IEEE 802.15.4, which is the most

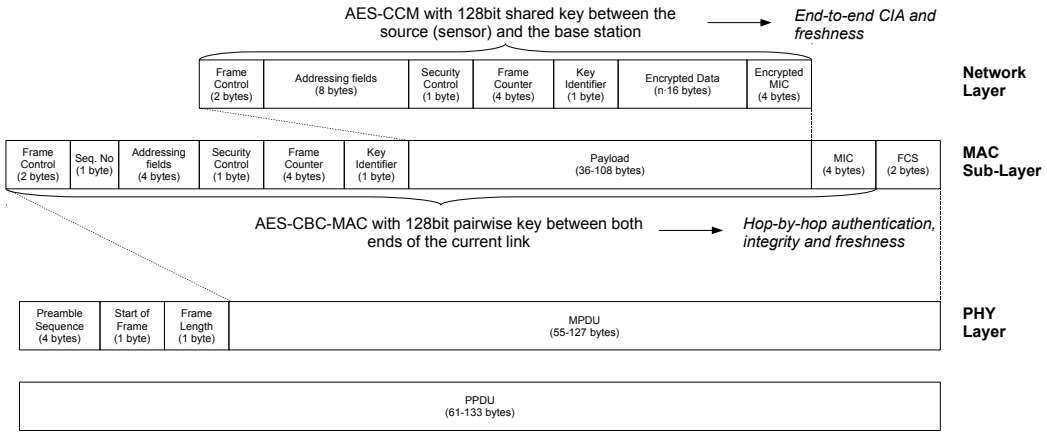

Figure 3.5: The proposed aggregation packet format.

extended wireless communications technology for remote metering to-date. It is understood, however, that the analysis is equally applicable to emerging IEEE 802.15.4g as well as a large set of proprietary networks. Regarding the secure aggregation protocol, we thus first describe the actual frame structure in more details, after which we quantify the performance and energy benefits of the aggregated solution assuming a lossless channel first and extending it then to lossy channels.

3.3.1 Frame Structure

We define subsequently a complete frame structure that allows both hop-by-hop and end-to-end security based on the IEEE 802.15.4 frame format [Network layer, 2007].

At the physical layer, we have assumed that we are operating, for the sake of simplicity, at the 868-868.6 MHz or 902-928 MHz frequency bands with binary phase-shift keying modulation. These options lead to a physical preamble of 4 bytes and a start of frame delimiter (SFD) of 1 byte [IEEE Computer Society, 2010]. Then, the standard physical header is applied (1 byte more). Consequently, and assuming BPSK, the resulting frame structure in bytes is shown in Figure 3.5.

At link layer, we use short addressing (2 bytes per address) since we can assume that a given node will not have more than 2^{16} one-hop neighbors. Further, we apply the standard IEEE 802.15.4 security headers in order to generate a MAC with Advanced Encryption Standard (AES) in Cipher Block Chaining Message Authentication Code (CBC-MAC) mode with a 128 bit pairwise key between both ends of the current link. Since the MAC is applied to the link layer header as well as the link layer payload, it guarantees: 1) *authentication*, since the receiving node can certify the transmitting node link layer address; 2) *integrity*, since the link layer frame contents cannot be modified without being detected (with a high probability); and 3) *freshness*, since the receiving node can verify sequence number and frame counter fields in order to discard

old or replayed packets.

Above link layer, which we will call for simplicity network layer, we have assumed standard addressing (4 bytes per address) thus supporting standard network protocols such as the Internet Protocol (IP). Moreover, we define the use of AES in Counter with CBC-MAC (CCM) mode with 128 bit shared key between the source (meter) and the gateway for end-to-end security. AES-CCM provides: 1) *confidentiality*, since the metered data is encrypted; 2) *integrity*, since the entire network layer packet can be verified with the provided MAC; *authentication*, since it allows verifying the original source of data; and *freshness* since a frame counter or timestamp is also provided at this level. Because of the use of AES-CCM, the payload length is a multiple of the key length, i.e. 16 bytes; therefore, padding is applied whenever necessary.

3.3.2 Energy Cost

In order to conduct a realistic analysis, we use the Berkeley/Crossbow motes platform on the Mica2dots [Wander et al., 2005] which is a popular platform for WSN research. The major energy consumers on these sensor devices are the Atmel ATmega128L 8-bit microcontroller and the Chipcon CC1000 low-power wireless transceiver. The Atmega128L runs at a clock frequency of 4MHz.

Our analysis is based on [Wander et al., 2005], which has approximated the energy consumption for individual cryptographic algorithms and other activities such as data transmission by measuring the current drawn from the power supply. A more accurate approach using an oscilloscope and a sense resistor, conducted from the same authors, showed the error to be less than 5%. Table 3.2 presents the characteristic data for the Mica2dot platform. It is interesting to note that the power required to transmit 1 bit is equivalent to roughly 2090 clock cycles of execution on the microcontroller alone.

The cost of receiving one byte (28.6uJ) is roughly half of that required to transmit a byte (59.2uJ). During transmission and reception, the microcontroller is powered by the wireless transceiver. We used different packet sizes, but for example a packet of 61 bytes costs 61*28.6uJ=1.7446mJ to receive and 61*59.2uJ=3.6112mJ to transmit. In addition, we chose to focus on AES with 128-bit keys for data encryption/decryption [Wander et al., 2005] and SHA-1 for hashing; see, Table 3.3.

We thus observe that transmission and reception energy costs are within the same order of magnitude. Furthermore, the energy consumption of security operations is by an order of magnitude lower than the communication costs. We will therefore subsequently neglect the security energy cost and assume that transmission and reception cost the same amount of energy. The spent energy is thus directly proportional to the number of bits sent, which will be our metric of choice when quantifying gains.

Table 3.2: Cost of transmission and reception.

Fields	Value
Effective data rate (ρ)	12.4kbps
Energy to transmit (ε_{tx})	59.2uJ/byte
Energy to receive (ε_{rx})	28.6uJ/byte
ATmega 128L active mode	13.8mW
ATmega 128L power down mode	0.0075mW
ATmega 128L MIPS/Watt	289 MIPS/W
$\mathscr{C}_A^K$ in ATmega 128L	K * 0.07 uJ

Table 3.3: Cost of security operations.

Algorithm	Value
AES-128 Enc Φ_{enc}	1.62uJ/byte
AES-128 Dec Φ_{dec}	2.49uJ/byte
SHA-1 η	5.9uJ/byte

3.3.3 Per-Aggregator Byte Savings Over Lossless Channels

As depicted in Figure 3.5, the minimum payload at link layer is 36 bytes and the maximum is 108 bytes. That is to say, a maximum of 3 packets can be aggregated at link layer into one packet ($3 \cdot 36$ bytes $= 108$ bytes).

We next assume a varying number of meters N attached to a given aggregator and two possible lengths of collected data, 16 and 32 bytes (multiples of 128 bits as previously stated). As a result, since the total overhead (see Figure 3.5) is $OH_N + OH_M + OH_P = 20 + 19 + 6 = 45$ bytes, the size of the packets generated by the meters P_m is either 61 or 77 bytes. Considering that the aggregator concatenates network packets and that the maximum physical layer packet size is P_a, then the maximum number of aggregated packets at the output frame is $A = \left\lceil \frac{P_a - OH_P - OH_M}{P_m - OH_P - OH_N} \right\rceil$ and the total amount of packets at the aggregator output is $O = \lceil N/A \rceil$.

From the above reasoning, assuming an error-free lossless channel, the total amount of bytes at the output of the aggregator with and without aggregation as well as the Δ in bytes are respectively obtained as per below expressions.

$$
\begin{aligned}
B_{agg} &= N \cdot (P_m - OH_P - OH_M) + O \cdot (OH_P + OH_M) \\
B_{\cancel{agg}} &= N \cdot P_m \\
\Delta &= B_{\cancel{agg}} - B_{agg} = (N - O) \cdot (OH_P + OH_M)
\end{aligned}
$$

Figure 3.6 presents the percentage of saving $\frac{\Delta}{B_{\cancel{agg}}} \cdot 100\%$ in bytes transmitted at the aggregator node when using lossless aggregation on a perfect and lossless communication channel. Figure 3.6 clearly shows how the aggregation efficiency grows when the length of the collected

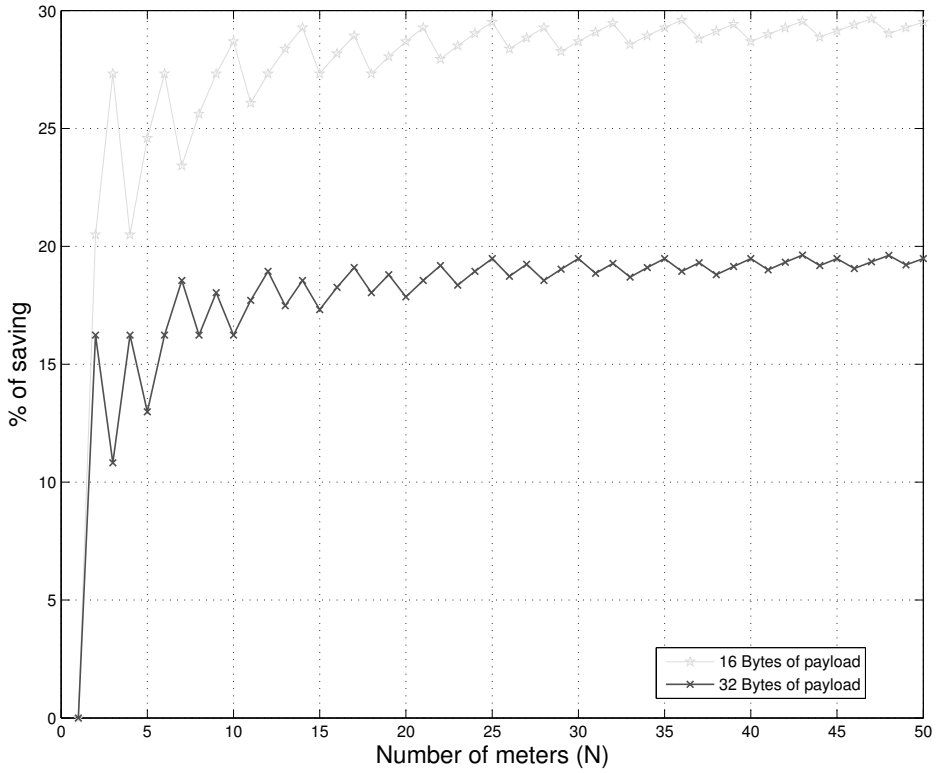

Figure 3.6: Percentage of transmitted bytes and thus energy saved when using lossless aggregation.

data decreases. Since typically collected data in Smart Grid metering applications are just a few bits long, we can save up to a 27% of the bits transmitted at the output of the aggregator, a gain which is further pronounced if multiple hops are present. This gain in overhead translates directly in energy gains since the energy needed to accomplish proposed security is by orders of magnitude lower than the communication energy (Table 3.2 and Table 3.3).

3.3.4 Performance Optimization Over Lossy Channel

We now utilize the analytical body derived in Annex 1 and apply it to the performance analysis and optimization of the secure aggregation protocol. We will subsequently only consider the case of fast fading with shadowing and block channel coder; the analysis and insights for the other cases are very similar and thus omitted here. With the mathematical body and set of protocols at hand, various issues pertaining to energy efficiency can be looked at. We will subsequently concentrate on two issues, notably the optimal hop-per-hop packet length and the best end-to-end

Table 3.4: Comparison of bytes transmitted by the aggregator node with and without aggregation.

N	collected data (bytes)	P_m (bytes)	$bytes_{na}$	$bytes_a$	Δ (bytes)	Δ (%)
2	16	61	122	97 ($O=1$)	25	20.49%
3	16	61	183	133 ($O=1$)	50	27.32%
19	16	61	1159	859 ($O=7$)	300	25.88%
31	16	61	1891	1391 ($O=11$)	500	26.44%
53	16	61	3233	2358 ($O=18$)	875	27.06%
97	16	61	5917	4353 ($O=33$)	1564	26.43%
2	32	77	154	129 ($O=1$)	25	16.23%
3	32	77	231	206 ($O=2$)	25	10.82%
19	32	77	1463	1238 ($O=10$)	225	15.37%
31	32	77	2387	2012 ($O=16$)	375	15.71%
53	32	77	4081	3431 ($O=27$)	650	15.92%
97	32	77	7469	6269 ($O=49$)	1200	16.06%

aggregation strategy.

As for the optimal hop-per-hop packet length, we observe that having longer packets allows us to send more data at the caveat that packets are more likely corrupted due to noise, fading and shadowing. There is hence a clear tradeoff in terms of energy efficiency, which is defined as the number of useful data bits N sent over the average energy spent in sending this amount of data, i.e.:

$$\eta = \frac{\varepsilon' N}{\varepsilon'(N+OH)\overline{N}_{tx}(N+OH)}, \tag{3.1}$$

where $\varepsilon' = \varepsilon/8$ is the energy spent per bit and OH is the overhead discussed before. Using (A.12) and differentiating w.r.t. the packet length N, we obtain the optimal packet length which allows us to send a given amount of information with least energy:

$$N_{\text{opt}} = -\frac{OH}{2} + \sqrt{\left(\frac{OH}{2}\right)^2 - \frac{OH}{\ln F}}, \tag{3.2}$$

where $F = \sqrt[k]{1 - c \cdot \left(\alpha \left(\frac{m}{m+\beta\overline{\gamma}}\right)^m\right)^{t+1}}$. Since in embedded systems the channel quality $\overline{\gamma}$ can typically be known at the transmitter, each transmitting node could modulate its packet length to improve energy efficiency. If each node along the routing path performs this operation, energy expenditure will be minimal. For instance, assuming a shadowing of $\sigma_{\text{dB}} = 4\text{dB}$, a dynamic packet length yields about 15% energy gains over a static packet length of 127 bytes.

The motivation for designing optimal end-to-end aggregation strategies is to find the aggregation threshold beyond which an aggregated packet requires more energy to deliver

the information than the separate non-aggregated packets. It is thus a more holistic approach w.r.t. above per-hop strategy and also avoids bottlenecks which may arise with above strategy if one particular link is very poor. To conduct the analysis, we assume a multi-hop network of H hops where the first hop is formed by K nodes communicating with one parent node, which in turn communicates with its parent nodes, etc, until the gateway is reached after H. To not unnecessarily cluttering the analysis, we assume here a fixed packet length N. The energy difference between the aggregated and non-aggregated cases appears after the aggregation node; we will thus not consider the energy spent in the first hop. In the non-aggregated case, the energy spent to send the K packets of length N with overhead OH from the first parent node to the sink is:

$$\begin{aligned} E_{\not{agg}} &= \varepsilon' \sum_{j=2}^{H} K \cdot (OH+N) \cdot \overline{N}_{tx}(OH+N) \\ &= \varepsilon' \sum_{j=2}^{H} K \cdot (OH+N) \cdot G(\overline{\gamma}_j)^{OH+N}, \end{aligned} \tag{3.3}$$

where $G(\overline{\gamma}_j) = \left(1 - c \cdot \left(\alpha \left(\frac{m}{m+\beta\overline{\gamma}_j}\right)^m\right)^{t+1}\right)^{-1/k}$. In the aggregated case, the energy can be calculated as:

$$E_{agg} = \varepsilon' \sum_{j=2}^{H} (OH + K \cdot N) \cdot G(\overline{\gamma}_j)^{OH+K\cdot N}. \tag{3.4}$$

The aggregation threshold occurs when aggregating data from K_{th} nodes into a single packet becomes energy inefficient, i.e. $E_{agg} \leq E_{\not{agg}}$, simply because the resulting packet is too long and thus requires too many retransmissions. To obtain the aggregation threshold K_{th} requires above highly nonlinear equations to be equated and solved w.r.t. K, requiring numerical tools. However, assuming that $K_{th} \cdot N$ is sufficiently large which allows us to invoke some limiting properties of generalized means with exponent $N_{th} \cdot K$, yields an approximate closed-form solution as:

$$K_{th} \approx \frac{\log \sum_{j=2}^{H} (1 + OH/N) \cdot G(\overline{\gamma}_j)^{OH+N} - \log(H-1)}{N \cdot \log \max_j \left\{G(\overline{\gamma}_j)\right\}} \tag{3.5}$$

Various performance dependencies can be deduced from above analysis, where e.g. Figure 3.7 illustrates the energy gains of the aggregated solution until the breakpoint beyond which aggregation is detrimental.

Above analysis hence facilitated the understanding, quantification and optimization of secure aggregation protocols.

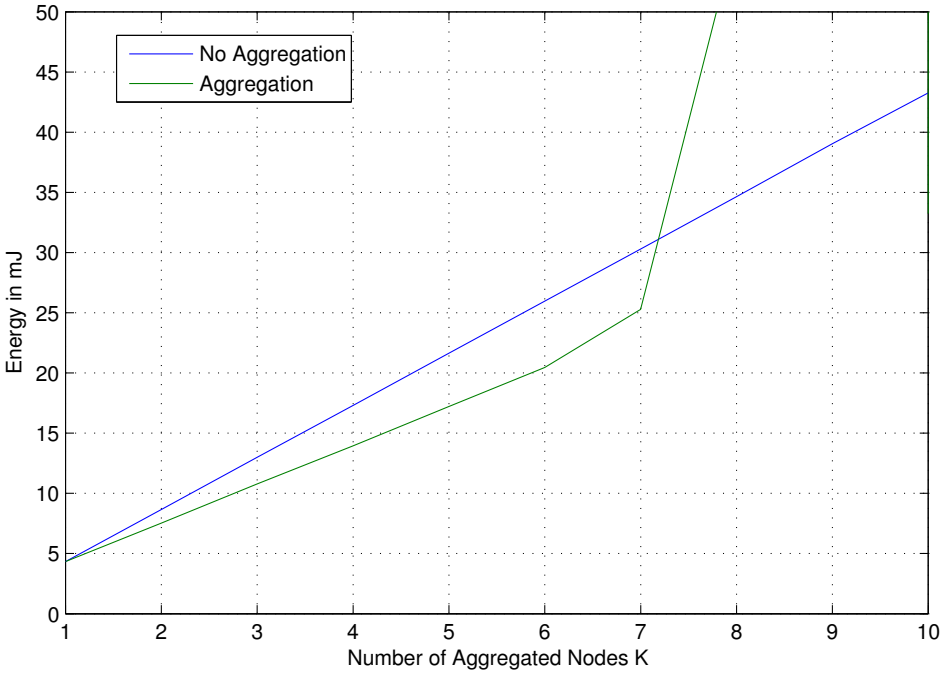

Figure 3.7: Energy of aggregated versus non-aggregated scheme over a 5-hop network and prior discussed operating conditions.

Chapter 4

PHY-layer Authentication Preamble

"A day without a smile is a lost day"
Charlie Chaplin.

Contents

Machine-to-machine networks are spreading over every sector of our society due to their self-organization capabilities. In these networks, thousands of devices are left unattended for years of operation without the possibility of human intervention. In this sense, every step forward into avoiding early exhaustion of the network nodes is of paramount importance. In our studies, we have introduced a novel authentication scheme that is able to discard non-intended and/or non-legitimate packets just after the reception of the physical preamble. This proposal was shown to yield enormous energy saving with regard to both node exhaustion attacks and normal network operation. In this chapter we present all our contributions regarding this issue.

4.1 Introduction to Availability Issue in M2M Networks

Machine-to-Machine technology is a novel communication paradigm that is rapidly gaining in importance in the vision of telecommunication service providers. M2M devices are typically required to be small, inexpensive, able to operate in a self-organized mode and to communicate through the wireless medium. An important constituent of this technology is its capacity to interconnect machines, which are often referred to as devices, to exchange data using energy-efficiency communication protocols.

Unfortunately, several security challenges must be addressed in order to take advantage of all the benefits offered by the M2M technology. Consider for example an unattended M2M network in charge of monitoring a given utility, such as energy consumption. If the M2M nodes can be easily forged or compromised, the supply service is no longer reliable for both the customer and the provider. Moreover, reliability of the service relies on the resilience of the nodes and the communication links to Denial-of-Service attacks, that is to say, in the service availability. The specific nature of M2M networks, that are often made up of hundreds to thousands of devices which are left unattended for years of operation without human intervention, makes securing availability in M2M networks especially challenging.

Whilst security understood as protecting the communication data has been widely studied [Hwang et al., 2011, Zahariadis et al., 2010, Lv et al., 2012, Labraoui et al., 2011, Sodagari et al., 2012,Ishmanov et al., 2011], guaranteeing the service availability in M2M communication systems still remains as an open field with a lot of work to be done.

Generally speaking, availability relies on the proper operation of the network nodes and their communication links. Therefore, said availability is mainly jeopardized by DoS attacks, which can be classified into: i) attacks damaging network nodes; ii) attacks disturbing the communication links, e.g. by means of jamming techniques; and iii) attacks exhausting the network nodes, e.g. by engaging them into meaningless packet exchanges that consume their precious batteries and thus significantly shorten their lifetime.

Protection of the network nodes is mainly related with physical security that should allow keeping nodes out of reach of attackers. Regarding communications, several security proposals have been proposed in order to prevent or mitigate jamming-based attacks, most of them based on frequency-hopping and channel-surfing techniques [Wang and Wyglinski, 2011]. However, to the best of our knowledge, there are still no energy-efficient solutions aiming at mitigating the effects of exhaustion attacks [Bartoli et al., 2011].

Exhaustion attacks often exploits the lack of authentication at link or network layer in two ways: 1) injecting fake packets in the network, which is related to network exhaustion attack; and 2) forcing link layer dialogs that makes a given node continuously transmit and receive messages, which is related to node exhaustion attack. In the former case, the malicious packet is often detected at the application layer and thus precious network resources are wasted relaying the packet to its destination. In the latter case, an attacker can exhaust the node's resources by repeatedly sending useless fake messages that are completely received before being discarded.

Focusing on node exhaustion attacks, one can argue that even the normal operation of a group of nodes can be in some manner considered as a node exhaustion attack. In fact, besides all the packets intended to a given node, due to the broadcast nature of wireless media, that node receives all packets sent by its in-range neighbors. Typically, energy is spent in receiving the entire packets, performing the security checks, checking on intended destinations; only then the packets are discarded. Since non-intended legitimate packets arrive at a fairly regular frequency which depends on the neighborhood cardinality and the neighboring nodes' transmission rates, the energy spent on these to-be-rejected packets is not negligible.

In order to provide a solution for the presented issues (protection against node exhaustion attacks and avoiding of unnecessary waste of resources due to complete reception of non-intended packets), in [Bartoli et al., 2011] we have proposed an authentication/verification method at the PHY-layer that is able to challenge exhaustion problems and attacks in M2M networks. The presented method is able to reject non-intended non-malicious and/or malicious packets after the reception of just an authentication preamble (AP) at the physical layer, saving energy and therefore increasing the system lifetime, that is to say guarantying its long-term availability. However, as pointed out in [Bartoli et al., 2011], the use of the proposed AP could lead to out-of-synchronization states at the physical layer, and thus it may deadlock the communication link. Consequently, in this chapter we present the following contributions: i) the proposed AP solution comparing it with the sate-of-art, ii) a recovery protocol to the out-of-sync state, iii) an in-depth study of the optimal parameters of our proposal, and iv) an evaluation of the proposal in terms of neighbors nodes density thresholds and energy consumptions under an example scenario based on the IEEE 802.15.4e amendment to the IEEE 802.15.4-2006 standard.

Taking into account the IEEE 802.15.4e amendment, it inherits the physical layer defined in the IEEE 802.15.4-2006 standard while largely modifies the link layer. This amendment is intended to add functionalities to the traditional 15.4 link layer to (a) better support the industrial markets and (b) permit compatibility with modifications being proposed within the Chinese

WPAN. In particular, it aims to specifically advocate time-frequency scheduling approaches between transmitter and receiver with the twofold benefit of significantly reducing outage and also facilitating routing. For such a reason, several telecommunication companies, such as ORANGE/France Telecom and Siemens [Chen et al., 2010], have decided to further work in this specification, and this is also the reason why we have selected it for our implementation case.

However, the explicit contribution of this study is to provide a security tool able to guarantee availability and efficiency. To this end, we evaluate the optimal parameters, which depend on the link's BER, to compute the minimum energy consumption of the system.

4.1.1 DoS Exhaustion Threats

Malicious attacks targeting network availability in M2M networks can be considered as DoS attacks, which attempt to delay, exhaust, block or corrupt information transmission in order to make network resources unavailable to nodes that need information exchange.

DoS attacks against TCP/IP have been well studied in the literature regarding attacking types, prevention and response [Schuba et al., 1997] [Mirkovic and Reiher, 2004], therefore, in the following, we will discuss potential attacks that specifically target power network availability by exhausting the network and the devices' limited resources.

These typical attacks in low-power embedded networks aim to drain the devices' battery and can be launched against the network or against a specific node:

- **Network exhaustion attack**.

  The attacker can fake a message asking the nodes to continuously retransmit messages to exhaust its energy. As M2M network is composed of multi-hop nodes, if only the destination node is able to check the authenticity of a message, the intermediate nodes could waste energy to send and receive fake packets.

- **Node exhaustion attack**.

  Every fake message recited from a valid node involves an energy consumption, thus an attacker can exhaust the receiver energy by only sending invalid messages.

A specific example of node exhaustion exploits the two way request-to-send/clear-to-send (RTS/CTS) handshake that many link layer protocols use to mitigate the hidden node problem. An attacker can exhaust a node's resources by repeatedly sending RTS messages to elicit CTS responses from a targeted neighbor node; strong link-layer authentication can mitigate these attacks however a targeted node receiving the bogus RTS messages still consumes energy and network bandwidth.

To counter this, access security mechanisms are typically deployed at link layer. Energy is thus spent in receiving the entire packet and performing the security checks on the entire packet;

only then the packet is discarded. Our aim is to improve these techniques to better meet M2M networks requirements.

4.1.2 Exhaustion Attacks State-of-Art

Exploring the State of Art regarding exhaustion attacks the following results are emerged: packets being received can either be intended or not intended for the specific receiver. The non-intended packets can be of non-malicious but as well as of malicious nature. Non-intended packets are typically received fully, just to be rejected at higher layers due to non-matching link-layer address, IP address or security primitives.

The energy spent on these to-be-rejected packets is not negligible. Non-intended non-malicious packets arrive at a fairly regular frequency which depends on the neighborhood cardinality and the neighboring nodes' transmission rates. Non-intended malicious packets arrive rarely but consistently in the case of a denial of service attack.

Non-intended non-malicious packets typically originate from transmitting nodes in the one-hop neighborhood of the receiving node. These can be data packets as well as control packets. Typically, energy is spent in receiving the entire packet, performing the security checks on the entire packet, checking on intended destination; only then the packet is discarded. No specific security solutions are known to provide energy savings against these packets.

Non-intended malicious packets yield DoS attacks with the aim to jeopardize device and/or link availability. Typical attacks in low-power embedded networks are exhaustion attacks with the aim to drain a device's battery. To counter this, access security mechanisms are typically deployed. Energy is thus spent in receiving the entire packet and performing the security checks on the entire packet; only then the packet is discarded.

Commercial and industrial standards [Lopez et al., 2009] for embedded networks, such as Zigbee, WirelessHART and ISA 100.11a, are based on the physical and link layers of the IEEE 802.15.4 standard, and they provide a simple solution against exhaustion attacks: authentication/integrity mechanism at link-layer. This mechanism permits to identify an invalid message only after the reception of the whole packet by verifying the Message Authentication Code presented in the last bytes of the packet. With these authentication mechanisms at link layer, the intention of such attacks is not solved because the exhaustion of the victim's limited energy resources is still possible due to the high reception costs.

In [Baig, 2011], an alternative technique is presented, which is based on defining specific network topology-based patterns to model normal network traffic flow, and to facilitate differentiation between legitimate traffic packets and anomalous attack traffic packets. In this paper, the performance of the proposed attack detection scheme is evaluated in terms of the size of the sensor resource set required for participating in the detection process for achieving a desired level of attack detection accuracy. The results signify the need for distributed pattern recognition for detecting distributed node exhaustion attacks in a timely and accurate manner. This solution seems to be interesting but numerous drawbacks are identifying comparing it with

our physical authentication: our solution does not depend on patterns or in detecting the attack by a huge participation of all the nodes and also it is simple and very fast just because it should not receive the whole packet.

In [Wang et al., 2006], a possible solution to exhaustion is to apply rate limits to the link layer admission control such that the network ignores excessive requests, thus preventing the energy drain caused by repeated transmissions [Akyildiz et al., 2002]. A second technique presented in the same paper consists of using time-division multiplexing where each node is allotted a time slot in which it can transmit [Akyildiz et al., 2002]. This eliminates the need of arbitration for each frame and can solve the indefinite postponement problem in a back-off algorithm. All these solutions do not decrease the reception costs; in others words, they mitigate the problem but do not solve it integrally.

Following, we propose a new and innovative mechanism to protect networks and the nodes that compose them, from node and network exhaustion threats, which does not involve a large additional cost in terms of energy. The goal of our solution is to ensure that the energy necessary to provide the defense against exhaustion attacks is very low compared with the solutions discussed above.

4.2 Proposed Solution to Guarantee Availability: PHY-layer Authentication Preamble

Our proposed PHY-layer authentication solution relies on the use of an authentication preamble of 32 bits that allows to discard a non-intended or a malicious packet just after the AP's reception. The earlier the *AP* is placed in the packet, the more energy can be saved due to early-discards. With such a purpose, the *AP* must be placed in the packet's PHY header, preferably right after the synchronization preamble [Bartoli et al., 2011]. In the following, we detail how the AP is generated and how it is used for lightweight authentication of the received packets.

4.2.1 AP During Normal Operation

Figure 4.1 depicts the process of generating a valid AP for packets transmitted from node A to B during normal operation, being such nodes one-hop neighbors. With this purpose, A and B use their respective identification tags or IDs and the pairwise keys, shared between them as input data, to compute a hash-based message authentication code (HMAC). The first 32-bits of the output is the initial AP field to be used in the first packet transmitted from A to B. This AP, included at the physical-layer of the packet, provides a lightweight authentication or verification, confirming the appropriate origin and destination of the packet.

In order to avoid replay attacks, once an AP value is transmitted, it cannot be reused. Recursively computing the HMAC of the previous AP with the shared pairwise key as in Figure 4.1 guarantees AP's freshness over the time.

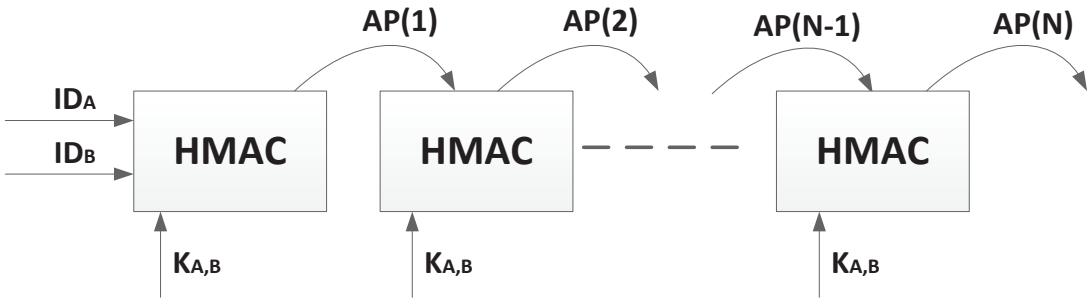

Figure 4.1: The Authentication Preamble method used during normal operation.

The AP verification process is done at the destination and consists in a deterministic method where the receiver checks if the received AP is a potential AP from any of its neighbors. Obviously, aiming at minimizing additional delays due to this authentication, all the potential future APs should be previously computed and stored in a list. If the received AP is in the list, the whole packet is received and processed, otherwise it is rejected just after the AP reception.

4.2.2 Frame Format Proposition

In this sub-section, we first bravely present the main specifications of the IEEE 802.15.4e version of the 15.4 standard and then we settle the proposed *AP* field in this amendment. In general, the physical layer defined in the 15.4e is the same of the traditional IEEE 802.15.4-2006 standard while the link layer layer is slightly modified. This amendment, thus intent to add functionalities to the traditional 15.4 link-layer to (a) better support the industrial markets and (b) permit compatibility with modifications being proposed within the Chinese WPAN. In particular, it aims to specifically advocate time-frequency scheduling approaches between transmitter and receiver with the twofold benefit of significantly reducing outage and also facilitating routing.

Focus on the proposed *AP* solution, as this mechanism is the short result of a hash function (32-bits or even 20-bits are enough to reach our intent), it can be fixed in several standard specifications without modifying their functionalities. As shown figure 4.2, we define it after the synchronization preamble symbols because it is the first modifiable place where we can adequate it without restrict the flexibility of the network; with the synchronization preamble filed defined before the *AP* method, also non-security frames can be used. In addition, as the *AP* mechanism is designed to save as much energy as possible with a fast "on-air" authentication verification test, the numbers of bits that are not received, when this test failed, are directly related to the energy savings. In a few words, the proposed *AP* method should be received as soon as possible to completely take advantage of its characteristics and thus to save as much energy as possible. However, rather studies are necessary to fix the proposed method also for the 15.4k and the 15.4g standard versions. Our adaptation also consider security suite at link and physical layer.

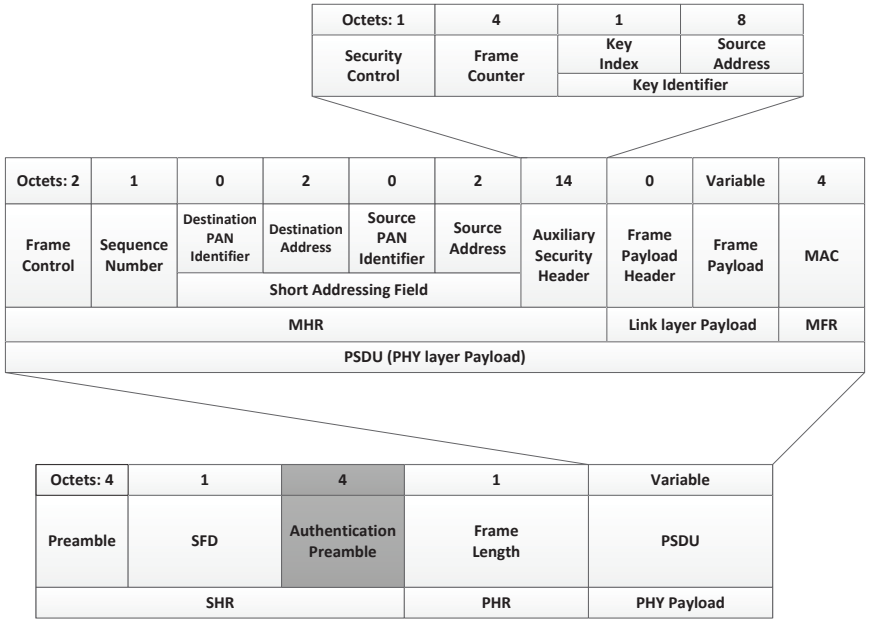

Figure 4.2: 802.15.4e Standard packet format adapted to include the AP mechanism at physical-layer.

4.2.3 Authentication Preamble Window

However, maintaining a window of W future APs for each node may correct for packet losses and potential out-of-sync states. Notice that without a window of potential future APs, if a packet is lost, the receiver will remain expecting the lost AP value and thus rejecting all future packets sent by this transmitter; that is to say that both ends get in an out-of-sync state.

With an AP window of W, besides potential packet losses, there are W valid APs form a given emitter. Once a valid AP packet from a that emitter is received, the receiver updates the window to keep W potential future AP values for this emitter. As a result, synchronization and thus communication between two nodes would be lost only if W or more consecutive packets are lost. That is to say that, assuming independent packet losses, the probability of "out-of-sync" between pairs of nodes decreases from p, where p is the probability of losing a packet, to p^W.

The optimum AP window size, which is able to minimize the probability of reaching an out-of-sync states while minimizing the energy consumption is determined in Section 4.4.

4.2.4 APs Window Protocol Uses

Nevertheless, for the sake of clarity, in this sub-section we describe the *AP* communication protocol uses, considering the implementation of *AP* window for M2M applications:

1. Every node pre-calculates the values of the Authentication Preamble window for each

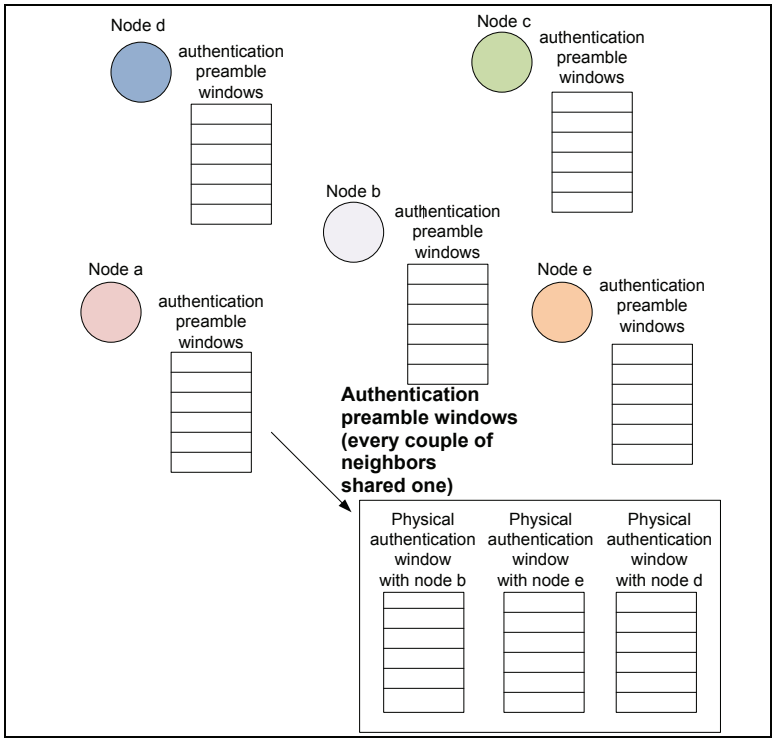

Figure 4.3: The initial scenario with all the Authentication Preamble Windows.

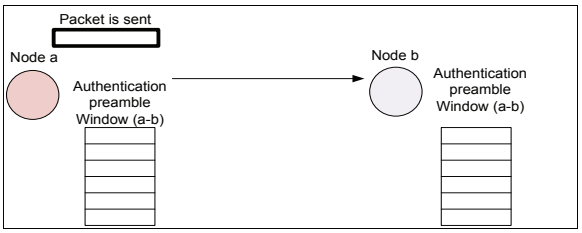

Figure 4.4: Node 'A' sends a packet to node 'B'.

neighbor. These specific values are the output results of a special hash function that has the ID addresses and secret keys as input (Figure 4.3).

2. Node 'A' builds a packet and sends it with the first authentication preamble value contented in the authentication preamble window, for the relative emitter-destination nodes (Figure 4.4).

3. The sender updates the authentication preamble window (Figure 4.5).

4. Once node 'B' has identified the synchronization symbols, it starts receiving the authentication preamble field. The authentication preamble field should be placed immediately

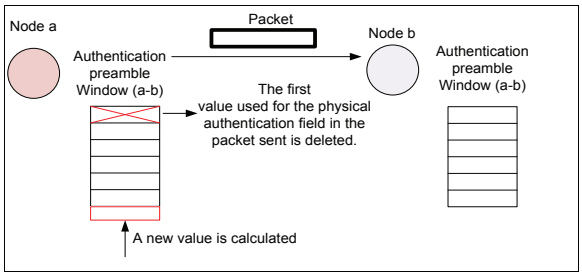

Figure 4.5: Updates of the Authentication Preamble Window of node 'A'.

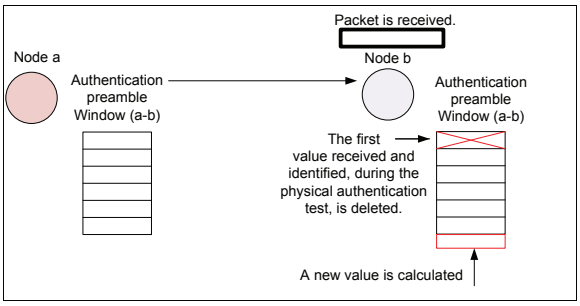

Figure 4.6: Updates of the Authentication Preamble Window of node 'B'.

after the synchronization field just to save as much energy as possible.

5. When the node completes the reception of the authentication preamble field, it starts the comparison between it and the results stored in all the authentication preamble windows.

6. If the result is identified as valid, the node continues the reception otherwise the packet is discarded.

7. If the result is correct and the packet also passes the MAC test, the receiver sends the encrypted ACK to the sender, if it is necessary, and updates the physical authentication window with a new value (Figure 4.6).

4.2.5 Possible Situation Considering all the Security Suite

In addition, the proposed solution, has to be designed as an adding scheme to improve the traditional security solutions at link layer; it is not proposed to replace them. Considering all the security services that we take into account for our security suite, it has to say the authentication preamble at physical layer and the MAC at link layer, when a node receives a packet three different situations are possible:

1. The reception fails after the authentication preamble test (Figure 4.7).

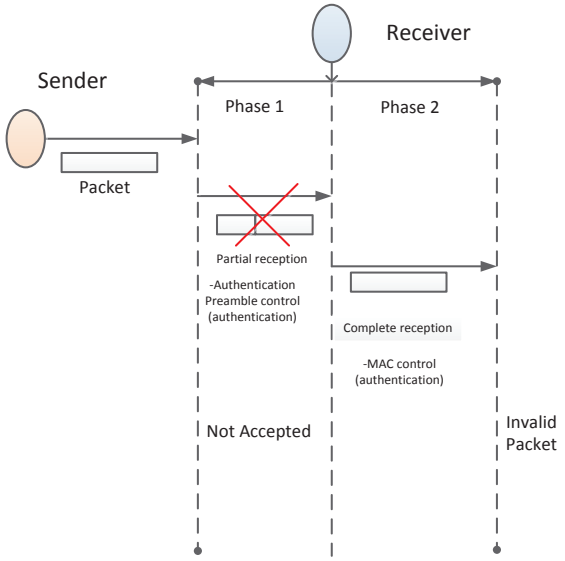

Figure 4.7: Invalid packet identified at physical layer.

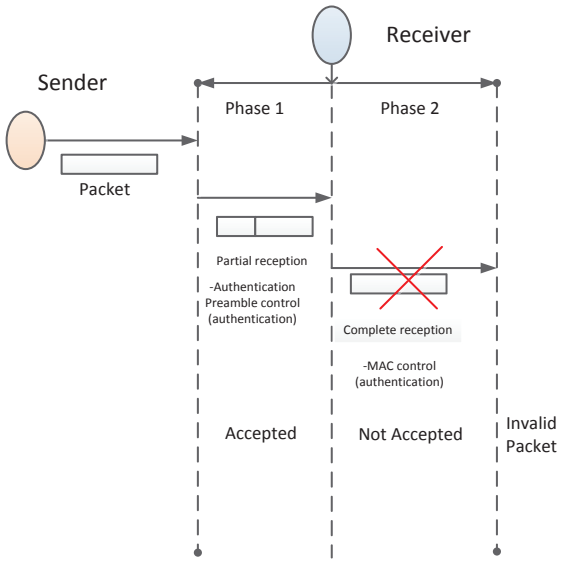

Figure 4.8: Invalid packet identified when the packet is completely received.

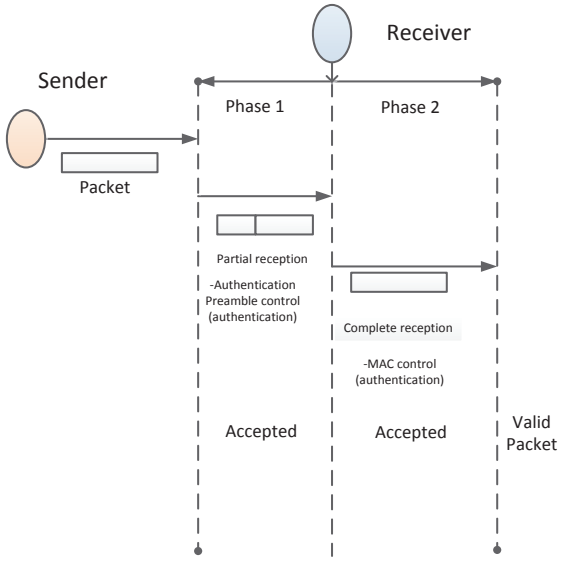

Figure 4.9: Valid packet.

2. The reception fails after the MAC test (Figure 4.8).

3. The received packet is accepted (Figure 4.9).

4.2.6 Comparative Table Among Secure-Availability Techniques

After a first presentation of the authentication preamble, we compare it with the literature to emphasize which are their differences. The proposed protocol offers a high level of security with an authentication test already at physical-layer. It is perfect for the scenario where availability is a top-priority security service and efficient communication protocols are important due to constrained devices. Commercial and industrial standards [Lopez et al., 2009], for example, use Message Authentication Code protection at link layer that only mitigate the DoS exhaustion attack; it is high protection process but with low efficiency. All prior schemes have some problems of different nature: security, energy consumption, flexibility, etc. Our solution is a good trade-off saving energy with a simple lightweight-mechanism of authentication and also it is not jeopardized by traditional attacks. *AP* limitation is represented by synchronization concerns: nodes have to be synchronized to allow traffic and control communication.

4.3 Authentication Preamble for Out-of-Sync

In this section we focus on the proposed protocol concerns. Indeed, the proposed authentication window solution is able to mitigate the out-of-sync vulnerability. However, since there is still a probability that all packets over this window get corrupted, especially in the presence

Table 4.1: Comparing of previous secure availability schemes with the proposed one.

Scheme	*Protection*	*Efficiency*	*Limitations*
Industrial solutions: Zigbee	High	Low	Only Mitigation
Network topology model	High	Medium	No Flexibility
Limited Link-layer rate	Medium	Medium	Limited Number of Messages
AP	High	High	Nodes Synchronization

of low signal to noise ratios, nodes can still reach an out-of-sync state. In this situation, a synchronization recovery protocol is to be started by the desynchronized nodes.

The synchronization protocol requires the nodes to periodically send at least a keep alive packet to their neighbors. That is to say that the receiver expects at least a legitimate packet from every neighbor every Δt. Therefore, if after $(W+1)$ Δt, a given receiver has not received any legitimate packet from a certain neighbor, it starts the synchronization recovery protocol with that neighbor. In the following, we describe the message protocol and how to physically authenticate these messages given that both ends are in an out-of-sync state.

4.3.1 AP During the Recovery Process

In order to propose a secure and efficient mechanism to secure availability for low-power networks during all the network's lifetime, we focus our efforts to identify which is the best re-synchronization method when an "out-of-sync" process is needed.

The first solution proposed consists into define an out-of-synchronization message with specific synchronization symbols and "in-clear" (without security mechanisms). As this solution could lead a possible hole in the security system, it is not recommended for scenario where DoS attacks are possible with reasonable probability. The proposed AP protocol was designed as secure mechanism to provide availability and siding as defense against DoS attacks at PHY-layer. If a no-secure message is implemented as mechanism against out-of-sync situation, malicious parties can take advantage of this message and sending it to realize exhaustion attacks. In this case, the valid parties are further deceived: the valid emitter will start the out-of-synchronization process without a real need.

The second solution is more expensive in terms of energy consumption but it provides a secure method. The potential receiver, besides the AP window described in Section 4.2, should keep and compute a potential out-of-sync AP every Δt.

As shown in Figure 4.10, the generation of this out-of-sync AP between node A and B is similar to that described in Section 4.2 for normal traffic AP. This verification field is the output result of an hash function that use confidential information as input data. In this case, the input

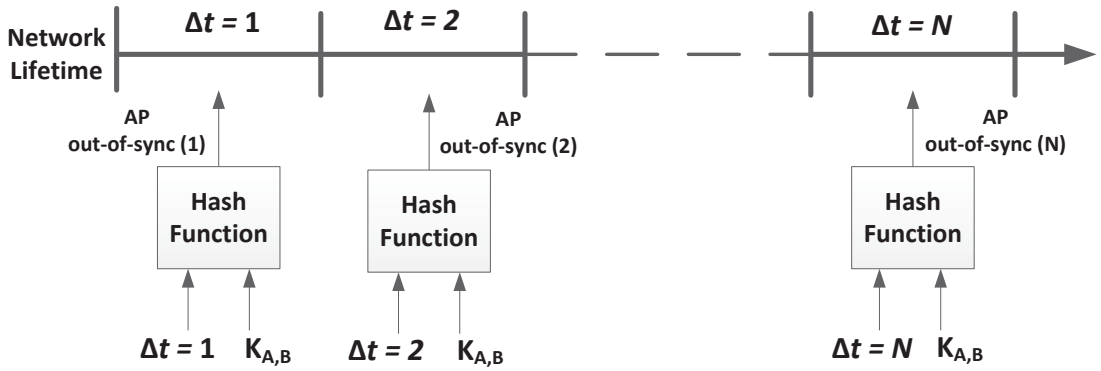

Figure 4.10: The Authentication Preamble method used to recover synchronization between couples of devices.

data are related to a counter field or timestamp of the current period and a specific pair-wise key shared between couples of nodes, $K_{A,B}$.

Since this mechanism is proposed for low rate networks, the receiver expects a legitimate message every Δt so it is able to pre-compute a different and valid out-of-sync AP every fixed period of time. This value will be used as a valid AP field at the physical layer only when a receiver realizes that it is out-of-sync with respect to a specific transmitter.

With the previous reasoning, every network node has not only to precompute an AP window during normal operation (Figure 4.1) but also an AP field for the potential packets needed during the synchronization recovery process. Regarding the two AP methods, the difference between them is really important for our intent.

The AP method for transmission/reception of traffic flow is a chain of results and for this reason infinite different values are possible. The limitation here is that the N-th value of the chain strictly depends on the $(N-1)$-th one; thus an out-of-sync situation is possible. However, an attacker would not be able to foresee the valid AP sequence even if AP results are repeated. This is because for every communication, the attacker is only able to sniff the first 32-bits of every hash function output, i.e., the transmitted AP field, while the remaining bits are safely stored in the node's memory. When the next AP field has to be computed, the node uses the total amount of bits of the previous output as new input. Summarizing, the correspondence between consecutive values of the AP fields for transmission/reception is very difficult to predict for an attacker due to our method's definition and the hash function characteristics.

As for the AP for out-of-sync packets, it is again the result of a hash function but the values are independent from each others; they depend on the network time-rate, the Δt, and on the secret key. The limitation here is represented by the maximum number of possible secure values for this field: i.e. with the Δt field of 32 bits, the maximum number of possible secure values of the AP for out-of-sync messages is 2^{32}. Assuming that the out-of-sync packets are not used with high frequency, this drawback does not really compromise the network operations.

Nevertheless, the AP for out-of-sync messages, it is a secure scheme because the attacker is not able to guess the valid key with simple sniffing methods and brute-force attacks and a a lightweight method because it is not able to influence the energy and memory consumption of the meters. Figure 4.11 shows the memory statement during an out-of-sync process. Here, it is possible to observe that while the APs for transmission/reception are out-of-sync, the AP for out-of-sync are always synchronized between couples of nodes at every Δt with minimum costs in terms of energy and memory.

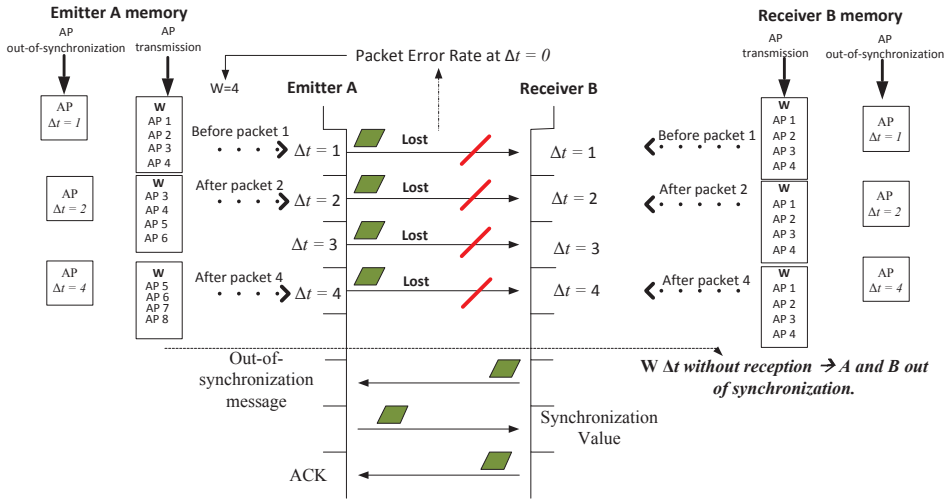

Figure 4.11: “Out-of-sync" situation.

4.3.2 Out-of-Sync Handshake Model

The out-of-sync three-way handshake is the specific method proposed to recover synchronization between pair of devices that are no longer able to exchange valid packets when the proposed *AP* protocol is implemented. Similar to the TCP-handshake socket connection, this method is referred to the “Out-of-SYN”, “SYN” and “ACK” messages which are defined below.

This mechanism is designed so that two devices are able to communicate and to exchange the parameters to recover the first valid *AP* value of the chain before re-starting the typical traffic flow. This handshaking is also designed so that both ends are able to initiate the process whenever they need to. For our study case, we adapt the IEEE 802.15.4e standard packet format to include the *AP* field at the PHY-layer; we like to underline that this insertion does not compromise the normal network functionalities.

Out-of-SYN Message When a node A realizes that it is in an out-of-sync state with node B, it sends to B an “Out-of-SYN” packet authenticated with the non-desynchronized AP for

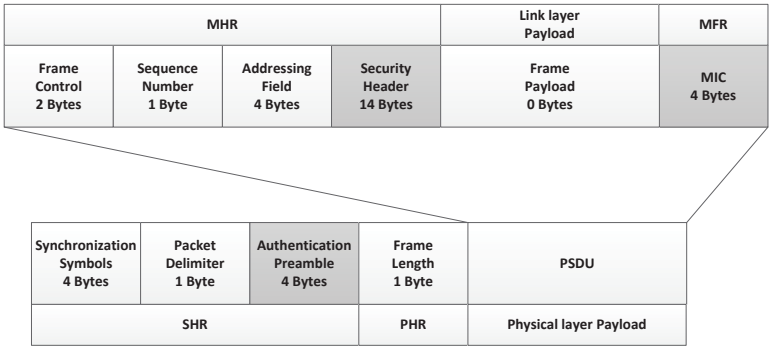

Figure 4.12: The out-of-sync message from the receiver to the emitter.

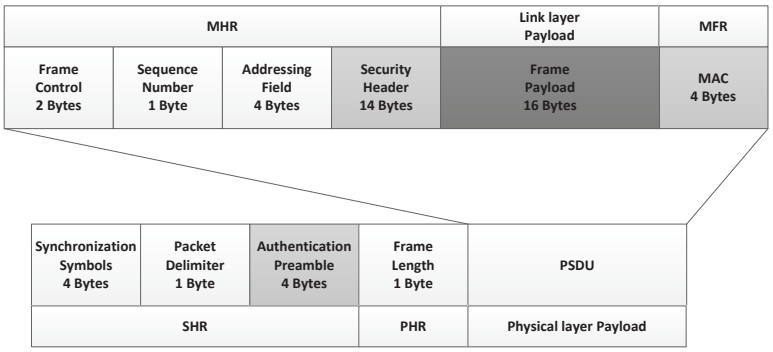

Figure 4.13: The sync message from the emitter to the receiver.

the recovery process, Figure 4.12. This frame is a control frame with an empty payload and thus it needs just authentication mode at link layer.

SYN Message When B receives the "Out-of-SYN" message, it sends to A a "SYN" message that contains the encrypted synchronization value, Figure 4.13. This value is the first valid element of the AP window for normal operation. Since it provides sensitive data, this frame must be both authenticated and encrypted.

ACK Message After the "SYN" message reception, A is able to resynchronize with B. However, as this state needs a confirmation, A has to report such to B by sending an "ACK" message, in which only authentication mode is required, Figure 4.14. Once B has received the "ACK", it will start again the normal traffic flow from the next Δt.

4.3.3 Out-of-Sync Handshake Overview

The main characteristics of the out-of-sync AP are its capacity: i) to be synchronized at every Δt, and ii) to be difficult to predict. This is very important to guarantee a secure mechanism able

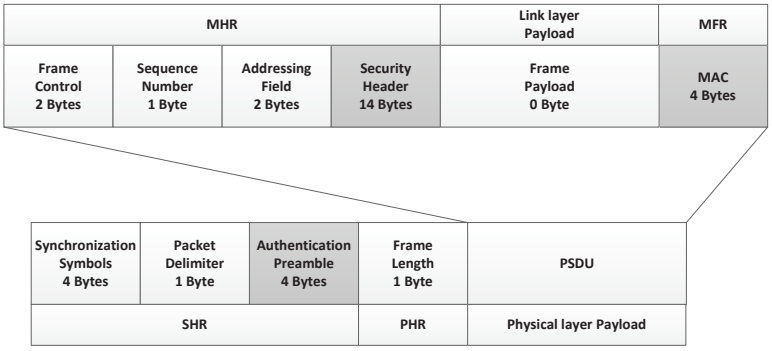

Figure 4.14: The confirmation message from the receiver to the emitter.

to synchronize a couple of nodes in every situation. Following, we will analyze the out-of-sync protocol, emphasizing how the system will react to any packets lost:

- **Situation A, N out-of-sync messages are lost.** Though N consecutive "out-of-sync" messages sent from the receiver 'B' to the emitter 'A' are lost, the same receiver will send new different messages with the out-of-sync AP computed with the corresponding Δt until the reception of a valid "SYN" value. Figure 4.15 shows this situation.

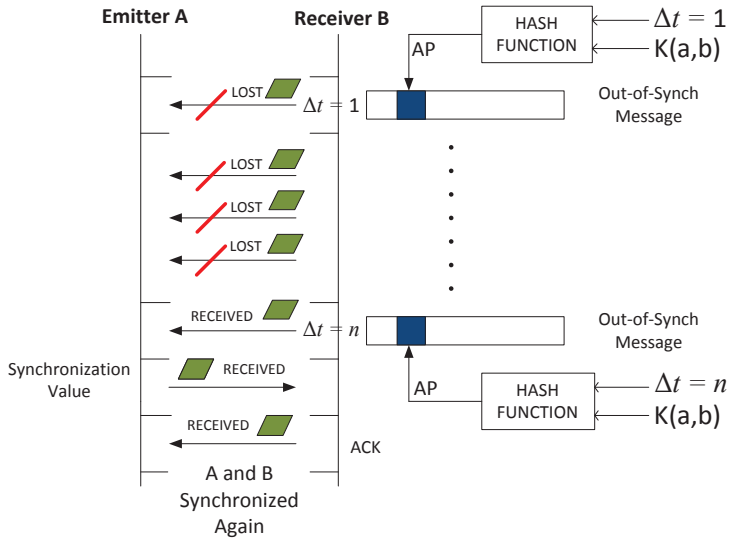

Figure 4.15: The transmission of the out-of-sync message from the receiver 'B' is concluded only when a valid "SYN" value, emitted from 'A', is received.

- **Situation B, N synchronization messages are lost.** Though N "SYN" messages sent from 'A' to 'B' are lost, the emitter 'A' will continue the transmission of this value (with

the correspondent out-of-sync AP) until the reception of the ACK from 'B'. Figure 4.16 shows this situation.

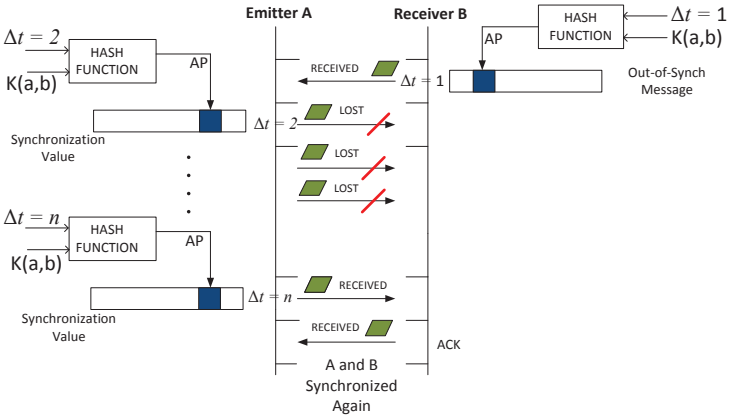

Figure 4.16: The emitter 'A' receives an out-of-sync packet from the receiver 'B'. The same emitter sent the "SYN" value until the ACK reception from 'B'.

- **Situation C, N ACKs messages are lost.** The receiver 'B' is now able to receive a traffic message from the emitter 'A' but the same emitter needs a confirmation of this state. In this case, the maximum number of possible ACKs sent from 'B' to 'A' is limited, situation D. Figure 4.17 shows this situation.

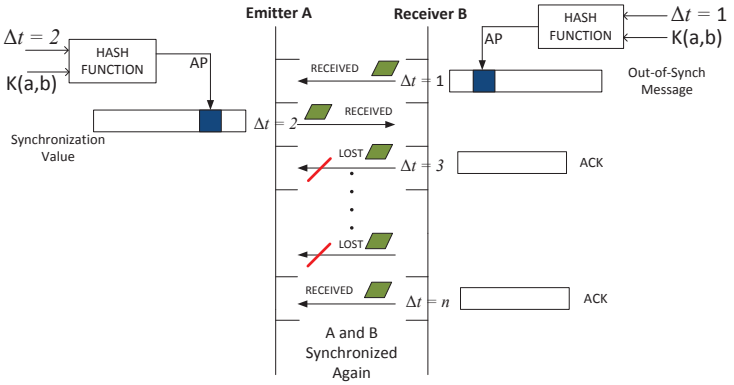

Figure 4.17: The receiver 'B' receives the "SYN" value from the emitter 'A'. The same receiver sent the ACK until the reception of a valid traffic packet from the emitter 'A'.

- **Situation D, W consecutive normal traffic packets are lost.** The maximum possible ACKs sent form 'B' to 'A' is correspondent to the AP window length (strictly dependent on the packet error rate of the channel). If after W Δt rates the receiver 'B' dose not receive any valid messages from the emitter 'A', it will start again the "out-of-synchronization" process. Instead, if a valid message sent from 'A' is received from 'B', this means that the meters are synchronized again. This restriction is mandatory to avoid energy wasting when the emitter 'A' receives the ACK sent from 'B' but all the following W traffic messages sent form 'A' to 'B' are lost consecutively. Figure 4.18 shows this situation.

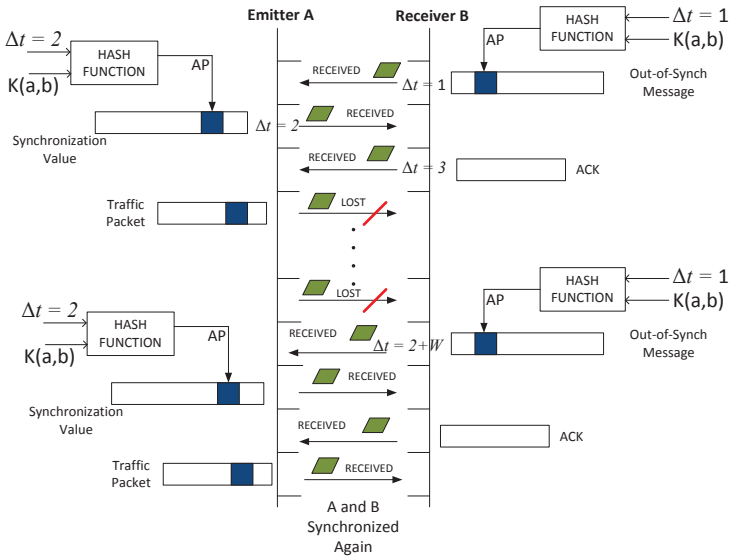

Figure 4.18: If 'B' is not able to receive a normal traffic packet after W Δt, it has to start again the out-of-sync process

Finally, further studies are necessary to evaluate the concerns that are caused by the nodes' clocks (clock drift) on the out-of-sync process. This last concern has been left for future analysis.

4.4 Energy Consumption Analysis & Optimal Setup Parameters

As previously mentioned, the proposed AP mechanism allows energy saving not only when receiving an exhaustion attack, but also during the normal operation of the network by discarding non-intended non-malicious packets.

During a node exhaustion attack, the attacker usually floods the victim with fake packets. As a result, under this situation the benefits of using our proposal seem clear, as long as it allows the victim to not receive and process the entire fake packets. However, its cost under normal

operation, that involves to actually send/receive an extra AP in every packet and to check the AP of every incoming package, should be more in-depth analyzed.

In this section, we provide an analysis of the energy cost derived from the use of the AP method. We first compute the optimal value of the AP window W according to realistic communication conditions by taking the error rates into account. Next, based on the device's capabilities in terms of memory and microprocessor-power, we study the optimal and maximum number of in-range neighbors per node which lead to energy saving under normal operation. Later, in Sections 4.4.3 and 4.4.4, we use this analysis to get real values in an example scenario.

4.4.1 Optimal AP Window Length

Under normal operation, between every pair in a communication link, the AP of every incoming packet is checked against the set of W precomputed AP for the emitter. If more than W APs from the same emitter arrive corrupted, the devices reach an out-of-sync state and must resync again. Obviously, the higher the number of precomputed APs, the lower the probability to get into out-of-sync state.

However, even if longer windows save the node to perform the synchronization recovery process, at the same time, they imply high mean number of load/compare operations. This is because when a generic frame is received, its AP field is loaded and then compared with all the potential APs that were previously precomputed. All these operations are able to influence the node's performance. In conclusion, to conduct a fair analysis, we have to identify the tradeoff between the benefits of a system that rarely gets in an out-of-sync state and the extra costs that arise due to the use of a long window.

The presented analysis relies on the following assumptions:

1. The channel can corrupt the AP field. When this happens, the authentication test fails and the system is involved in an extra-operational cost due to AP field comparisons.

2. The packets for the synchronization recovery handshake are successfully transmitted in a single attempt. This assumption relies on the fact that these packets are much smaller and hence much likely to go through.

With that said, the mean energy consumption of the system $\overline{E}$ can be obtained as in (4.1) as the sum of the mean energy consumed when the nodes are synchronized and in normal operation $\overline{E}_{syn}$ and the mean energy used when they reach an out-of-sync state $\overline{E}_{\cancel{syn}}$.

$$\overline{E} = \overline{E}_{syn} + \overline{E}_{\cancel{syn}} \tag{4.1}$$

During normal operation, the extra energy consumed by the system depends only on the number of comparisons per received AP that the system computes. While the comparisons

would likely be implemented in hardware, we opt for the worst case where all the comparisons are sequentially made in software. As a result, $\overline{E}_{syn}$ can be expressed as in (4.2), where p as in (4.3) is the probability to receive a corrupted AP field, BER is the bit error rate of the link after any error correcting code is applied, l_{AP} is the length of the AP field, $\overline{E}_{cmp}$ is the average energy used to perform a comparison, and W is the window length.

$$\overline{E}_{syn} = (1-p)\overline{E}_{cmp}\sum_{i=1}^{W}((i-1)W+i)p^{i-1} \tag{4.2}$$

$$p = (1-(1-\mathrm{BER})^{l_{AP}}) \tag{4.3}$$

Notice that both BER and p are probabilities defined between 0 and 1.

With that said, a node reaches an out-of-sync state when the APs of W packets are corrupted, thus with probability p^W. Therefore, it performs a comparison for each one of the W received packets before starting the synchronization recovery handshake. As a result, $\overline{E}_{\cancel{syn}}$ can be expressed as in (4.4), with $\overline{E}_{rec}$ the energy consumption used to synchronize the nodes implementing the recovery handshake model presented in Section 4.3.2.

$$\overline{E}_{\cancel{syn}} = p^W(\overline{E}_{rec} + W^2 \cdot \overline{E}_{cmp}) \tag{4.4}$$

Since the synchronization process consumes by far more than the operation of comparing APs or bits sequences, we can conclude that the optimal AP window length is obtained when the mean number of comparisons by the cost of comparison meets the cost of resynchronizing any two neighbors. On one hand, if W is lower than its optimal value, energy is wasted due to avoidable resynchronization processes. On the other hand, when W is higher, the energy is wasted due to the large number of comparisons per received AP.

From the previous reasoning, the optimal length of the AP window is the value of W that minimizes (4.1), which in this case can be obtained by equaling to 0 the first derivative with respect to W. Known that a geometric series can be expressed as

$$\begin{aligned}\sum_{i=1}^{W} i \cdot p^{i-1} &= \frac{\partial(\sum_{i=i}^{W} p^i)}{\partial p} \\ &= \frac{\partial((p-p^{W+1})/(1-p))}{\partial p} \\ &= \frac{1-p^W}{(1-p)^2} - \frac{W \cdot p^W}{1-p},\end{aligned}$$

the result of this derivative can be expressed as in (4.5), where for the sake of clarity a represents $\overline{E}_{rec}$ and b represents $\overline{E}_{cmp}$. Consequently, the optimal AP window length can be obtained by solving (4.6), with $\alpha = \frac{(a\,p+b-a)\log p+b}{b\log p}$ and $\beta = \frac{p}{\log p}$.

$$\frac{\partial E}{\partial W} = \frac{b\,W\,p^W \log p + p^W\left((a\,p+b-a)\log p+b\right) - b\,p}{p-1} \tag{4.5}$$

$$(W_{opt}+\alpha)p^{W_{opt}} = \beta \tag{4.6}$$

Obviously, the expression in (4.6) does not have a general solution and thus it should be analyzed for the specific values of a, b and p obtained from a given implementation. Section 4.4.3 deals with it in a specific IEEE 802.15.4e scenario.

4.4.2 Maximum Allowed Neighborhood

In this subsection, we analyze under which conditions the use of the AP method actually saves energy. This study is crucial in order to explore the costs of the proposed solution and to identify when these costs $\overline{E}^{AP}_{costs}$ exceed the energy savings $\overline{E}^{AP}_{savings}$. In conclusion, our proposal is efficient when inequity in (4.7) is satisfied.

$$\overline{E}^{AP}_{savings} - \overline{E}^{AP}_{costs} > 0 \tag{4.7}$$

Intuitively, the energy costs are related to the extra AP field bytes sent/received at every transmission/reception process and to the number of AP comparisons for every received packet. In an standard wireless M2M network, normal nodes (not sinks or base stations) transmits data at specific rates, and thus we can assume that in a given period a node transmit once while senses non-intended data from its N in-range neighbors. Therefore, these costs can be expressed as in (4.8), with $\overline{E}_{tx}$ and $\overline{E}_{rx}$ the mean energy costs to transmit and receive, and $N \cdot W$ the number of comparisons for every received packet.

$$\overline{E}^{AP}_{costs} = l_{AP} \cdot \overline{E}_{tx} + N(\overline{E}_{rx} \cdot l_{AP} + N \cdot W \cdot \overline{E}_{cmp}) \tag{4.8}$$

The energy saved by the use of the AP method $\overline{E}^{AP}_{savings}$ as in (4.9) depends on the number of non-intended packets received N, and the bytes $l_{\not{rx}}$ that are not received due to AP verification failures. Notably, the former is influenced by the neighbor nodes cardinality while the latter

is influenced by the number of comparisons that every node has to compute at every reception. Obviously the energy savings are bounded to the speed of the AP verification, being the proposed mechanism only useful if the AP verification is finished before the entire packet reception.

$$\overline{E}^{AP}_{savings} = N \cdot l_{\not{rx}} \cdot \overline{E}_{rx} \tag{4.9}$$

$l_{\not{rx}}$ can be expressed as in (4.10) as the subtraction of the bytes that are received during the AP verification l_{cmp} from the total length of the link-layer frame l_{Link}.

$$l_{\not{rx}} = l_{Link} - l_{cmp} \tag{4.10}$$

In order to estimate l_{cmp}, we should obtain the average time required to make the $N \cdot W$ comparisons and then check how many bytes have been received during this time. As a result, l_{cmp} can be denoted as in (4.11), with B the bit rate, CC the CPU cycles required to do a byte comparison, and f_{CPU} the microprocessor clock.

$$l_{cmp} = B \cdot N \cdot W \frac{l_{AP} \cdot CC}{f_{CPU}} \tag{4.11}$$

At this point, inequity (4.7) can be expressed as in (4.12). Assuming that the first order coefficient is greater than second order coefficient, which is very likely to happen, (4.12) clearly shows that the optimal number of in-range neighbors can be expressed as in (4.13). Additionally, the maximum number of in-range neighbors that guarantees savings is the positive value of N that makes (4.12) equal to 0.

$$\begin{aligned} -N^2(\overline{E}_{rx} \cdot B \cdot W \frac{l_{AP} \cdot CC}{f_{CPU}} + W \cdot \overline{E}_{cmp}) \\ + N \cdot \overline{E}_{rx}(l_{MAC} - l_{AP}) \\ - l_{AP} \cdot \overline{E}_{tx} > 0 \end{aligned} \tag{4.12}$$

$$N_{opt} = \frac{(l_{Link} - l_{AP})\overline{E}_{rx}}{2(\overline{E}_{rx} \cdot B \cdot W \frac{l_{AP} \cdot CC}{f_{CPU}} + W \cdot \overline{E}_{cmp})} \tag{4.13}$$

4.4.3 Energy Consumption and Recommended Network Density

According to Equation 4.6, we compute the mean energy needed by the system to implement the proposed solution. The chosen values for the different parameters are shown in Table 4.2, which are typical of 802.15.4 devices, and the packets used to complete the synchronization process are those defined in Section 4.3.2. Figure 4.19 depict the obtained results for different BER values ranging from $3 * 10^{-5}$ to $9.5 * 10^{-5}$ (average maximum BER in 802.15.4 as per [Ahn

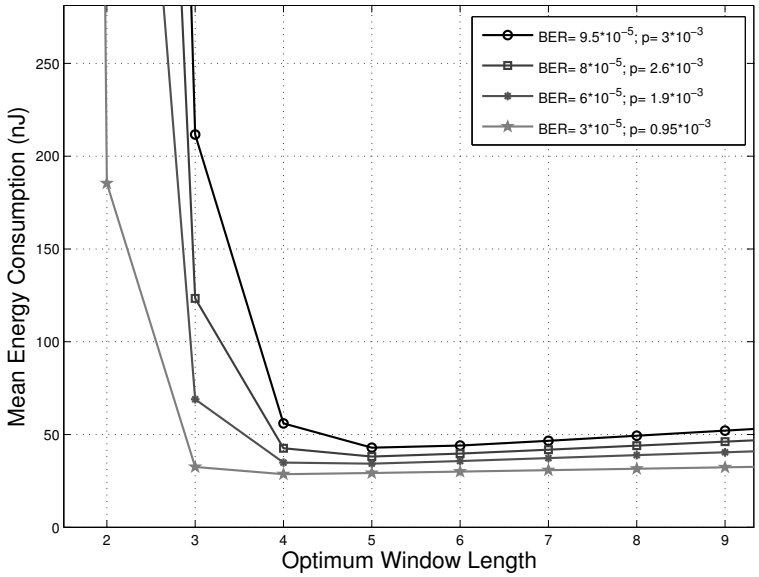

Figure 4.19: Average energy consumption for a varying AP window and different values of p. The minimum of each curve, since it involves the lower energy consumption, represents the optimum AP window length.

Table 4.2: Power and Energy costs with Dust Network LTP5901/LTP5902-IPM [Dust Smart Mesh Network, 2012]. The following characteristics are measured with V_{SUPPLY} = 3.6 V at 25° C.

Fields	**Value**
Current to Rx	4.5 mA
Current to Tx	5.4 mA
Current for the CPU	2.4 mA
V_{core} CPU	1.8 V
Data Rate (B)	250 Kbit/s
Time to Rx and Tx 1 Byte	32 μs
Cycles to Compare (CC)	5 Cycles/Byte
CPU frequency (f_{CPU})	7.37 MHz
Energy to Rx ($\overline{E}_{rx}$)	0.518 μJ/Byte
Energy to Tx ($\overline{E}_{tx}$)	0.622 μJ/Byte
Energy to Compare ($\overline{E}_{cmp}$)	11.72 nJ/Byte

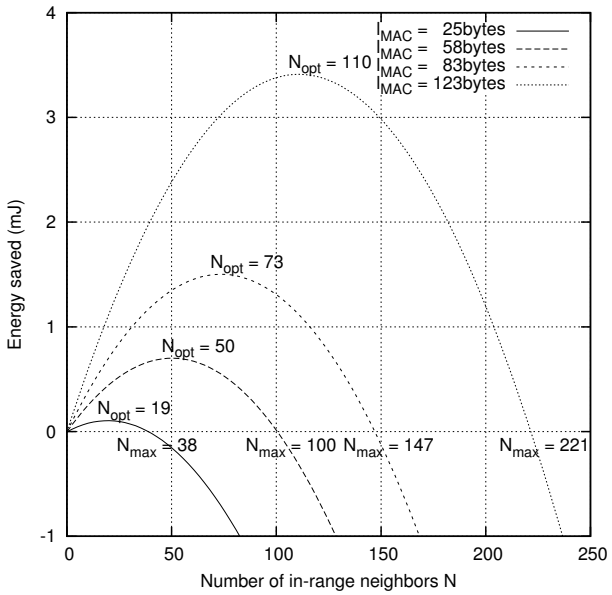

Figure 4.20: Energy savings for a varying link-layer frame length l_{Link} and number of in-range neighbors N. Network devices are the ones defined in Table 4.2

and Heidemann, 2002]). As it can be seen, the optimal value of the window AP size W_{opt} that maximizes the energy savings even in the worst conditions is bounded to $W_{opt} = 5$.

Once the AP window is obtained, it is still necessary to establish under which conditions our proposal saves energy even during normal network operation. From the study in Section 4.4.2, it is easy to obtain the desirable number of in-range neighbor nodes N in this specific 802.15.4e scenario. Figure 4.20 shows the energy savings with the following assumptions: the size of the MAC frame l_{MAC}, as shown in Figure 4.2, ranges from 25 bytes (its minimum value) to 123 bytes; *CC* is set to 5 cycles/Byte, 2 cycles/Byte to “load" and 3 cycles/Byte to “branch if equal”; and AP window is set to its optimal value in the worst conditions, i.e. BER of $9.5 * 10^{-5}$ with W=5.

According to Figure 4.20, the optimal and maximum number of in-range neighbors varies with the average link-layer frame length l_{Link}. Obviously, larger packets produce more savings since early discard saves to receive more bytes. In any case, even in the worst case, with the smallest frame lengths, our proposal, besides protecting against node exhaustion attacks, actually saves energy for a maximum number of in-range neighbors of 38, and can easily get to 100 for larger Link-layer frames. This result shows the wide scope of this proposal and its applicability in a great variety of scenarios.

4.4.4 Memory Requirements

The cost of the physical authentication in terms of memory storage are mainly related to: i) the number of in-range neighbors N; ii) the length of the AP l_{AP}; iii) the number of precomputed APs per neighbor node, i.e. a window of W APs for normal traffic flow and 1 AP for the synchronization recovery protocol; iv) the length of the pairwise keys l_{key}; and v) the number of stored keys per neighbor node (3 pairwise keys per in-range neighbor are sufficient for our intents; one key for the *AP*, one key for the *AP* for out-of-sync state and one key to refresh the valid keys). That is to say that the total storage costs M can be computed as in (4.14).

$$M = N\left(3l_{key} + (W+1)l_{AP}\right) \tag{4.14}$$

The worst case for this example scenario, as shown in Figure 4.20, would be a topology with a $N = 221$ in-range neighbors. In this case, with an authentication preamble of $l_{AP} = 32$ bits, an AP window $W = 5$, and pairwise keys of $l_{key} = 128$ bits, the necessary storage costs are of $221(128 \cdot 3 + 6 \cdot 32)$bits ≈ 15.54KBytes. Storage requirements are assumable by most of the nowadays M2M devices, and specifically the ones in Table 4.2.

Chapter 5

Secure Key Management Protocol

"The future belongs to those who believe in the beauty of their dreams"
Eleanor Roosevelt.

Contents

5.1 Introduction to Key Management Issue in M2M Networks

Current cryptographic mechanisms can be considered as the strongest link in the security chain. However, cryptographic mechanisms rely on the secure management of the necessary keying

material. Keys are analogous to the combination of a safe. If the combination becomes known to an adversary, the strongest safe provides no security against penetration. Similarly, poor Key Management (KM) may easily compromise strong algorithms. Ultimately, the security of information protected by cryptography directly depends on the strength of the keys, the effectiveness of mechanisms and protocols associated with keys, and the protection given to the keys.

In this Chapter, we focus on KM mechanisms describing their intents, the key materials that are used with such mechanisms, the security requirements that are provided with such key materials, the cryptoperiod for each key, the state of the art and the proposed solution related to key generation and key updating.

5.1.1 KM Intent

Cryptography concerns itself with securing information so that unauthorized individuals cannot understand the messages sent by valid actors of a network and KM is the mechanism responsible for the proper and secure distribution, creation, and revocation of the keys used from cryptographic algorithms. With that said, is clear that cryptography can be rendered ineffective by the use of inappropriate algorithm pairing, poor physical security and weak KM protocol. In this Chapter we focus on the KM starting presenting the following issues:

- **Data protection**. Following, according to their uses, we classify different types of keys and other cryptographic information.
- **Security requirements**. We then present which protection requirements must be provided for the different classes of keys, integrity, confidentiality, etc.
- **Phases**. Consequently, we introduce in which phases KM is subdivided, pre-deployment, distribution and special messages phase.
- **Cryptoperiod**. Finally, we define the importance of cryptoperiod for the implementation of reliable communication systems.

5.1.2 Cryptographic Key

KM has to create, distribute and revoke several different types of cryptographic keys each used for a different purpose. In this sub-section are classified all the most used cryptographic keys and in which way they should be used:

- **Authentication key**. Authentication keys are used with symmetric/asymmetric key algorithms to provide assurance of the integrity and source of messages, communication sessions, or stored data. Considering symmetric solutions, only one symmetric authentication key is used from both sender and receiver side instead, taking into account asymmetric

solutions, a pair of public/private keys is defined. Public key is used from the sender side instead, private key is used to verify the integrity (and the authenticity) of the message received.

- **Data encryption key**. These keys are used with symmetric/asymmetric key algorithms to apply confidentiality protection to information. Also here asymmetric solutions implement a pair of private/public keys.

- **Authorization key**. Authorization keys are used to provide privileges to an entity using a symmetric/asymmetric cryptographic method. The authorization key is known by the entity responsible for monitoring and granting access privileges for authorized entities and by the entity seeking access to resources. Considering symmetric solution there is a unique key. In the case of asymmetric solutions, the private key is used to provide privileges instead the public key is used to verify if these privileges match with the sender.

- **Random number generation keys**. These symmetric/asymmetric keys are keys used to generate random numbers.

- **Symmetric key wrapping key**. Symmetric key wrapping keys are used to encrypt other keys using symmetric key algorithms. Key wrapping keys are also known as key encrypting keys.

- **Symmetric Master key**. A symmetric master key is used to derive other symmetric keys (e.g., data encryption keys, key wrapping keys, or authentication keys) using symmetric cryptographic methods. Normally a master key is a long-term key while data encryption key is a short-term key, such as in Zigbee PRO [Zigbee, 2010].

- **Symmetric key agreement key**. These symmetric keys are used to establish keys (e.g., key wrapping keys, data encryption keys) and, optionally, other keying material (e.g., Initialization Vectors) using a symmetric key agreement algorithm.

- **Private and Public key transport key**. Private key transport keys are the private keys of asymmetric key pairs that are used to decrypt keys that have been encrypted with the associated public key using a public key algorithm. While public key transport keys are the public keys of asymmetric key pairs that are used to encrypt keys using a public key algorithm. Key transport keys are usually used to establish keys (e.g., key wrapping keys, data encryption keys) and other keying material (e.g., Initialization Vectors)

- **Private and Public ephemeral key agreement key**. Private and Public ephemeral key agreement keys are the private and/or public keys of asymmetric key pairs that are used only once to establish one or more keys (e.g., key wrapping keys, data encryption keys) and other keying material (e.g., Initialization Vectors).

- **Private and Public static key agreement key**. Private and Public static key agreement keys are the private keys of asymmetric key pairs that are used to establish keys (e.g., key wrapping keys, data encryption keys) and other keying material (e.g., Initialization Vectors). They are different from the Private and Public ephemeral keys agreement key because, as they are "static", they can be used for more than one session.

- **Private signature key and public signature verification key**. Private signature keys are the private keys of asymmetric key pairs that are used by public key algorithms to generate digital signatures with possible long-term implications. When properly handled, private signature keys can be used to provide authentication, integrity and non-repudiation. Instead, a public signature verification key is the public key of an asymmetric key pair that is used by a public key algorithm to verify digital signatures, either to authenticate a user's identity, to determine the integrity of the data, for non-repudiation, or a combination thereof.

Even if all these keys are implemented in very different applications, all of them should be used for only one purpose. The following feature are important considerations for properly define the key uses:

- **Multiple process uses**. The use of the same key for two or more different cryptographic processes may weaken the security provided by one or both of the processes.

- **Key compromised consequences**. Limiting the use of a key limits the damage that could be done if the key is compromised.

- **Cross-uses of the same key**. Some uses of keys interfere with each other. For example, consider a key pair used for both key transport and digital signatures. In this case the private key is used as both a private key transport key to decrypt data encryption keys and a private signature key to apply digital signatures. It may be necessary to retain the private key transport key beyond the cryptoperiod of the corresponding public key transport key in order to decrypt the data encryption keys needed to access encrypted data. On the other hand, the private signature key should be destroyed at the expiration of its cryptoperiod to prevent its compromise. In this example, the longevity requirements for the private key transport key and the private digital signature key contradict each other.

These principles do not preclude using a single key in cases where the same process can provide multiple services. This is the case, for example, when a digital signature provides non-repudiation, authentication and integrity protection using a single digital signature, or when a single symmetric data encryption key can be used to encrypt and authenticate data in a single cryptographic operation.

5.1.3 Protection of Security Requirements

Cryptographic keying material is defined as the cryptographic key and associated information required to the key uses. The specific information varies depending on the type of key. The cryptographic keying material must be protected in order for the security services to be effective. The type of protection needed depends on the type of key and the security service for which the key is used. Whenever the keying material exists external to a cryptomodule, additional protection is required.

Keying material should be available as long as the associated cryptographic service is required and the protection that should be provided are:

- **Integrity**. Integrity must be provided for all keying material. Integrity protection always involves checking the source and format of received keying material. This security requirement can be provided by cryptographic integrity mechanisms (e.g. cryptographic checksums, cryptographic hashes, MACs, and signatures), non-cryptographic integrity mechanisms (e.g. CRCs, parity, etc.) or physical protection mechanisms. When cryptographic integrity mechanisms are implemented, also authentication of the keys' sender is provided (the owner of a valid key is recognized as an authenticate actor).

- **Confidentiality**. The confidentiality of all symmetric and private keys must be protected. Public keys generally do not require confidentiality protection. When the symmetric or private key exists internal to a validated cryptomodule, confidentiality protection is provided by the cryptomodule. When the symmetric or private key exists external to the cryptomodule, confidentiality protection is provided either by encryption (e.g., key wrapping) or by controlling access to the key via physical means (e.g. storing the keying material in a safe with limited access).

- **Associated protection**. Associated protection methods have to be provided for a cryptographic security service by ensuring that the correct keying material is used with the correct data in the correct application or equipment.

- **Key validity**. Assurance of domain parameter and public key validity provides confidence that the parameters and keys are arithmetically correct.

- **Private key validation**. Assurance of a private key validation provides assurance that the owner of a public key actually possesses the corresponding private key.

- **Cryptoperiod**. The period of protection for cryptographic keys, associated key information, and cryptographic parameters (e.g. initialization vectors) depends on the type of key, the associated cryptographic service, and the length of time for which the cryptographic service is required. The period of protection includes the cryptoperiod of the key. The period of protection is not necessarily the same for integrity as it is for confidentiality.

Integrity protection may be required until a key is no longer used, but confidentiality protection may be required until the key is destroyed.

5.1.4 KM Phases

The cryptographic KM life cycle can be subdivided into three phases in which determinate KM functions are typically performed; for completeness, we define these functions as necessary operations for the management of the keys and their associated attributes (key material).

Going into details, attributes are leveraged by applications to select the appropriate cryptographic key for a particular service and they might include a system user identity associated with that key or the types of information that this user is authorized to access. While these attributes do not appear in cryptographic algorithms, they are crucial to the implementation of applications and protocols. However, the three phases are:

- **Registration phase**. The keying material is not yet available for normal cryptographic operations. Keys may not yet be generated, or may be in the pre-activation state. System attributes are established during this phase as well. In this phase the typical functions are: key material registration, key material storage and key material association with the corresponding system attributes.

- **Operation phase**. The keying material is available for normal uses. Keys are in the active state and they can be used for their purposes. In this phase the typical functions are encryption/decryption, integrity test, key recovery and public/private key pair and digital certificate validation.

- **Special messages phase**. The keying material is no longer in normal use, but access to the keying material is still possible. This phase is the most important part of the KM process because it permits to solve out possible threats if they are identified (e.g., revoking a key), to join a successful registered actor (key distribution/agreement) and to re-keying a old key (re-keying mechanism). In this phase the typical functions are: re-keying mechanism, key revocation, key distribution/agreement.

The three phases of KM are specified in the following sub-sections:

Registration phase

This phase is characterized from one basic function: user/device registration. During user/device registration function, an entity interacts with a registration authority or with a simple neighbor device to become an authorized member of a security domain. This is possible implementing a pre-activation key method (a pre-activation key should be stored in the user/device memory during the pre-deployment network phase) or alternately simply providing ID information; the first method is more secure than the second one because confidential information are requested.

After that, security infrastructures may associate the identification information with the entity's keys and a user identifier or device name may be established to identify the member during future transactions (to provide privacy security service). The entity may also establish various attributes during the registration function, such as email addresses or role/authorization information (key material attributes). After the registration function a key distribution/agreement function is implemented to provide valid key material to the new entity.

Since applications will depend upon the identity established during registration process, it is crucial that the registration authority establish appropriate procedures for the validation of identity. Identity may be established through an in-person appearance at a registration authority, or may be established entirely out-of-band. The strength (or weakness) of a security infrastructure will often depend upon the identification process.

Operation phase

In this second phase the keying material is available and in normal use. Keys are in the active state and may be used for the normal security processes. After registration phase, the key material should be available either on the device's cryptomodule that uses that material, and on a readily accessible storage media. When the keying material is required for operational use (encryption/decryption, integrity, etc.)and it is not present in active memory within the device's cryptomodule, it is acquired from immediately accessible storage.

In addition, to provide continuity of operations when the keying material becomes unavailable for use from normal operational storage during its cryptoperiod (e.g., because the material is lost or corrupted), keying material may need to be recoverable. If an analysis of system operations indicates that the keying material needs to be recoverable, then the keying material has to be either backed up or the system has to be designed to allow reconstruction of the keying material. Acquiring the keying material from backup or by reconstruction is commonly known as key recovery.

At the end of a key's cryptoperiod or if key recovery does not fix the key compromise problem, a new key needs to be available to replace the old one (if operations must continued); this can be accomplished by re-keying mechanisms during special message phase. Finally, a key has to be destroyed as soon as that key is no longer needed in order to reduce the risk of exposure.

Special messages phase

In this third phase, the key material is no presented in the device's memory (even if the devices are already registered) or it is no longer in normal use. In the former case, a key establish process needs to be requested from the new entity, while in the latter, access to the keying material is still possible and the keying material may be used for process only in certain particular circumstances.

Taking into account the key establish function, it involves the generation and distribution, or the agreement of keying material for communication between entities. All keys have to be generated and validated from the cryptographic module and during the key establishment process, some of the keying material may be in transit or retained locally. In either cases, the keying material has to be protected with integrity and confidentiality security services.

On the other hand, when the key material are presented in the device's memory but its state is no longer active, a re-keying mechanism or revocation process is implemented. Re-keying method refers to the process of changing the crypto-key in order to limit the amount of data protected with the same key or because the same key is recognized as compromised (compromised for human or machine errors and not under external/internal attacks). Instead, revocation process, is a method used to explicitly revoke a symmetric key or the public key of a key pair; in the latter case, the public key can be also revoked as well. Key revocation is implemented when a possible attack is identified (e.g., node compromised, etc.) and the node's key material has to be destroyed. If the compromised-key is a group-wise key, thus all the nodes of this group have to be advised of the dangerous; they must destroy the key in order to isolate the possible threat.

In conclusion, if a key is identified as compromised, different situations are possible: if only the secret key is known from an attacker, thus the system will implement a re-keying mechanism, instead, if an attacker is able to have physical access to the information stored in an entity thus, a revocation process is computed. In the first case the entity is still an active part of the network, on the contrary, in the second case the node is isolate from the security domain.

5.1.5 Cryptoperiod Definition

A cryptoperiod is the time span during which a specific key is authorized for use by legitimate entities. In order to find out a reliable cryptoperiod and to not increase the number of possible threats, some limitations must be considered:

- **Protected information**. For a reliable communication system, if the key is available for cryptanalysis, the amount of information protected by a such key should be limited.
- **Exposure uses**. If a single key is compromised, the amount of exposure should be limited.
- **Algorithm**. Taking into account a particular algorithm, its use should be limited to its estimated effective lifetime.
- **Access to memory**. The time available for attempts to penetrate physical, procedural, and logical access mechanisms that protect a key from unauthorized disclosure should be limited.
- **Time**. Finally, the key lifetime should be limited considering computationally intensive cryptanalytic attacks (in applications where long-term key protection is not required).

As the nature of cryptographic protocols is profoundly different, not all these limitations have the same importance to select the proper cryptoperiods for a given application; sometimes cryptoperiods are defined by an arbitrary time period and other times they are defined by maximum amount of data protected by the key.

Nevertheless, in general, short cryptoperiods enhance security because normally cryptographic algorithms are less vulnerable to cryptanalysis if the adversary has only a limited amount of information encrypted under a single key. On the other hand, where key distribution methods are subject to transmission/human/machine errors, more frequent key changes might actually increase the risk of exposure. In these cases, especially when very strong cryptography is employed, it may be more prudent to have fewer, well-controlled key distributions rather than more frequent, poorly controlled key distributions.

Moreover, where strong cryptography is employed, physical, procedural, and logical access protection considerations often have more impact on cryptoperiod selection than do algorithm and key size factors. In the case of approved algorithms, modes of operation, and key sizes, adversaries may be able to access keys through penetration or compromission of a system with less expenditure of time and resources than would be required to mount and execute a cryptographic attack (brute force attack).

In conclusion, in addition of the limitations previously presented, a proper cryptoperiods selection must taking into account also the following factors:

- The strength of the cryptographic mechanisms (e.g., the algorithm, key length, block size, and mode of operation.
- The operating environment.
- The volume of information flow or the number of transactions.
- The security life of the data.
- The security function (e.g., data encryption, digital signature, key production or derivation, key protection).
- The re-keying method.
- The key update or key derivation process.
- The number of nodes in a network that share a common key.
- The number of copies of a key and the distribution of those copies.
- And finally, the threat to the information (e.g., who the information is protected from, and what are their perceived technical capabilities and financial resources to mount an attack).

5.1.6 Symmetric KM

Symmetric key communication systems use a single key to both apply cryptographic protection to data (e.g., encrypt) and to those processes that are protecting data (e.g., decrypt). Thus, a single key must be shared between two or more entities that need to communicate. As any cryptographic systems, also symmetric key solutions have advantages and disadvantages. Symmetric cipher systems, relative to asymmetric ciphers, handle large amounts of data more efficiently although their keys have a shorter lifespan. However, in order to reduce the risk of compromise both the key and the data, reliable symmetric communication systems are limiting the amount of data that is protected by a symmetric key. This poses important challenges in the management of these keys. The primary considerations encompassing symmetric KM includes key generation, key distribution, and key agility (i.e., the ability to change keys quickly when needed to protect different data).

Key Generation

The protection of the symmetric key is mandatory in this type of system and is the greatest challenge in symmetric key system management. The generation of a symmetric key can essentially be accomplished in two ways: (1) locally, on the end device platform (e.g., implementing key agreement), or (2) remotely, at a single facility not physically attached to the end device platform.

In the local generation scenario, a DH key agreement process provides a good example for this style of generation (implementation of something similar can be found for the IEEE 802.11 Standard). A simplistic description of DH involves two parties that use private information known by each party and public information known by both parties to compute a symmetric key shared between the two parties. In this case, no outside influences are involved in key generation, only information known by the parties that wish to communicate is used. However, local key generation is not always possible, due to end device limitations, such as limited processor power and local memory constraints for storage of the values needed for computation.

In the remote generation scenario, the symmetric key is generated by one entity (e.g., a key server) and transported to one or more other entities (e.g., the end points that will use the key). Placement of the symmetric key into the end points can be accomplished using multiple methods that include pre-placed keys or electronically distributed keys. In the pre-placed method, the symmetric key is manually entered (i.e., physically loaded) into the key consuming device prior to the use of the key. This can be achieved at the factory or done when the device is deployed into the field. Electronically distributed keys need to be protected as they transit across the network to their destination. This can be achieved by encrypting the symmetric key so that only the end device can decrypt the key.

However the remote generation scenario has more complexity associated with it because of distribution and trust risks. However, in the remote generation and distribution model, the

concept of Perfect Forward Secrecy (PFS) can be managed for a large population of devices. PFS is dependent on the use of an ephemeral key, such that no previously used key is reused. In remote or central key generation and distribution models, PFS can be ensured because the key generation node can keep track of all previously used keys.

Key Distribution

An important area of consideration relative to key distribution is the method to establish the trust relationship between the end device and a key loader. In today typical practices, it is necessary for the system managers to determine how this trust relationship is established. Establishing the trust relationship should be based on a number of factors that focus on risks to the physical transport of the keys to the end point.

In the electronic distribution scenario where the symmetric key is generated by a key server, which is placed external to the key consumer (i.e., the end point), the trust problem and the protection of the symmetric key in transit are mandatory considerations to the successful implementation of this solution. To mitigate the risk of disclosure, the key should be transported to the key consumer by wrapping (i.e., encrypting) the plaintext symmetric key, used for data protection, with a key encryption key (KEK). An individual KEK can be created by using the public key issued to the key consumer device. In this way the symmetric key can be wrapped by the key generation server using the end devices public key and only unwrapped by the end devices private key. By using this method only the key consumer is able to extract the symmetric key, because only the key consumer has the associated private key, which of course remains protected on the key consumer's platform memory.

Key Agility

The final topic to discuss in symmetric KM is related to key agility. Key agility becomes critical when a compromise takes place as well as in normal operational mode.

In the case of a key compromise, key agility allows the key consumer to change to another key so that uninterrupted communication between end points can continue. However, key agility must be part of the overall key management function of planning and distribution. The key distribution package must also contain enough key material to provide operational keys to support a compromise recovery. In the scenario where a compromise takes place, the compromise recovery key would be used, which would allow the key distribution point enough time to generate a new key package for distribution. Optionally, the compromise recovery key may not be part of the same numerical branch as the previously used key to prevent a follow-on compromise where the attacker was able to determine the roll over key, based on the previously compromised key.

In the normal operational scenario, where the key's lifetime comes to a natural end, the next key needs to be available to all key consumers within the same crypto group prior to usage in

order to ensure continuous communications. It should be noted that key roll over and the roll over strategy is highly dependent on how the system uses the symmetric key and the frequency of communications using that key. Thus, in a scenario where communications is infrequent and the key distribution channel is secure, only a single key might be distributed to the consumer devices.

5.1.7 Asymmetric KM

Communication systems that use asymmetric primitive to provide security services, normally implement distinct encryption and decryption functions with different keys as inputs data. The two key values are typically generated together, and often one of the two keys is published (public key) while the other is kept secret (private key). Unlike in symmetric encryption schemes, it is computationally infeasible to deduce the private key from the public key. Asymmetric primitives allow secure communication although one of the two key values was published. Therefore, they are often referred to as public-key cryptosystems. This characteristic facilities key establishment and management over insecure channel compared to systems using only symmetric cryptographic methods. The major disadvantage of asymmetric cryptography is the high computational effort that is necessary to compute the algorithms. On small microcontrollers (e.g. 16-bit word size) some of the algorithms can lead to an unacceptably long computation time or require too much memory. Furthermore, the energy consumption due to the higher computation effort is often not negligible. As aforementioned, the size of the key to achieve the same level of security is furthermore bigger than for symmetric crypto algorithms. This increased key length often results in higher communication effort (end therefore higher energy consumption for transmitting data) to perform cryptographic protocols with asymmetric primitives.

However, the use of PKC alone is not enough for protecting a M2M network: it is necessary to have a Public Key Infrastructure that can be able to establish a trusted identity, amongst other things. The major components of a PKI, according to the several models, are the following: the clients, which are the users of a PKI certificate; the Certification Authority (CA), which establishes identities and creates digital certificates; the Registration Authority (RA), which is responsible for the registration and initial authentication of the clients; and the Repository, which stores the certificates and the Certification Revocation Lists (CRLs). In order to provide the services of a PKI, such as initialization and certification, these components and their functionality must be mapped to the entities of a M2M wireless network.

Notably, certificates are issued with a validity period. The validity period is defined in the same certificate with two fields called “notBefore” and “notAfter”. The notAfter field is often referred to as the expiration date of the certificate. It is important to consider certificates as valid only if they are being used during the validity period.

If it is determined that a certificate has been issued to an entity that is no longer trustworthy (for example the certification was issued to a device that was lost, stolen, or sent to repair), the

certificate can be revoked. Certificate Revocation Lists are used to store the certificate serial number and revocation date for all revoked certificates. An entity that bases its actions on the information in a certificate is called a Relying Party (RP). To determine if the RP can accept the certificate, the RP needs to check the following criteria, at a minimum:

- The certificate was issued by a trusted CA, in the best situation the final gateway. (This may require the device to provide on the RP to obtain a chain of certificates back to the RP's trust anchor; trust root)
- The certificates being validated (including any necessary chain back to the RP's trust anchor) are being used between the notBefore and notAfter dates.
- The certificates are not in the CRL.
- Other steps may be required, depending on the RP's local policy, such as verifying that the distinguished name of the certificate subject or the certificate policy fields are appropriate for the given application for which the certificate is being used.

Certificate Issue

As mentioned above, when a certificate subject is no longer trustworthy or the private key has been compromised, the certificate is placed into a CRL. This allows RPs to check the CRL to determine a certificate's validity status by obtaining a recent copy of the CRL and determining whether or not the certificate is listed. Over time, a CRL can become very large as more and more certificates are added to the revocation list, (e.g., devices are replaced and no longer needed, but the certificate has not expired). To prevent the CRL from growing too large, PKI administrators determine an appropriate length of time for the validity period of the certificates being issued. When a previously revoked certificate has expired, it need no longer be kept on the CRL, because an RP will see that the certificate has expired and would not need to further check the CRL. Administrators must consider the balance between issuing certificates with short validity periods and more operational overhead, but with more manageably-sized CRLs, against issuing certificates with longer validity periods and lower operational overhead, but with potentially large and unwieldy CRLs.

When certificates are issued to devices whose "employment" status or level of responsibility may change every few years, it would be appropriate to issue certificates with relatively short lifetimes, such as a year or two. In this way, if an device's status changes and it becomes necessary to revoke his/her certificate, then this certificate would only need to be maintained on the CRL until the certificate expiration date. In this way (by issuing relatively short life certificates), the CRLs can be kept to a reasonable size. When certificates are issued to devices that are expected to last for many years, and these devices are housed in a secure environment, it may not be necessary to issue a certificate with such short validity periods, as the likelihood

of ever needing to revoke a certificate is low. Therefore, the CRLs would not be expected to be very large. When a M2M network RP receives a certificate from an entity, and the certificate has expired, it has to reject the message.

Taking into account that M2M devices will be deployed with the intent to keep them operational for many years (probably in the neighborhood of 10 to 15 years) and replacing these devices should not occur very often, the certificate lifetime's becomes a vital factor to not induce unnecessary energy consumption. Of course, there will be unplanned defects that will cause devices to be replaced from time to time. The certificates of these defective devices will need to be listed on the CRL when the devices are removed from service, unless their keys can be guaranteed to be securely destroyed. In order to avoid the unlimited growth of CRLs, it would be prudent to issue device certificates with an appropriate lifetime. For devices expected to last 20 years, which are housed in secure facilities, and have a low mean-time-before-failure (MTBF), a 10-year certificate may be appropriate. This means that when a device having a certificate of this length is installed in the system and subsequently fails, it may need to be on a CRL for up to ten years.

If a good device never gets a new certificate before its certificate expires, the device will no longer be able to communicate in the system. To avoid this, the device could be provisioned with a "renewed" certificate quite some time before its current certificate expires. For example, the device may be provisioned with a new certificate a year before its current certificate expires. If the renewal attempt failed for any reason, the device would have a whole year to retry to obtain a new certificate. It is therefore easy to see that the probability of a critical device not being able to participate in the system because of an expired certificate can be made as low as desirable by provisioning the device with a new certificate sufficiently before the expiration of the old certificate.

It is worth mentioning that because of the size and scale of the typical M2M networks, other techniques may be needed to keep CRLs from growing excessively. These would include the partitioning of CRLs into a number of smaller CRLs by "scoping" CRLs, based on specific parameters, such as the devices' location in the network, the type of device, or the year in which the certificate was issued. Methods for supporting such partitioning are documented in [Haas et al., 2011]. Clearly with a system as large as M2M networks, multiple methods of limiting the size of CRLs will be required, but only with the use of reasonable expiration dates can CRLs be kept from growing without limit.

5.1.8 State-of-Art

In this section we are now focus on describing several KM mechanisms that are implemented in Standard (802.11) and industrial solutions (e.g., Zigbee, WirelessHART, etc.). All of them use wireless channel to exchange information thus they can be used as inspiration for future implementations on M2M networks. This section represents the state-of-art for KM solutions in this thesis.

802.11/WLAN

In 1997 the IEEE adopted IEEE Standard 802.11-1997, the first wireless LAN (WLAN) standard [802.X, 2010]. This technology is promoted from Wi-Fi Alliance that is a trade association in charge of certifies products if they conform to certain standards of interpretability.

Wireless network security protocols have advanced tremendously since the publication of the original 802.11 standard in 1997, with new encryption and authentication algorithms such as WEP and WPA. Many consumer wireless routers now come equipped with advanced security features.

WPA is today the security standard in wireless networking that is rapidly replacing the older WEP standard. WPA and its younger sibling WPA2 are newer standards based on the IEEE 802.11i ratified amendment set out to improve some of the disadvantages of WEP.

WPA builds upon WEP, making it more secure by adding extra security algorithms and mechanisms to fight intrusion. Perhaps the most important improvement over WEP is a dynamic security key exchange mechanism and much more improved authentication and encryption mechanisms. In this section we will focus on the key exchange protocol used with WAP/WPA2, because it is an interesting re-keying mechanism for KM in low-power networks.

While WEP uses the same static security key for both encryption and decryption of all communication (the key never expires), WPA implements a mechanism involving a number of security keys. This is done through so-called Temporal Key Integrity Protocol (the TKIP mechanism shares a starting key between devices, but each device then changes its encryption key for the ongoing communication). This is a revolutionary improvement because even if the attacker obtains one security key, he will not be able to use it for long. This system changes the security key used for data transmission every specified amount of time to prevent cracking attempts (re-keying mechanism).

First, initial authentication is done using the Pre-Shared Key set in the wireless configuration (the key that is set at the access point and then pre-configured to the nodes). So far, once the initial authentication is completed, then another so-called Pair-wise Master Key (PMK) is generated which is bound to the particular session between the access point and the node.

The Master Key is then further split into so-called Group Transient Key (GTK) which secures multi-cast and broadcast messages sent by the access point to the clients and to another security key called Pair-wise Transient Key (PTK) which secures the unicast messages sent from wireless clients to the access point. The PTK is generated by concatenating the following attributes: PMK, Access Point nonce (ANonce), Node nonce (NNonce), Access point link layer address, and Node link layer address. The product is then put through a cryptographic hash function.

Some wireless routers provide a function allowing the administrator to control how often the Group Transient Key is changed by the access point. As you can see, this mechanism is principally quite hard to crack because even if the attacker captures some security key from the data flow, it is limited to a single session and can even expire within that session as well.

This method securing the network using the optional Pre-Shared Key (PSK) authentication and it is designed for home users without an authentication server. Resuming, the key exchange protocol in 802.11 with WPA2 consists in four-way handshake messages and provides a secure re-keying mechanism in order to guarantee a secure Pair-wise Transport key to the every station (node):

1. The Access Point sends a nonce-value to the station (ANonce). The node now has all the attributes to construct the PTK.

2. The node sends its own nonce-value (NNonce) to the Access Point together with a MAC, including authentication, which is really a Message Authentication Code.

3. The Access Point sends the GTK and a sequence number together with another MAC. This sequence number will be used in the next multi-cast or broadcast frame, so that the receiving node can perform basic replay detection.

4. The node sends a confirmation to the Access Point.

Zigbee (standard security level)

ZigBee standard security mode [Zigbee, 2010], which is based on a 128-bit AES algorithm, adds to the security model provided by IEEE 802.15.4, three security services methods: frame protection, key establishment and transport and also device management; In our study we will focus on frame protection and key establish and transport. The message frame content generated at the Network (NWK) layer and higher is encrypted using 128-bit AES-based encryption. The NWK payload of the frame is encrypted, and the NWK header and payload are integrity-protected with a 32-bit MAC. Two types of frame protection can be applied:

- **Network-level security:** This uses a “network” key which is common throughout the network and is used to encrypt/decrypt all communications between all nodes. Network Keys are sent by the Trust Center (TC) either encrypted or un-encrypted mode. These keys, if security is enabled, guarantee the validity of a message in the network due to only valid nodes are owners of the secrets (keys).

- **Application-level security:** This uses an application ”link“ key which is used (in addition to the network key) to encrypt/decrypt communications between a particular pair of nodes, each pair of nodes has a unique link key. Link Keys secure unicast messages between two devices at the Application Layer, in this way these keys guarantee privacy between a pair of node.

Once network-level security has been set up, application-level security can be set up for an individual pair of nodes. Application-level security is used when the communications between

the two nodes must remain private from the rest of the network. In order to set up application-level security between two nodes, a specific function must be called from one of the nodes to request an application link key from the TC. The TC responds to this request by sending the same application link key to both nodes.

Zigbee PRO (standard and high security level)

The ZigBee PRO [Zigbee, 2010] feature set has two security modes (Standard Security, High Security), and three types of security keys (Network Key, Link Key, Master Key), whereas the ZigBee feature set has only Standard Security mode and two types of security keys (Network Key, Application Link Key). Both feature sets use symmetric encryption, the Advanced Encryption Standard (AES-128), and apply authentication/encryption on Network and Application layers; as in the last section we presented standard security in Zigbee, now we focus on high security in Zigbee PRO. In this kind of mode there are three keys:

- **Network Key:** it is shared amongst all devices in a network and it is used to secure broadcast communications in a network.
- **Link Key:** it is shared by two ZigBee devices and it is used to secure unicast communication between application peer entities.
- **Master Key:** it is not used to encrypt frames while it is used as an initial shared secret between two devices when they perform the Key Establishment Procedure (SKKE or PKKE) to generate Link Keys. Thus with these kinds of keys, a couple of node are able to deal with a secret key (Link key) without the supervision of a Trust Center and consequently saving energy.

However the security keys can be acquired in different ways depending on their types: Key Transport is the case that the TC of the network sends the key to the device and Key Establishment is the method that is used to establish a pair-wise key (Link Key) between two devices. Note that for this method, a pre-shared key (Master Key) is required between two devices. Pre-installation is the case that the device acquires the key before joining the network.

The real innovation in Zigbee PRO High security mode is the Key Establish method which can works with symmetric cryptography (SKKE) or with asymmetric cryptography (PKKE). The figure above represents the different mechanisms implemented to provide the link key between couples of node. The first method on the left is used in standard security mode while the second is used in high security mode.

Key Establishment Procedure with symmetric crypto (SKKE)

In the SKKE protocol, an initiator node establishes a link key with a responder device using a shared secret (master key). Master key may either be pre-installed or transported from the TC

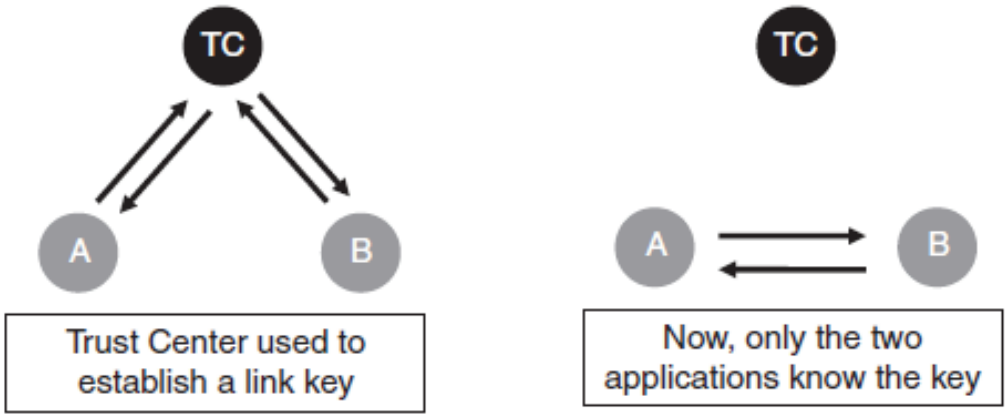

Figure 5.1: Link Key delivery methods.

encrypting with the link key. In the first messages, the devices exchange their 16-byte challenges and in the last messages, the devices exchange the data they have computed using the challenges and the device identities (note that a device identity is the unique 64-bit device address). Note that in this method SKKE sues a key derivation function that takes two parameters: the shared secret bit string, and the length of the keying data to be generated.

However, symmetric cryptography lacks (1) session key distribution without the intervention of a TC, (2) digital signature and the non-repudiation property. Public-key cryptography could alleviate these shortcomings but one needs to consider the trade-off between the performance and the security. The computational complexity of symmetric cryptography is much less, whereas KM (the process of selecting, distributing and storing keys) is complex and relatively insecure compared to the PK-cryptography.

Key Establishment Procedure with asymmetric crypto (PKKE)

In order to transport static public keys signed by a CA, a protocol called Certificate-Based Key Establishment (CBKE) is implemented with PKKE. With this protocol the Link key is providing to a couple of nodes with the use of a key agreement (ECMQV) and a certificate generation method (ECQV) based in the Elliptic Curve Cryptography.

The static data, which is a combination of device address and device static public key, and the ephemeral data are important input information to solve this process. With this protocol, in the first messages, the devices exchange static (which is a combination of device address and device static public key) and ephemeral data, instead, in the last messages, the devices exchange the data they have computed using the static and ephemeral data. As for SKKE, after verifying that the devices have received the correct values, they use a new value, which is computed from both

ends-devices, as the link key. This operation allows to start private and secure communications between these two devices. Note that in this mechanism the two ends have to implement a key generating function with ECMQV key bit stream, and the same key derivation function as in SKKE.

WirelessHART

Security is a core element of the WirelessHART [WirelessHart, 2010] design and it can be broken down into two main categories which can be termed Data Protection and Network Protection. Data Protection, or Confidentiality, deals with maintaining the Confidentiality and Integrity of the information being passed over the network, while Network Protection, or Availability, deals with maintaining the functionality of the network in the face of internal and/or external attacks (intentional or unintentional).

The WirelessHART Security Manager is responsible for the generation, storage, and management of the keys that are used for device authentication and encryption of data. In order to provide authentication WirelessHart provides the MAC that is generated with CCM* (counter with CBC-MAC) using the AES-128 algorithm. For its generation is necessary to include a 128-bit key, a nonce of 13 bytes and the message header without encryption. Public, Join, Network and Session Keys must be provided from the WirelessHART Network Manager:

- **Network Keys:** which are shared by all network devices and used by existing devices in the network to generate link layer MACs.
- **Public Keys:** which are used to generate MACs on the link layer layer by the joining devices (network-wise keys).
- **Join Keys:** that are unique to each network device and is used during the joining process to authenticate the joining device with the network manager (pair-wise keys).
- **Session Keys:** that are generated by the network manager and are unique for each end-to-end connection between two network devices. It provides end-to-end confidentiality and data integrity.

The key distribution protocol used in WirelessHart is the same implemented in Zigbee standard security mode: the Network Manager manages the keys and send them after have received a node's request. However, the problems that have been identified in WirelessHART are the same of which were presented in Zigbee standard security mode section.

Final Table

In this sub-section is shown the summary table, Table 5.1, concerning the main features for the presented KM schemes. These schemes are an important started point to come out with

an innovative mechanism for new M2M networks; they are solutions implemented on scenario where wireless channel is used as transmission medium.

Table 5.1: KM characteristics.

Standard	**Drawbacks**	**Solutions**	**Comments**
802.11/ WLAN	1-Network-wise for authentication. 2-Trust Center for re-keying and key exchange. 3-No presence of link key.	1-Pair-wise key for authentication. 2-Re-keying and key exchange managed from the nodes. 3-Add pair-wise keys.	The idea of exchange ephemeral data to deal with a common key between couples of nodes is interesting.
Zigbee standard security	1-Network-wise key for authentication. 2-Trust Center for re-keying and key exchange.	1-Pair-wise key for authentication. 2-Re-keying and key exchange managed from the nodes.	Zigbee standard security mode introduces the application key between every couple of nodes but no provides efficient re-keying, key distribution and authentication methods.
Zigbee high security	No problem identified	X	Interesting starting point to come out with a solution for M2M scenario.
Wireless HART	1-Join process is expensive in terms of energy consumption. 2-Trust Center manages the key distribution for session keys. 3-Network-wise key to provide authentication.	1-To pre-configure the Session and the Network keys. 2-Key exchange manages form the nodes. 3-Pair-wise key for authentication.	Same problem of Zigbee standard security mode and in addition expensive process to join new nodes.

5.2 Proposed Solution

In this Section, we present the proposed solutions regarding key generation and key updating.

5.2.1 Key Generation

This method was studied to reach the needed security requirements that are necessary to secure a typical M2M network. However, in order to be more practical, we would like to adapt it for the aforementioned solutions. Taking into account the implementation of the previous mechanism, the authentication preamble presented in Sections 4.2 and 4.3, every node in our proposal needs 2 pairwise keys with each of its one-hop neighbors, and, as later explained in Section 5.2.2, another pairwise key in order to update the previous keys. Summarizing, considering two neighbors A and B, the three pairwise keys are:

- $K_{A,B}$, that is a short-term key used to generate the APs of the data exchanged between A and B during normal operation;
- $K^r_{A,B}$, that is a short-term key used to generate the APs of the data exchanged between A and B during the recovery process; and
- $K^u_{A,B}$, that is a long-term key that allows to securely update $K_{A,B}$ and $K^r_{A,B}$ when their cryptoperiod is about to expire.

The above-mentioned pairwise keys could be previously stored in the devices assuming that there is a prior complete knowledge of the network deployment. That is to say that one knows the location of every node, and its neighbors alike, prior deployment. This is however a strong assumption that is probably not suiting many deployments' requirements. As result, we opt to provide a method to generate the necessary pairwise keys upon or post deployment.

However, there are many approaches in the literature dealing with post-deployment pairwise agreements, most of them based on shared secrets, a trusted third party or asymmetric relationships [Loree et al., 2009, Gagneja, 2012]. Aiming at minimizing the cost of these agreements, we propose to use a network-wide pre-shared master key (MK) that allows to securely agree the necessary pairwise keys, and that it must be stored in every node prior deployment.

As shown in Figure 5.2, in this proposal every pairwise key is obtained by means of a hash-based message authentication code (HMAC) of the IDs of both peers. The chosen HMAC depends on the security policy of the specific implementation, but involves at least the use of a cryptographic hash function, e.g. SHA-2, and a secret key, which in our proposal is the MK. In order to allow different type of keys, besides the node IDs, there is another input to the hash function that accounts for a given type of key. This input could be, in the simplest implementation, a 2-bit registry that is:

- "00" for the short-term pairwise keys used for the AP generation during normal operation;

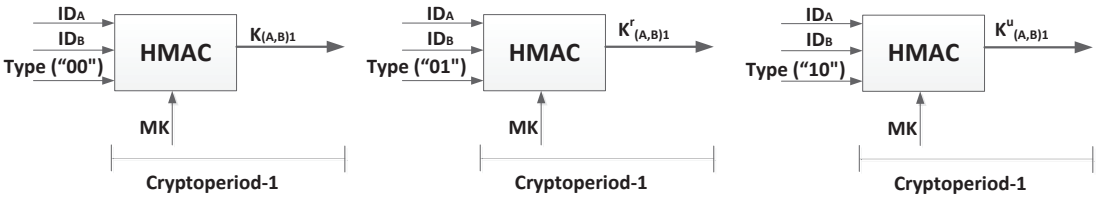

Figure 5.2: Key generation process for $K_{A,B}$, $K^r_{A,B}$ and $K^u_{A,B}$.

- "01" for the short-term pairwise keys used for the AP generation during the synchronization recovery process;
- "10" for long-term pairwise keys used for updating the above short-term keys.
- "11" reserved

After deployment, every network node generates the necessary pairwise keys with every in-range or neighbor node. The necessary unique IDs, e.g. the unique physical address of the network devices, are broadcasted in clear, and the generation process is not exposed since its security relies on the shared MK. After the pairwise key generation is over, the MK can be safely deleted as every node stores three pairwise keys with each one of its neighbors. We underline that this process is not avoiding the inclusion of new members to the network. Indeed, in this case, new nodes are able to generate the short-term pairwise keys and the long-term key, by getting the numbers of key-generation processes already computed by their neighbors for each key. However, in this solution proposal, we take into account only static devices that thus should not be able to change their neighbors during the entire network's lifetime.

5.2.2 Key Updating

As cryptograms are exposed to external entities, the security of the cryptographic keys in use becomes weaker. Because of that, keys are usually authorized for use by legitimate entities only during a given time slot or cryptoperiod. As the nature of the different cryptographic protocols is profoundly different, several limitations must be considered to select the proper cryptoperiods for a given application; sometimes cryptoperiods are defined by an arbitrary time period and other times they are defined by the maximum amount of data protected by the key. On the one hand, Section 5.3 expands on this topic with an implementation example on top of the IEEE 802.15.4e standard. On the other hand, this section covers how to securely update the keying material (the pairwise keys) when a given cryptoperiod is over.

The overall updating process is fairly simple. As shown in Figure 5.3, upon expiration of a given key's cyrptoperiod a new short-term key replaces the previous one. This new key, that should be previously computed, is obtained as the result of a HMAC involving the previous one as input and the long-term pairwise key as the secret key. That is to say, between two neighbors

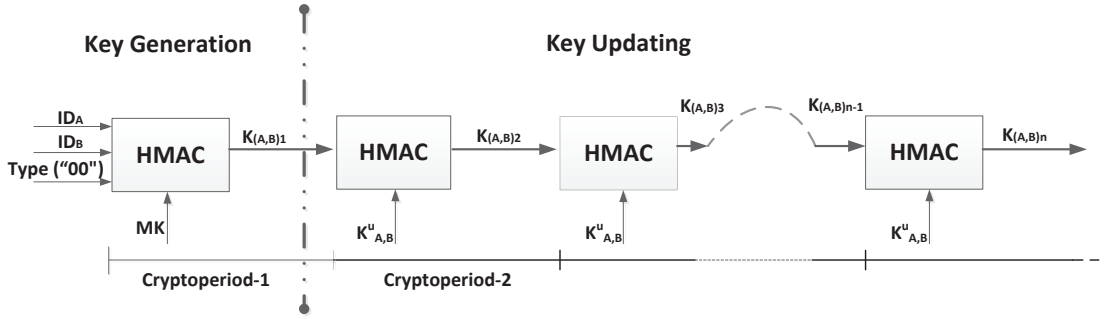

Figure 5.3: Key generation and updating processes.

A and B, the new $K_{A,B}$ is a HMAC of the old $K_{A,B}$ with secret key $K^u_{A,B}$ and the new $K^r_{A,B}$ is a HMAC of the old $K^r_{A,B}$ with secret key $K^u_{A,B}$.

5.3 Protocol Analysis & Optimization

In this Section we come out with the key lifetime for each one of the previously presented/generated key. Key lifetime is the interval in which a key is authorized to provide specific security services. In general, a key lifetime is related to a cryptoperiod. As the nature of cryptographic protocols is profoundly different; different limitations should be take into account to select the proper cryptoperiods for a given application. Sometimes cryptoperiods are defined by an arbitrary time period and other times they are defined by maximum amount of data protected by the key.

In our proposal the keying material can be classified into three types of keys:

- **Master key.** The preloaded MK that must be safely deleted when the key generation process is over.
- **Long-term pairwise keys.** These are the pairwise keys $K^r_{A,B}$ that are used during the short-term keys updating process and that are not updated during the whole lifetime of the node.
- **Short-term keys.** These are the pairwise keys $K_{A,B}$ that are used to generate the APs during both normal operation and the synchronization recovery process.

These keys do not need to be updated since they are very sparingly used and the cryptograms obtained by these keys are not exposed to an attacker. As the above keys are used for different intents, they have different cryptoperiods. The MK is used only during the initial pairwise keys generation process and then it is safely deleted. Consequently, the MK's cryptoperiod is by far longer than its lifetime. As long as the long-term pairwise keys are very sparingly used (just for key updating) and the cryptograms obtained by these keys are not exposed to an attacker, we

can assume that their cryptoperiod is long enough to avoid updating them during the node life. However, the cryptoperiod for the short-term keys should be defined since they are widely used and the obtained cryptograms (the APs) are exposed to the attacker.

Generally speaking the AP is exposed to two external threats: i) the maximum number of times that a key should be reused to be safe against cryptanalysis, and ii) attacks that find a collision and thus bypass the PHY-authentication.

In the IEEE 802.15.4 standard, the preferred protocol of encryption is AES with 128-bit key length. In this case, the NIST recommendation [NIST, 2005] for 128-bit keys with AES is to limit the use of the key to 2^{48} times. With that said, every device will compute a short-term key update process after transmitting 2^{48} packets with AP generated using that key. As a result, assuming a constant packet rate r between neighbors, the cryptoperiod t is $\frac{2^{48}}{r}$ which would be hardly achieved, close to 9000 years for a rate of 1000 packets/second. With such a result, we can consider that the former is not a real threat.

In the case of the latter threat, an attacker could find a collision after an average $2^{\frac{32}{2}}$ of tries. The only effect of the attack is to bypass once between thousands of time the lightweight PHY-authentication. As a result for this one time, the whole packet will be received, processed and probably rejected by the security mechanisms implemented at upper-layers. In any case, in order to minimize the threat of collisions, we recommend updating short-term keys after 2^{16} uses, that is to say that the cryptoperiod would be in the order of hours depending on the network rate.

Chapter 6

Security working group of the WAVE2M Community

"Intelligence without ambition is a bird without wings"
Salvador Dalí.

Contents

Community has quickly realized that a plethora of proprietary technologies is counter-productive to the vision of a quickly scaling and reliable M2M networks. The emergence of Standards Developing Organizations, in the area of M2M applications hence is a natural development. The aim of said bodies is to create a common understanding of the architecture, protocols and functionality to support operations and communications. In this chapter we present WAVE2M Standard that aims to define an appropriate protocol suite to allow reliable communication for M2M networks.

6.1 WAVE2M Standard Alliance

WAVE2M [WAVE2M Community, 2012] is one of the SDO initiative that is working since 2004 to develop an appropriate communication suite for low-power networks. Driven by metering giants, such as Elster, communications giants, such as Orange France Telecom, and utilities, such as Veolia, the members are involved in defining a technology roadmap with viable technologies tailored not only to electrical but also gas and water grids, with main focus on the meter and sub-meter end points. The standard is constantly evolving and, thanks to its working groups (WGs), enjoys today a very advanced ultra low-power PHY layer (which is also being considered within IEEE 802.15.4k), an advanced asynchronous and shortly-to-be-released synchronous (similar to IEEE 802.15.4e PA) medium access control, and an IP-enabled networking layer which is using latest IETF 6LoWPAN, ROLL and other developments. Security has been recognized as a transversal issue and the respective WG hence tightly interacts with all other WGs to specify one of the most advanced security suit available to date in any standards body.

6.2 Security Working Group of WAVE2M

Thanks the era of wireless sensor networks, which had been occupied mainly academic circles for more than a decade, now, wireless channel is considered enough reliable to exchange information. Nevertheless, despite of the progress in the communication field, there are still some requirements for M2M communication that brings new challenges, especially regarding the security topics.

First of all, nodes are often “low-end" devices with constrained resources and hence the use of well-known but expensive security algorithms (e.g. asymmetric cryptography) is often not feasible and may even be questionable. Second, the availability of a powerful super-node and/or gateway which can manage security issue in a centralized manner, varies with the application space; i.e., in some applications such powerful nodes are available and in some they are not and thus rendering the network to an unattended self-organized ad-hoc entity where security management must be assumed within the group of sensors. And third, the wireless broadcast medium makes the physical layer very accessible for an attacker, which can jam, inject or modify link layer packets without difficulty and which can easily compromise and spoof a sensor node.

The role of the Security Working Group is to provide a reliable security system, for WAVE2M community to make possible the use of technologies like M2M communication, which uses wireless channels. The complications are due to the fact that new solutions are needed because of the use of wireless channels for devices with limited resources; here the traditional protocols are not feasible. To have a great impact in markets where Coronis and Wavenis are engaged, key-words for the Security Working Group are: efficiency and reliability. These two characteristics are very important because the new solutions have to optimize resources uses (efficiency) without jeopardize the networks (reliability).

6.3 WAVE2M Packet Formats Definition

Taking into account the security solutions proposed in this deliverable to address security and efficiency issues, in this section we present the packet formats at physical and link layer implemented for WAVE2M Standard Alliance.

6.3.1 Physical layer

The PPDU packet structure is illustrated in Figure 6.1. Each PPDU packet consists of the following basic components:

- SHR, which allows a receiving device to synchronize and provides the first line of defense against non-intended packets, if security is requested,
- PHR, which contains frame length, frequency hopping and encoded method information,
- PSDU, a variable length payload which carries the link layer sublayer frame.

Authentication Preamble and Synchronization	SFD	PHR1	Opt Stuff Bits	Opt Resyn	PHR2	PSDU
Variable	32 bits	32 bits	X bits	X bits	32 bits	Variable
SHR (Synchronization Header)		PHR (PHY Header)				PHY payload

Figure 6.1: PPDU format.

In this sub-section we give special attention to the security method included in the PHY layer. Regarding the position of the Authentication Preamble field, we suggest to put it immediately after the IPSW (Interleaved Preamble Synchronization Word, which will become the Authentication Preamble Synchronization Word) since this is the first modifiable field in the packet structure of the WAVE2M Community. The basic idea of this method is to save energy rejecting non-intended packets before the total reception, thus less byte are received and energy saved given the radio is duty cycled.

Regarding the old WAVE2M specification, the only field that was changed to include the Authentication Preamble, is the Interleaved Preamble Data where the 20 bits of useful data was replaced by the proposed Authentication Preamble field. This field is thus defined as follows:

- PULSE is a 16-bit alternating sequence of 1 and 0 (i.e. 1010101010101010),
- APSW clearly has been chosen for breaking both PULSE and Manchester encoding, thus resulting in a 11100010 value = 0xE2,

16 bits	8 bits	40 bits
PULSE	APSW (Authentication Preamble Synchronization Word)	Authentication Preamble
1010101010101010	11100010= 0xE2	Variable

Figure 6.2: Authentication Preamble and Synchronization 64 bits structure.

- Authentication Preamble is Manchester encoded. The useful data are thus only 20 bits. (populated with our secure PHY preamble)

We have recommended this position because it just replaces the mechanism for the identification of the emitter without security with a similar method but which a secure authentication. Also, it was located in the first bytes of each packet, allowing rejecting non-intended packets with larger savings.

6.3.2 Link layer

The link-layer frame format is different depends on the kind of packets. In Figure 6.3 is presented the unsecured packet format where security is just avoided, in Figure 6.4 is presented the secure packet format for reliable transmission and in Figure 6.5 the secure ACK structured is shown. Each link-layer frame consists of the following basic components:

- MHR, which comprises frame control, sequence number, and address information
- A link-layer payload of variable length, which contains information specific to the frame type. Acknowledgement frames contain 1 Byte of payload.
- A MFR, which contains FCS for unsecured packets and MAC for secure ones.

Unsecured packet format

Fields description:

- Frame Control: Frame information
- DA: Destination address
- SA: Source address
- MSDU: Data payload
- FCS: Frame Check Sequence

Frame Control	DA	SA	MSDU	MAC
2 Bytes	8 Bytes	8 Bytes	Variable	4 Bytes
MHR (Link-layer Header)			Link-layer Payload	MFR (Link-layer Footer)

Figure 6.3: Unsecured packet format.

This packet format is invocated from a sender when the data transmitted are not sensible information or when the sender is secure that the area is completely safe. For this reason, and in order to save energy, in this structure is avoided the “Security Header Field" and the “MIC". Instead of the MIC, we define a Frame Check Sequence field of 2 Bytes, which is an error detection and correction protocol.

Secured packet format

Frame Control	DA	SA	Security Header	MSDU	MAC
2 Bytes	8 Bytes	8 Bytes	6-14 Bytes	Variable	4 Bytes
MHR (Link-layer Header)				Link-layer Payload	MFR (Link-layer Footer)

Figure 6.4: Secured packet format.

Fields description:

- Frame Control: Frame information
- DA: Destination address
- SA: Source address
- Security Header: Security information used to encrypt and decrypt (key identifier and mode of operation).

- MSDU: Data payload
- MAC: Message Authentication Code

This packet format is invocated from a sender when the data transmitted are sensible information and the sender is secure that the area, where the transmission is in place, may be threatened by attacks. For this reason, in this structure is included the “Security Header Field" and the “MAC". The Security Header contains information used from the receiver to decrypt the packet and it can be from 6 to 14 Bytes long; depends on which techniques are implemented to pass the information. Instead, the MAC is used to guarantee Authentication and Integrity and can be 4, 8 and 16 Bytes long, depends on which security level the packet sent has.

Acknowledge

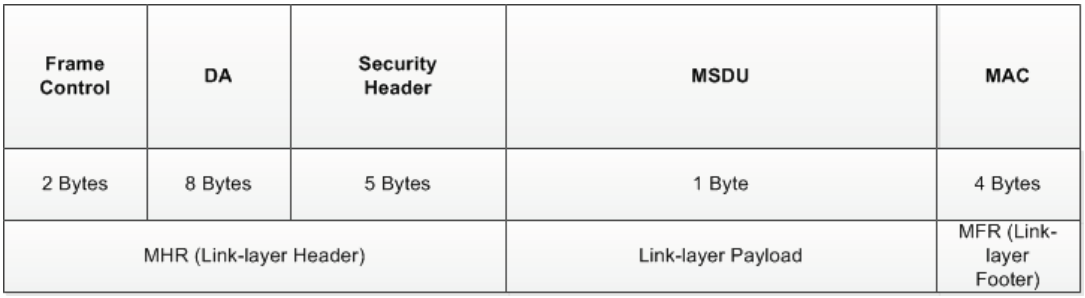

Figure 6.5: Acknowledge packet format.

Fields description:

- Frame Control: Frame information
- DA: Destination address
- Security Header: Security information used to authenticate the ACK.
- MSDU: Data payload
- MAC: Message Authentication Code

This packet format is invocated from a receiver every time that a packet is successfully received. Acknowledgement frames are numbered in coordination with the frames that have been received, and then sent to the transmitter. This allows the transmitter to become aware of which frames were actually received correctly. In this structure, in order to save energy, the Source Address field is avoided; the sender of the packet acknowledged known in advance who is the sender of

the ACK. This choice will not open any security hole because, in WAVE2M network, all the ACKs are authenticated using a pair-wise key shared between the sender and the receiver, in this way only the valid source can send an authenticated packet. The Security Header field contains information that are used to calculate the MAC and thus to provide integrity and authentication.

6.4 WAVE2M Final Security Protocol Suite

The MAC is very important in M2M networks, because with only a few bytes of overhead (4 bytes, 8 bytes or sometimes 16 bytes) we are able to verify the authenticity and the integrity of a message. As we also encrypt both message payload and MAC of the message, we are able to guarantee confidentiality, integrity and authentication that are the cornerstone in M2M security services. Notably, considering secure packet format, the MHR field is not encrypted thus this information can be read in clear if an attacker is able to sniff a packet. Regarding the Acknowledge packet, WAVE2M Community offers authentication including a MAC in every ACK packets that need security; in this way, an attacker in not able to fake it.

Finally, is this sub-section are presented the security services enabled with the packet format for secure transmission with WAVE2M Community (secure packet format):

- Authentication Preamble at physical layer
- Hop-by-hop security at link layer
- End-to-end security at application layer

Authentication Preamble (Physical layer)

With the Authentication Preamble field we add an authentication test, at physical layer, to identify those packets that are not destined to the receiving nodes or are potential attacks. In this way, if the packets are identifying as invalid, the receiving nodes will interrupt the reception after receiving only the first bytes (synchronization preamble + authentication preamble). In conclusion the proposal method permits to increase the security level and the lifetime of the network (optimization of the limited resources).

Secure Lossless Data Aggregation (Link layer)

With the secure Lossless Data Aggregation at link layer we provide a first step for the utmost power efficiency in multi-hop networks. Each node aggregates the received packets from its leaf nodes prior to forwarding the aggregated packet to its parent node. Given that one of the core requirements of some M2M applications is to be able to obtain an exact reading from each node and also to be able to uniquely associate a node to the data, lossless aggregation

must be used. Among the very few lossless techniques available, packet concatenation is a suitable solution which yields ease of use at notable performance gains. Lossless aggregation based on concatenation is particularly important when the communication infrastructure is to be shared among different inference M2M applications, such as automated electricity, gas and water metering, third party services, etc., since messages pertaining to different applications need to be securely aggregated without loss. Aggregator nodes, instead of retransmitting the raw received data, thus forward the aggregated data by combining the packets (saving headers) or even removing redundant information. Regarding security services in place at link layer, following we describe hop-by-hop and end-to-end security techniques that are used with the proposed lossless data aggregation.

Hop-by-hop security (Link layer)

End-to-end security is checked at the final destination; however, before reaching the gateway, the packets must go through one or more wireless links that are by nature exposed to attackers. As a result, if no security is provided in order to restrict the access to the media, only the destination point will be able to detect altered, dropped or fake packets. This fact exposes the network to exhaustion attacks since those packets will waste precious energy at the intermediate nodes (routers). Consequently, hop-by-hop integrity, authentication should also be provided at link layer. From above reasoning, the protocol requires the use of the MACs also at link-layer layer.

End-to-end security (Application layer)

The aim of end-to-end security is to protect the data from unauthorized eavesdropping (confidentiality); to allow the destination to check the integrity of the received data; and to unequivocally identify the source of such data (authentication). End-to-end security is achieved here as follows. The metering node creates a packet with the sensed data and the headers include: the source of data (addressing field), destination (gateway), a key identifier, a security control and the data length; the data is encrypted with the key shared with the gateway; and a message authentication code is appended. Consequently, end-to-end security is provided between the meter and the gateway (authentication and confidentiality).

Chapter 7

Security and Privacy for Future M2M Services

"Tradition is the illusion of immortality."
Woody Allen.

Contents

In Smart Cities, all structures, whether they are targeted to powering, watering, transportation, etc., are designed, constructed, and maintained making use of advanced, integrated materials, sensors, electronics, and networks which are interfaced with computerized systems comprised of databases, tracking, and decision-making algorithms. This complex system is the natural evolution of the today's urban center because current global trends in energy supply and consumption are patently unsustainable; environmentally, economically, and socially. The

scientific community believes that the future of human prosperity depends on how successfully we (people) tackle the two central energy challenges facing us today: securing the supply of reliable and affordable energy; and effecting a rapid transformation to a low-carbon, efficient and environmentally benign system of energy supply. In a few words: an energy revolution.

It would be overly simplistic, and probably a big mistake, to believe that traditional networking technologies can simply be added into a city's critical infrastructure (to make it "smarter") or that today regulations can be applied without adequate modifications (to make it "adapted") to future smart city scenario. New solutions are absolutely necessary not only to improve the quality of daily-life with innovative efficient protocols but also in terms of privacy. Privacy because the networks, especially those using wireless medium and energy-efficient communication protocols, will be exposed to a broad range of threats. Internal and external parties are not trusted and privacy will be a vital prerequisite to consumer acceptance. In addition, since assumptions and requirements for smart critical infrastructures are very different, networks for smart cities should be engineered quite differently. This also raises two challenges: an integration and a huge amount of data management problem.

In general, privacy is a controversial concept but the need to lock it down, it is more necessary now than ever. M2M smart city networks, such as metering communication systems, will be implemented in private home "violating" the most intimate spaces of citizens and if end-consumers will classify it as insecure, the M2M vision in smart city will collapse irreparably. However, focus and main contribution of this thesis is on the importance of privacy in emerging smart cities wireless M2M networks [Karnouskos et al., 2012]. In this context, we present privacy challenges, threats and requirements for M2M wireless communication systems used to enable smart grid services, and finally we introduce the most interesting mechanisms related to PET-solutions (Privacy Enhancing Technology) that should be implemented in this context.

7.1 Privacy Challenges

Important social challenges of privacy stem from the necessity to adapt M2M services to the specific characteristics of every user. A service has many configurations options, depending on user expectations and preferences; the knowledge of these preferences usually means the success or failure of a service. In order to adapt a service to the specific user's preferences, it is necessary to know them, and this is basically done based on a characterization of that specific user. Nevertheless, a complete characterization of user preferences and behavior can be considered as a personal threat, so the great societal challenge for this, and for any service requiring user characterization, is to assure user's privacy and security. Thus, in order to achieve user consent, trust in, and acceptance of smart cities, integration of security and privacy-preserving mechanisms must be a key concern of future research.

The overall priority must be to establish user confidence in the upcoming technologies, as otherwise users will hesitate to accept the services provided by smart cities. Although smart

cities are not a new technology concept by itself, but rather denote the intelligent combination of currently established systems, new challenges arise in the area of security and privacy. These challenges can be classified into two aspects following populated in details.

7.1.1 Advanced Systems Inter-operability

First, by interconnecting systems that serve totally different purposes (e.g., traffic control and energy management), and thereby creating a "system of systems", the complexity of such collaborating systems increases exponentially. As a result, the number of vulnerabilities in future advanced communication systems will be significantly higher than that of each of its subsystems. Furthermore, the pure interconnection of two systems might open new attack vectors that have not been considered before, when securing either of the individual systems.

Therefore, research into ways of handling the increasing complexity of distributed systems from the security perspective is required, which includes: cost-effective and tamper resistant smart systems or device architectures (crypto and key management for platforms with limited memory and computation); evolutionary trust models (i.e., trust is not static but dynamic, and associated values can change along time) for scalable and secure inter-system interaction; abstract and comprehensive security policy languages; self-monitoring and self-protecting systems, as well as development of (formal) methods for designing security and privacy into complex and interdependent systems; overall thread models that allow to take multiple subsystems into account.

7.1.2 M2M Networks Data Volume

Second, the number of users, and the volume and quality of collected data, will also increase with the development of M2M networks in smart cities. When personal data is collected by smart meters, smart phones, connected plug-in hybrid electric vehicles, and other types of ubiquitous sensors, privacy becomes all the more important.

The challenge is, on the one hand, in the area of identity and privacy management, where, for instance, pseudonymization must be applied throughout the whole system, in order to separate the data collected about a user (which is required in order to provide high-quality personalized services) from the user's real identity (which is required for purposes such as accounting); this includes that the usage of addressing identifiers, such as IP or Link-layer addresses, for the purpose of identification must be avoided in future systems. On the other hand, security technologies, such as advanced encryption and access control, and intelligent data aggregation techniques, must be integrated into all systems, in order to reduce the amount of personal data as far as possible, without limiting the quality of service. For future research, work towards inter-operability of different identity management systems, as well as automatic consideration of user's preferences, is required. The latter aspect goes along with the development of privacy policy languages, which allow users to express their preferences on service quality and data

minimization.

7.2 Privacy Threats

In order to properly analyze privacy issues for M2M networks, we will study the most interesting scenarios where this technology is implemented. Considering the nature of wireless medium and the fact that smart city applications may implement thousands of smart devices deployed in private spaces where they are left unattended for years of operation, the possible attacks that are able to break the privacy regulations are countless. However, in order to justify the proposed privacy requirements and solutions, which are presented in the next sections, we now sub-divide them in the following major categories:

- **Real-time surveillance.** M2M networks allows some real-time services. However, access to sensible data in real-time adds a further privacy concern to the development of security techniques that aim to avoid the creation of personal behavioral patterns. In this case, the attacker's goal is not to identify consumers behaviors, but mainly to track their behaviors in real-time.

- **Physical private space invasion.** European and Spanish regulations define home space as inviolable. However, more detailed information gathered by smart M2M meters may expose consumers to more targeted and nefarious physical private space invasions, since it may be possible to glean such information from the meter data as when residents are away from home, and even whether or not they have an electronic security system.

- **Habits identification.** High-resolution profiles can expose individual behavior patterns through the identification of each specific service requested. Considering smart grid applications for example, high-resolution profiles, regarding home power consumption, identify not only when a consumer is at home and when she is away, but further when she cooks dinner, watches TV, takes a shower.

- **Details collection.** Consumers cast off small pieces of private information in the course of routine transactions and e-commerce operations. This information, known as details or detritus, can, when aggregated, paint a detailed picture of an individual. If conclusions drawn from smart meter data were sold to third parties, it could significantly expand the set of such information. Further, as intuitions regarding the private nature of M2M data may be shifting as it is increasingly tied to notions of social responsibility, such details may be viewed as particularly sensitive.

7.3 Privacy Requirements

M2M advanced services, such as metering, pose future city network's administrators with an interconnected set of concerns, at times moving in opposite directions toward contradictory policy goals. The very characteristics that make M2M applications in-formation valuable to nascent technology industries and service providers also make it potentially damaging to consumer privacy. While consumers should be protected and informed, an overly-restrictive privacy regime could kill still nascent businesses models, such as efficient-personalized metering services.

In order to properly analyze possible PET solutions for general M2M services, we first present privacy definition and then we focus on its requirements. In [Privacy, 2010] privacy is defined as a complicate concept that may be classified in four dimensions: i) privacy of personal information, ii) privacy of the person, iii) privacy of personal behavior, and iv) privacy of personal communications. Notably, privacy of personal information means protection to any information related to an individual, privacy of the person is the right to control the integrity of one's own body, privacy of personal behavior is the right of individuals to keep any knowledge of their activities from being shared with others and, finally, privacy of personal communications is the right to communicate without undue surveillance and monitoring.

However, since M2M services are composed of two different actors, service providers and end consumers, we classify privacy-requirements considering their different points of view. On one hand, the service providers must guarantee end-consumers' privacy with appropriate privacy-preserving mechanisms. These entities are responsible for network's communication thus they must identify which threats are possible and which solutions are adequate to safeguard privacy at transmission level. On the other hand, end-consumers must realize that the appealing advantages of M2M services, such us metering personalized-services, pose privacy-risk by the submission of frequent, data-rich measurements. End-consumers in this case should be able to decide which, for what purpose and how often to send their private data to the service-providers.

Considering provider's side, communication should take into account the following issues:

- **Confidentiality**. As for all communication systems, data confidentiality must be ensured with encryption techniques. Encryption is the typical defense against eavesdropping, and the only method for preserving the confidentiality of the content of exchanged messages. Even if confidentiality is not the first requirement for smart metering networks, it is still a crucial security service because an outsider can read and induce sensitive information by correlating the results reported from multiple meters in different moments and locations. In addition, because of the in-network data aggregation operations, data of different granularity and sensitivity with respect to the user's privacy is being communicated and needs to be protected in every moment of the network's lifetime. The most reliable encryption techniques are those recommended by NIST [GCM, 2007, CCM, 2007].

- **Traffic Analysis**. Traffic analysis is a possible threat in M2M networks. In this

scenario, an adversary can be able to collect sensitive information by observing the communications' contextual data especially since they can be correlated with prior information about the people. This is the reason why confidentiality of messages' content is not enough to guarantee privacy; for example, the disclosure of both spatial and temporal data through traffic analysis, may allow tracking the relative or actual data, through correlation techniques, value which constitutes serious privacy vulnerability.

The information that can be considered sensitive for typical M2M communication systems are: i) network identity of the communicating parties; ii) transmission frequency; iii) traffic patterns; iv) size of the messages, v) location and time at which the device's measurements are being sent. Whether this information is not properly protected, the aforementioned attacks are possible.

However, in order to protect the identities of the communicating parties, multiple levels of protection should be guaranteed: sender, recipient, or mutual anonymity pertains to protecting the identity of the source, the destination, or both sides; sender and recipient unlinkability protects the relation of the nodes from third parties analysis. Techniques like time-stamping, padding, using serial numbers, or frequent re-keying can be used so that the communicated cipher texts do not reveal information through their similarity or size. However, protecting the traffic patterns within the network and the identities of the nodes is not trivial, as it requires interference with the routing protocol.

Instead, considering consumer's side, communication should take into account the following issues:

- **Statistical disclosure**. In a setting with multiple data collectors, with varying levels of trust associated to them by each user, that offer a variety of services in exchange of information, users need to able to reject, accept or negotiate the release of private data.

  Empowering the M2M service's consumer to control the level of information, privacy mainly entails two actions: first, mechanisms should be provided to allow users to define their privacy preferences and to inform them what privacy policies are being announced by data requestors. Second, having provided the users with access to information about the service provider, the service offered and the purpose of data collection, mechanisms that enable the enforcement of their preferences would allow them to control the disclosure of their personal data, its level of detail, and the pseudonym utilized.

  The issues that thus mainly need to be addressed are related to data access control, data granularity control, and protection from inference through information correlation. The issue of context awareness for the release decisions is also crucial. It has been argued, for example, that the system needs to support special exceptions for emergencies in crisis situations, where safety outweighs privacy needs [Hong and Landay, 2004]. Or alternatively, statistical information can be used as real database to solve scientific

issues, publications or provide improvements after simulations studies. Moreover, the enforcement of privacy preferences should be made transparently and with minimal user interaction demands for the system to be non-intrusive.

- **Hard privacy against consumption profiling**. In general, hard privacy means the maximum level of privacy. Taking into account M2M systems, hard privacy is required when the consumers do trust neither outsiders nor server providers or data collectors; these entities are normally considered network's insider and thus trusted parties. However, since telecommunication experience has taught that not always consumers trust on insiders, special mechanisms should be defined to enable maximum privacy protection level regarding smart meters information gathering.

  With that said, M2M network cannot be considered privacy sensitive unless two conditions are offered to every consumer. First, mechanisms should be provided for the notification of individuals within the sensing areas; for example, which data are collected, at what time and in which location. Second, data collection should be restricted to the minimum required level for the services to be provided.

  User's notice and choice mechanisms address the first issue by providing awareness of data collection and requiring user consent. A first step towards addressing the second issue is restricting the network's ability to gather data at a detail level that could compromise privacy, for example through depersonalizing the results reported by devices or through applying discrete information flows instead of continuous, when continuous data is not required by the applications. This last requirement is more proactive than the previous one, in the sense that it protects user privacy at the point of information capture, before any data release decisions are made.

7.4 Related Privacy Enhancing-Technologies

We now proceed to offer a brief overview of privacy-enhancing technologies addressing the requirements itemized in the previous section, beyond the issue of confidentiality because this aspect was deeply treated in previous sections of this thesis.

7.4.1 Anonymous-Communication Systems

Taking into account privacy risks, we stand that they cannot be solved using only security services related to encryption/confidentiality. Traffic analysis based on routing, size, timing and frequency patterns of messages between peers still can be used to supervise and overlook costumer service and behavior. Motivated by these concerns, secure and reliable M2M networks have to implement novel privacy-protecting methods to allow anonymous-communication systems.

In this sense, typical strategies to challenge traffic analysis based on message routing and size involve header-overhead encryption, message padding and splitting, and even the insertion of dummy traffic. Introducing such techniques, external attackers will find more difficulty on tracing which are the sender and the receiver of the messages and which kind of service the costumer is requesting. However, analysis of the time instants in which messages are sent, routed and received are still possible.

With this problem in mind, we have explored the literature and several solutions are emerged: the first proposed anonymous-communication system was the Chaum mix method [Venkitasubramaniam and Anantharam, 2008], essentially a trusted node placed at middleware, between the service provider and the final costumer, that is able to delay the messages sequence in order to provide the unlinkability between the incoming and outgoing messages. After this first solution, several variations were proposed with the same purpose [Serjantov et al., 2003].

However, in this section, we will present three methods to counter the network communication analysis threats and hence provide anonymous-communications:

- **Threshold pool mixes**. One of the most relevant varieties to Chaum mix method, it is a family of mixes known as threshold pool mixes. The leading idea here, is for the mix to collect a number of incoming messages, store them in the internal memory, and output some of them when the number of messages kept in its memory reaches a certain threshold. In order to eliminate any correlation between outgoing and incoming messages, the mix modifies the communication flow by resorting to two strategies, namely the delay and reordering the messages. Timely analysis are thus avoided with this technique.

- **Tor protocol (onion routing)**. This mechanism provides anonymity without introducing additional delays and it is used in Internet. In simple terms, the emitter, using kind of protocols such as Tor [Dingledine et al., 2004], encrypts the message in a multilayered fashion due to not allow the detection of the final destination to internal and external attackers. Each intermediate node is able to decrypt only a portion of the message, which allows to identify the next hop to reach the destination point, and thus avoids the traffic analysis threat.

- **Crowds protocol**. This solution is based on several levels of trust to allow network communications. An user will flip a biased coin to decide which will be the next hop to send the message. This decision is repeated from all the intermediate nodes until the destination. In the end, anonymity comes at the expense of traffic overhead and delay. This solution is used in the Crowds protocol [Reiter and Rubin, 1998]. Anonymity is possible because the internal/external parties disclaim the identity of their neighborhood; they know only their level of trust.

7.4.2 Statistical Disclosure Control

Focusing on the threats related to statistical disclosure data for M2M services, we now introduce statistical disclosure control (SDC) [Smith et al., 2012] solutions. These kinds of protocols aim to control the risk that information about specific individuals can be extracted analyzing several statistical summary results.

In general, several entities can take advantages on the disclosure of microdata set related to M2M services. Microdata set is a database table whose records carry information concerning individual respondents, either people or companies. Tropically, these tables contain three kinds of information: (i) identifier data, which aims to clearly identify a costumer, such as name and ID; (ii) semi-identifier data, which are data that can partially identify a costumer, such as ZIP number and gender; and finally (iii) confidential data, which are data related to private information, such us salary and religion. Taking into account possible privacy risks, attackers could use these tables to threaten the costumer privacy. In order to show which is the real potentiality of this attack, we now present some numbers: 87% of the population in the United States may be reidentified solely on the basis of the triple consisting of their ZIP code, gender, and date of birth, according to 1990 census data [Sweeney, 1000].

Hence, from the standpoint of privacy protection, simple removal of identifiers, for example ID number, it is in general insufficient in the publication of microdata sets for statistical studies. In order to solve this concern, we propose k-Anonymity [Samarati, 2001] techniques, which aim to perturbing numerical or categorical key attributes to preserve privacy. Naturally, this means losing some of the data usability of the statistic results comparing them with respect to the unperturbed version.

However, analyzing the proposed solutions, the mandatory requirement of k-anonymity is that each tuple of semi-identifier, also called key-attributes values, have to share at least the same k records in the dataset. This may be archived by a partial aggregation of k costumer's dataset to common values. For example, a group composed of k costumers should share the same values related to key-attributes (gender, age and ZIP code). Moreover, the confidential information, such as the monthly energy consumption, must be unchanged has it represents the interesting value for the statistical disclosure related to M2M services.

Summarizing, rather than making the original table available, we publish a k-anonymous version, avoiding the user's identifiers, partial aggregating semi-identifier records, in the sense that all key-attribute values within each group are replaced by a common representative tuple, and finally publishing the respective confidential data for each user. As a result, a record cannot be unambiguously linked to the corresponding user in any additional database assigning identifiers to key attributes. In principle, this prevents a privacy attacker from statistical disclosure threats for M2M services.

7.4.3 Hard Privacy Against Consumption Profiling

Hard privacy is a novel concern related to privacy, which emerges when the M2M service provider is a potential untrusted party. In this section, we specify data-perturbative strategies in order to counter this threat.

Recent mathematical studies of hard privacy for user profiles include [Rebollo-Monedero and Forné, 2010, Parra-Arnau et al., 2012], which investigate the submission of search queries to an information provider and the tagging of resources in the semantic Web, respectively. In a few words, the aforementioned study defines user profile as a histogram of relative frequencies of activity across predefined categories of interest, e.g., business, science, sports, etc. Then, it explores the effects of forging dummy queries and of suppressing certain tags in terms of privacy gains due to an effective perturbation of the apparent profile from the point of view of an external observer. Costs in data usability are defined as the relative amount of forged queries and suppressed tags. Finally, the cited work proceeds to analyze the maximization of the privacy gains for given usability constraints resorting to convex optimization techniques. Summarizing, this work investigates the opportunity to cover a specific profile, respectively introducing and suppressing dummy and real activities in order to change specific tags but not the final results of, for example, the sum of activities or requested services. Key point here is to analyze the trade-off between data usability after performs the data-perturbative strategies and requested level of privacy for the specific costumer.

If we take the M2M metering service for Smart Grid into account, any perturbation of the user's accurate profile of the energy consumption, computed with the goal of attaining a higher degree of privacy, must conform to any billing requirements, or at least be accompanied with any unaltered data needed for accurate billing. The perturbation can be possible modifying the punctual energy consumption, but the final monthly bill has to be unteachable.

7.5 Further Considerations

In our overview of various privacy-enhancing solutions for M2M applications/services, we have looked beyond the more traditional approaches of access control and confidentiality through encryption. In particular, when the intended recipient of sensitive information is not fully trusted (service provider) and may thus be construed as a privacy attacker as well, we are faced with a dilemma of great practical relevance. Most of the privacy mechanisms explored in this section resort to perturbing or obfuscating certain information released, to a certain degree, in lieu of simply making it either completely available or unavailable.

Chapter 8

Conclusions and Future Lines of Research

"It is much more easy to be a hero than a doctor, because hero can be sometimes, but doctor has to be always"
Andrea Bartoli.

Contents

Embedded devices in M2M networks aim to infer the sensed data and deliver this information reliably and securely to the final gateway that in general represents the service provider center or network actuator. In order to complete this process, the sensed data, the infrastructure and the communication protocols should be secured against numerous threats. In this thesis proposal, we have presented several approaches to save energy and to challenge typical energy-constrained networks attacks. These approaches provide confidentiality, authentication and availability during all the network's lifetime and permit to guarantee the end-consumer privacy while being efficient.

Notably, the first contribution of this proposal is a novel security taxonomy for M2M wireless networks. This taxonomy is important because it suits the needs of wireless M2M networks regarding their specific security issues. In this work, the analysis of the most-common architectures, valuable assets, typical threats, the counteracting services, the protocols carrying out these services and the algorithms which these protocols has given rise to the following general own assessments:

- Security is a truly global issue spanning through all protocol layers and across all network elements.

- Preventing attacks at a given OSI layer essentially eliminates threats at higher layers (given that an attacker does not gain physical access to a node).

With that said, we strongly believe that some basic security mechanisms at Link and Physical layer can be tremendously useful when protecting higher layers, notably the networking and application layers. Therefore, we focused our research onto these lowest layers.

Regarding the Link-layer, we have proposed an efficient/secure data aggregation protocol able to increase the network lifetime by reducing the resource consumption of the nodes (such as battery energy and bandwidth). This task was particularly challenging because the data aggregation designer has to provide efficiency without degrading important quality of service metrics, such as data accuracy, latency, fault-tolerance, and security. In order to achieve this trade-off, our proposal of a secure lossless data aggregation scheme not only avoids an extra cost for security, but also reduces the overall cost of the data transmission process.

The simulation analysis carried out helped to find the optimal link layer frame length in compliance with the SNR that makes our aggregation protocol a potential resources-saver with no loss of security and data granularity. Assuming that aggregation implies the transmission of longer packets through the network and that longer packets have higher probability to be retransmitted, we identified that, in the case of 802.15.4 our proposal is beneficial if the SNR is greater than 3.5dB, which fits most real scenarios. Moreover, the potential savings in terms of energy consumption with our proposed secure lossless data aggregation scheme with the same technology can reach to 27%.

Secure aggregation was shown to improve the energy consumption while preserving security at the link layer, but, we soon notified that the black hole of energy consumption was the reception of non-intended and or fake packets. Moreover, an attacker could take advantage of this risk in order to perform exhaustion attacks based on fake packet injection, which are completely received before discarded.

Aiming at reducing above mentioned costs and thus to minimize the impact of the exhaustion attacks, we decided to work on finding a way to discard non-intended and/or packets prior the reception of the entire frame. Result of this work, we proposed a lightweight verification/authentication protocol that is embedded at the PHY layer header just after the PHY preamble. The promising results of this proposal, that besides providing a basic authentication mechanism reduces to a great extent the normal operation of a wireless network, led us to patent the mechanism. The patent is now owned by France Telecom, which wants to exploit it in a near future.

However, the PHY-layer authenticator originally presented has been complemented in order to deal with supplementary risks found during its implementation. Specifically, we had to define a protocol that overcomes potential losses of synchronization due to poor channel conditions.

This protocol makes use of a sliding window of precomputed preambles and an emergency recovery process. After this proposal, we carried out a more in-depth evaluation of the authenticator that leaded to the following results: i) using typical values of the optimal window value and the energy consumption for 802.15.4e devices, this method is able to provide the minimum power consumption when the optimal window is selected at any channel condition; ii) considering exhaustion attacks, this method is able to introduce energy savings of up to $10^5\%$. Nevertheless, in real implementations cases, optimum window lengths also depend on the specific memory and CPU-capabilities of the device. As an example, for a node (a Dust Network Smart Mesh product) with 5 neighbor nodes and an optimum window length of 5 precomputed authentication preambles, a microprocessor of at least 15.25MHz (which is fairly common) is required.

Once we had defined two protocols that actually improve the efficiency, availability and overall security of wireless M2M networks, we decided to further research on the mechanisms to provide the necessary keying material for these protocols to work. Therefore, for completeness, we provided a detailed analysis of the generation process of such keys, we described how to upload these keys and we presented an insight of how to derive their cryptoperiod or key lifetime, e.g., the amount of time in which they can be regarded as safe, which is directly related to their application.

During our previous research, we started to work in parallel with the Security Working Group of the WAVE2M Community, an independent standards alliance whose participants work together to define a new wireless energy-saving communication technology. This technology was developed by Coronis for its wireless platforms, finished products and customizable vertical solutions. In conjunction with other communication technologies, such as Bluetooth and cellular networks, WAVE2M enables Coronis to innovate in their markets with wireless devices that are not only cost-effective, but also offer significant wireless range and run on a single set of batteries for many years. As members of the Security Working Group we started, among other things, to fit our previous proposal into the WAVE2M standard. Broadly speaking, we forced to WAVE2M frame formats to conform to the requirements of our proposals.

Until this point, our security research in wireless M2M networks was mainly targeted to protect services by guarantying the availability (which in the wireless world is closely related to efficiency), confidentiality and authentication of the data between a consumer and a service provider. Nevertheless, although today's connected world, with its increased availability of data, can improve customer experience, efficiency and business options in different areas, it does come with an unwanted side effect: the same data used to improve the consumers' life can also be used for purposes that consumers have not approved. This applies equally to M2M services: the data that enables the efficient running of personalized services can also be used as a source of information about the personal behavior of consumers.

From the above reasoning, for completeness, we decided to account for privacy concerns regarding the transmission of sensitive data of the end-consumers, and with such a goal, we

conducted a preliminary study regarding privacy in this scenario. In this study, we analyzed end-consumer requirements regarding privacy on M2M wireless networks and we recommended privacy-preserving solutions for efficient data management by service providers in the same context.

In conclusion, the research work in this thesis proposal has addressed security to all its extent in wireless M2M network. We had focused on the specific challenges of these networks with more emphasis on: 1) securing while making more efficient the wireless Physical and Link layers; and 2) analyzing and proposing best practices for guarantying the privacy of end-consumers. Several contributions are the result of this work: a secure lossless aggregation protocol, a PHY-layer verification/authentication protocol, the overall security of the potentially new standard WAVE2M and a code of best practices for privacy in wireless M2M networks.

All the previous contributions are efficient, reliable and feasible, as evidenced from simulation results, but they still lack of a proper management of the necessary keying material. This is the main reason why in order to conclude this thesis, in the next months, we are working on providing novel key management mechanisms for wireless M2M networks.

8.1 Future Lines of Research

After these three years spent studying security and M2M communication protocols, we recommend the following lines for future research:

- **Data Aggregation.** The main goal of data-aggregation algorithms in M2M scenario, it is to gather and aggregate data in an energy efficient manner so that network lifetime is enhanced. In this thesis we have first presented and then studied a novel data aggregation where security is provided at Link-layer. In order to optimize this protocol, future researches should include QoS-based data-aggregation parameters on our solution. Firstly, they should provide a definition of the different QoS parameters for data aggregation, such as energy efficiency, network lifetime, data latency and data quality. Then, they should compare the QoS-secure-lossless data aggregation with different secure data aggregation solutions on each QoS parameter, describe the main features of each algorithm, and highlight the trade-offs between each parameter. These results can be consequently used to define a cross-layer mechanism that can be really important to optimize the network's resources and thus provide a self-organized communication protocol. Finally, based on their own information, each device has to be able to efficiently auto-organize the data aggregation process.

- **Availability.** Taking into account the proposed Authentication Preamble at PHY-layer, more in-depth analysis is still to be done in order to assess the performance of our protocol. In particular, the four added bytes for the AP could be remove from the maximum payload length. In such a case, the are no extra cost in headers but new costs arises due to potential

need of more packets to send data and fragmentation. In any case, this specific evaluation should be done in compliance with the application using the M2M network. Moreover, apart from the theoretical analysis, this proposal still needs to be implemented and tested in a real wireless channel. A preliminary study may be carried out by means of the use of software-defined radios. These devices implement most of the radio components in software instead of in specific hardware, and therefore are highly, and often easily, reconfigurable.

- **Privacy.** In our overview of privacy challenges and solutions for M2M networks, we have looked beyond the typical approaches of access control and confidentiality through encryption. In particular, when the intended recipient of sensitive information (the service provider) is not fully trusted and may thus be construed as a privacy attacker as well, we are faced with a dilemma of great practical relevance: which is the relationship between data usability and privacy requirement. Normally, the price for high privacy it's paid in terms of several different concerns. In anonymous-communication systems, this price takes the form of message delay or traffic overhead; in SDC, the cost relates to the distortion introduced in the key attributes; and finally, hard privacy comes at the expense of accuracy in the user consumption profile. Further, both privacy and usability may be attained to various quantifiable degrees, constituting contrasting quantities in a privacy-usability trade-off. The existence of this inherent price is a strong motivation to develop adequate privacy metrics, and ultimately to design practical privacy tools achieving the maximum privacy for a desired usability level, or vice versa; ideally, we seek the optimal privacy-usability trade-off. Future research should aim at reconciling privacy and efficient M2M service management in M2M networks must therefore build upon a widely interdisciplinary variety of fields. These include not only extensive work on more traditional, cryptographically based security, but also a number of privacy-enhancing technologies intersecting with the fields of information theory and engineering optimization.

Appendices

Appendix A

Performance Over Wireless Channels

"Sapiens fingit fortunam sibi" - "The wise shapes luck himself"
Platon.

In this thesis we have dedicated part of our efforts on studying wireless channel performance. This is because in low-power networks energy-wasting have to be avoided and the most expensive component in terms of energy consumptions is the radio system. With that said, it is clear that improving the performance of the radio system implies an improvement on the performance of the wireless networks in terms of energy consumption; we cope this difficult challenge because the aim of this thesis is to provide security schemes though efficient-energy solutions.

The wireless channel is generally impaired by pathloss, shadowing and fading. The presence and interaction of these three phenomena influences the signal-to-noise (SNR) ratio and has thus a profound impact on the PHY and link layer performances of the M2M system at large.

The underlying processes of these phenomena are well known, and we will only state the distributions needed for subsequent calculus. Notably, the distribution of shadowing is typically assumed lognormal:

$$p_s(\overline{\gamma}|\mu_{\mathrm{dB}},\sigma_{\mathrm{dB}}) = \frac{1}{\sqrt{2\pi}\sigma_{\mathrm{dB}}}\frac{10}{\ln 10}\frac{1}{\overline{\gamma}}e^{-\left(\frac{10\log_{10}\overline{\gamma}-\mu_{\mathrm{dB}}}{\sqrt{2}\sigma_{\mathrm{dB}}}\right)^2}. \tag{A.1}$$

where σ_{dB} is the standard deviation (not variance) of the shadowing process in dB (not linear scale), and μ_{dB} is its mean which typically reflects the pathloss and thus relates to the SNR. Due to the fixed positions of the meters, we will subsequently assume that the shadowing process is static or very slowly varying in time but variable in space.

As for fading, indoor and many low-height transceiver measurements have shown that the Nakagami distribution is the most suitable modeling assumption, which occurs if a bunch of impinging waves phase-aligns. The PDF of the received power is Gamma distributed and given as:

$$p_\gamma(\gamma|\overline{\gamma}) = \frac{m^m(\gamma)^{m-1}}{(\overline{\gamma})^m\Gamma(m)} \exp\left(-\frac{m\gamma}{\overline{\gamma}}\right), \tag{A.2}$$

where $\Gamma(\cdot)$ is the complete Gamma function and m is the Nakagami fading factor. The spatial and temporal dynamics of fading heavily depend on the operating conditions, as will be explained below.

A.1 Physical Layer

The aim of the physical layer is to ensure that data is reliably delivered point-to-point. The core functionalities of the PHY layer of a capillary M2M radio at the transmitter are channel encoding of the bit stream and modulation of the bit stream to a symbol stream which form the over-the-air packet. At the receiver, the PHY layer is responsible for detecting the symbols, demodulating and decoding. Subsequent analysis hence pertains to average bit error rates at the input of the channel decoder and thus resulting packet error rates at the output of the decoder.

AWGN BER. The average bit error rates over non-faded additive white Gaussian noise (AWGN) channels for arbitrary quadrature amplitude modulation (QAM) and phase shift keying (PSK) is difficult to obtain in closed form. However, such a channel may occur in Smart Grid settings where the random fading realizations are assumed to be constant over the packet length or even over a very long duration. Note further that both QAM and PSK modulations are indeed used in capillary M2M solutions: QAM offers a higher spectral throughput and is used in advanced embedded radios; and PSK offers a constant envelope and thus cheaper power amplifiers and is throughout traditional embedded radios. To obtain a closed form expression for the average bit error probability (BEP), it is typically assumed that the system operates at sufficiently large signal-to-noise values and uses Gray coding, yielding one symbol error to cause approximately one bit error. The BEP is then approximated by [Proakis, 1995]:

$$\mathrm{BEP}_{\mathrm{AWGN}}(\gamma) \approx a \cdot Q\left(\sqrt{b\gamma}\right), \tag{A.3}$$

where γ is the SNR and $Q(x)$ the Q-function. The constants a and b depend on the choice of modulation and modulation order M, i.e. $a = 4/\log_2 M(1-1/\sqrt{(M)})$, $b = 3/(M-1)$ for M-QAM and $a = 2/\max(\log_2 M, 2)$, $b = 2\sin^2(\pi/M)$ for M-PSK.

Fading BER. The average BEP over Nakagami-m wireless fading channels for arbitrary QAM and PSK constellations is also difficult to obtain in closed form. However, assuming again sufficiently large SNRs and Gray coding, these can be approximated as [Shin and Lee,

2004]:

$$\begin{aligned} \mathrm{BEP}_{\mathrm{Nakagami}}(\overline{\gamma}) &\approx \alpha \cdot \left(\frac{m}{m+\beta\overline{\gamma}}\right)^{m} \times \\ &\quad {}_2F_1\left(m, \frac{1}{2}; m+1; \frac{m}{m+\beta\overline{\gamma}}\right), \end{aligned} \tag{A.4}$$

where $\overline{\gamma}$ is the average SNR and ${}_2F_1(x,y;c;u)$ is the Gauss hypergeometric function with 2 parameters of type 1 and 1 parameter of type 2 [Gradshteyn and Ryzhik, 2000, §9.14.1)]. The constants α and β depend on the choice of modulation and modulation order M, i.e. $\alpha = 2(1-1/\sqrt{(M)})/(\sqrt{\pi}\log_2 M)\cdot\Gamma(m+1/2)/\Gamma(m+1)$, $\beta = 3/2/(M-1)$ for M-QAM and $\alpha = 1/\sqrt{\pi}/\max(\log_2 M, 2)\cdot\Gamma(m+1/2)/\Gamma(m+1)$, $\beta = \sin^2(\pi/M)$ for M-PSK.

Block Channel Code. That erroneous bit stream is fed into the channel decoder, which can either be of convolutional or block type. Note that both channel coding methods are indeed in use today by advanced embedded metering radios, such as provided by [Maleysson and Dugas, 2005]. The error performance of channel coders is evaluated by means of the average word error probability (WEP), which quantifies the probability that a codeword is erroneous. For block coding, this probability can be approximated by [Proakis, 1995, Mary, 2008]:

$$\mathrm{WEP}_{\mathrm{block}}(\overline{\gamma}) \approx \sum_{j=t+1}^{J} \binom{J}{j} \mathrm{BEP}^{j} (1-\mathrm{BEP})^{J-j}, \tag{A.5}$$

where J is the word length in bits, t the number of errors which can be corrected by the code, BEP the uncoded bit error probability of the channel taking any form given above. Even though (A.5) only holds rigorously if all bit errors in the codeword are independent, [Mary, 2008] has shown that the approximation is sufficiently tight even for large modulation orders and a wide range of fading conditions. The parameters J, t are summarized in Table A.1 for some typical block codes. Several other important parameters are also stated in Table A.1: $d_{\min}$ is the minimal Hamming distance of the code and k is the uncoded number of information bits in the code word. This allows us to obtain the average packet error probability (PEP) as:

$$\mathrm{PEP}_{\mathrm{block}}(\overline{\gamma}) \approx 1 - (1 - \mathrm{WEP}_{\mathrm{block}}(\overline{\gamma}))^{N/k}, \tag{A.6}$$

where N is the total number of bits per packet and k the number of bits per block code word. This is only an approximation, albeit tight [Mary, 2008], because at higher order modulations the codewords are not strictly speaking independent.

Code	J	k	$d_{\min}$	coding rate	t
Hamming	7	4	3	0.57	1
Golay	23	12	7	0.52	3

Table A.1: Parameters for typical block codes.

Convolutional Channel Code. For convolutional codes, the average WEP can be approximated by [Manji and Mandayam, 2000, Mary, 2008]:

$$\mathrm{WEP}_{\mathrm{conv}}(\overline{\gamma}) \approx N \cdot \frac{1}{2T} a_{d_{\min}} \begin{pmatrix} d_{\min} \\ \frac{d_{\min}}{2} \end{pmatrix} \mathrm{BEP}^{\frac{d_{\min}}{2}} \tag{A.7}$$

where $a_{d_{\min}}$ is the number of paths with the minimal distance $d_{\min}$, and T the puncturing period. Above (A.7) only holds for $d_{\min}$ even which holds for typically used convolutional coder; for the general case, consult [Manji and Mandayam, 2000]. Table A.2 summarizes the parameterization of typically used convolutional codes without puncturing, i.e. $T = 1$. Finally, since convolutional decoding is typically done over the entire packet, the PEP is equal to the WEP:

$$\mathrm{PEP}_{\mathrm{conv}}(\overline{\gamma}) = \mathrm{WEP}_{\mathrm{conv}}(\overline{\gamma}). \tag{A.8}$$

K	R_c	T	d_{min}	$a_{d_{min}}$	asymptotic WEP
4	$1/2$	1	6	1	$10N\mathrm{BEP}^3$
7	$1/2$	1	10	11	$1386N\mathrm{BEP}^5$

Table A.2: Parameterizations of typical convolutional encoders.

A.2 Link Layer

The role of the link layer is to handle contention between potentially interfering links, assigning resources to a given link and also handle retransmissions in case the first transmission attempt fails. Since the amount of link layer protocols for capillary systems is extremely large [Bachir et al., 2010], we will not focus on a specific contention-based or contention-free realization but rather assume that a specific link between transmitter and receiver is already established. Furthermore, since resource allocation in the context of embedded systems essentially reduces to power control, we will also not further take this into account. Retransmissions, however, are explicitly considered as they are quintessential in ensuring reliable links. Following the notation of [Zhang, 2009], $\overline{N}_{tx}$ denotes the average number of transmissions needed to ensure a successful reception and is given by $\overline{N}_{tx} = \sum_{n=1}^{\infty} n(1-\mathrm{PEP}_{\mathrm{data}})(1-\mathrm{PEP}_{\mathrm{ack}})\left(\mathrm{PEP}_{\mathrm{data}}\mathrm{PEP}_{\mathrm{ack}}\right)^{n-1} = 1/((1-\mathrm{PEP}_{\mathrm{data}})(1-\mathrm{PEP}_{\mathrm{ack}}))$, where n is the number of transmissions, $\mathrm{PEP}_{\mathrm{data}}$ and $\mathrm{PEP}_{\mathrm{ack}}$ are the PEP of data and acknowledgement (ACK) packets, respectively. In the acknowledgment process, it is assumed that an ACK packet can be successfully transmitted in a single attempt. This is based on the fact that ACK packets are much smaller and thus much likelier to go through, and on temporal channel correlation which ensures that, if the data packet experienced a good channel, the return path experiences the same beneficial channel conditions. We can therefore assume

that $\mathrm{PEP}_{\mathrm{ack}} \approx 1$, yielding:

$$\overline{N}_{tx}(\overline{\gamma}) \approx \frac{1}{1-\mathrm{PEP}_{\mathrm{data}}(\overline{\gamma})}. \tag{A.9}$$

To characterize the average number of retransmissions, we will subsequently distinguish three wireless operating conditions, i.e.

1. fast fading where the Nakagami-*m* channel varies from symbol to symbol (ergodic over packet);

2. block fading where the channel remains constant over a packet but changes from packet to packet (ergodic over retransmission window); and

3. static fading where the channel remains constant over time but varies in space (non-ergodic).

For all three channel conditions, we assume that a static shadowing process is observed, i.e. shadowing remains constant over time but varies in space (non-ergodic). Ergodicity implies that averages can be invoked, whereas non-ergodic conditions require outages to be invoked since the averages simply have no meaning [Tse and Viswanath, 2005].

Fast Fading and Shadowing. In the first case, we observe that the BEPs are obtained from (A.4). We first deal with block coding, where we insert (A.4) into (A.5), the thus resulting expression into (A.6), and the finally resulting expression into (A.9). Unfortunately, this leads to a fairly intricate expression which does not lend itself to closed-form analysis. We thus invoke two further approximations, where the first one,

$${}_2F_1\left(m,\frac{1}{2};m+1;\frac{m}{m+\beta\overline{\gamma}}\right) \approx 1 \tag{A.10}$$

is due to the fact that the hypergeometric function converges to unity for large $\overline{\gamma}$; and the second one,

$$\mathrm{WEP}_{\mathrm{block}}(\overline{\gamma}) \quad \approx \quad c\cdot \mathrm{BEP}^{t+1}, \tag{A.11}$$

with $c = 1/((t+1)B(t+1,J-t))$ and $B(x,y)$ being the Beta function, is mainly due to the fact that the binomial sum in (A.5) can be expressed in closed form by a hypergeometric function, to which we apply the Laplace approximation and neglect again the terms which converge towards unity. The average number of transmissions under fast Nakagami-*m* fading conditions with block channel coder can thus be expressed as:

$$\overline{N}_{tx}(\overline{\gamma}) \approx \left(1-c\cdot\left(\alpha\left(\frac{m}{m+\beta\overline{\gamma}}\right)^m\right)^{t+1}\right)^{-N/k}. \tag{A.12}$$

The characterization of shadowing requires the concept of outage to be applied to the average number of retransmissions. Notably, we define the average transmission outage (ATO) as the probability that the average number of transmissions $\overline{N}_{tx}$ exceeds a given threshold $\overline{N}^*_{tx}$, i.e.

$$\text{ATO} = \text{Prob}(\overline{N}_{tx} \geq \overline{N}^*_{tx}). \tag{A.13}$$

As per our system assumptions, the the average number of transmissions is a non-increasing function in $\overline{\gamma}$, hence (A.13) can be shown to be equivalent to

$$\text{ATO}(\mu_{\text{dB}}, \sigma_{\text{dB}}) = \int_0^{\overline{\gamma}^*} p_s(\xi|\mu_{\text{dB}}, \sigma_{\text{dB}}) d\xi, \tag{A.14}$$

where $\overline{\gamma}^* = f\left(\overline{N}^*_{tx}\right)$ is the required SNR to reach the target average number of transmissions $\overline{N}^*_{tx}$. Assuming the lognormal shadowing distribution of (A.1), it can be calculated in closed form as:

$$\text{ATO}(\mu_{\text{dB}}, \sigma_{\text{dB}}) = Q\left(\frac{\mu_{\text{dB}} - 10\log_{10}\overline{\gamma}^*}{\sigma_{\text{dB}}}\right), \tag{A.15}$$

where $Q(x)$ is the Gaussian Q-Function and, following from (A.12),

$$\overline{\gamma}^* \approx \frac{m}{\beta}\left[\left(\frac{1-(\overline{N}^*_{tx})^{-\frac{k}{N}}}{c\cdot\alpha^{t+1}}\right)^{-\frac{1}{m(t+1)}} - 1\right] \tag{A.16}$$

The behavior of the outage probability ATO of the average number of transmissions $\overline{N}_{tx}$ is exemplified in Figure A.1 assuming QPSK modulation, a Nakagami fading channel with $m = 2$, various realizations of the shadowing channel $\sigma_{\text{dB}} = \{4, 8, 12\}$dB, a received SNR which yields $\mu_{\text{dB}} = 20$dB, a Golay block code with $t = 3$, $k = 12$, $J = 23$, and a packet length of $N = 8 \cdot 127$ bits. For instance, for a weak shadowing of $\sigma_{\text{dB}} = 4$dB, an average of 2 transmissions is required to guarantee .1% outage and thus 99.9% reliability of the entire smart grid region. We also observe that shadowing has a profound impact in that a stronger shadowing does not allow one to meet an outage requirement of .1%. The impact of channel coder, choice of modulation, packet length, fading strength, etc, is highly nonlinear but not depicted further here.

Above expressions allow an M2M service provider to estimate the coverage area. Notably, given a shadowing standard deviation σ_{dB} for the considered environment, a given required outage level ATO* and the tolerated average number of transmissions $\overline{N}^*_{tx}$, one can obtain the needed μ_{dB} as:

$$\mu_{\text{dB}} = \sqrt{2}\sigma_{\text{dB}}\text{erf}^{-1}(1 - \text{ATO}^*) + 10\log_{10}\overline{\gamma}^*, \tag{A.17}$$

where $\text{erf}^{-1}(x)$ is given in (A.18) on top of the next page [Wiki, 2010]. From above μ_{dB}, one can obtain the average communication distance with a given transmission power, receiver noise power density and communication bandwidth. Similarly, if the required distance is given, one

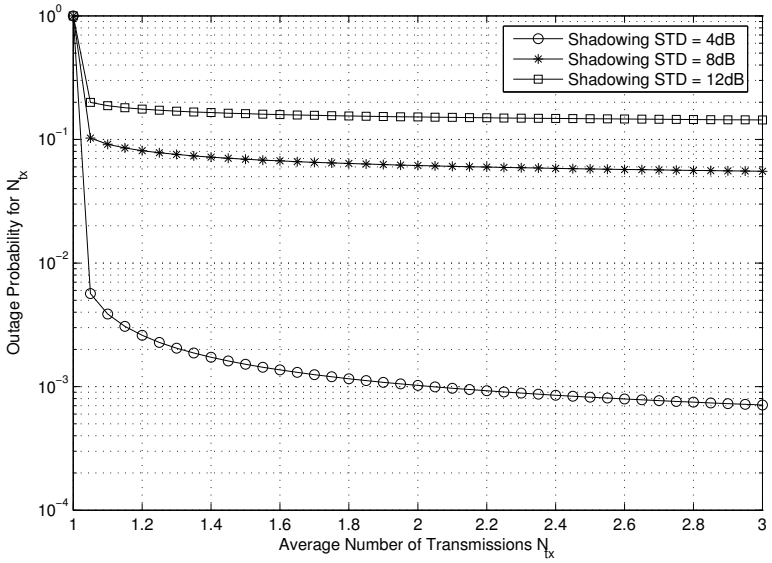

Figure A.1: Outage probability of the average number of transmissions parameterized on various shadowing channels; furthermore $\mu_{\text{dB}} = 20$, $t = 3$, $k = 12$, $J = 23$, $N = 8 \cdot 127$, $m = 2$, and QPSK modulation.

can estimate $\overline{N}_{tx}^{*}$ which impacts e.g. reservation protocols.

$$\begin{aligned} \text{erf}^{-1}(x) &\approx \frac{x}{|x|}\sqrt{\sqrt{\left(\frac{2}{\pi a}+\frac{\ln(1-x^2)}{2}\right)^2-\frac{\ln(1-x^2)}{a}}-\left(\frac{2}{\pi a}+\frac{\ln(1-x^2)}{2}\right)} \quad \text{(A.18)} \\ a &= \frac{8(\pi-3)}{3\pi(4-\pi)} \approx 0.140012. \end{aligned}$$

As for the use of convolutional channel coder, one can follow a similar procedure as above to derive the average number of transmissions under fast Nakagami-m fading conditions, i.e.

$$\overline{N}_{tx}(\overline{\gamma}) \approx \frac{1}{1-N\frac{a_{d_{\min}}}{2T}\left(\begin{array}{c} d_{\min} \\ \frac{d_{\min}}{2} \end{array}\right)\alpha^{d_{\min}/2}\cdot\left(\frac{m}{m+\beta\overline{\gamma}}\right)^{md_{\min}/2}}, \quad \text{(A.19)}$$

and its ATO threshold value as

$$\overline{\gamma}^{*} \approx \frac{m}{\beta}\left[\left(\frac{2T(1-1/\overline{N}_{tx})}{a_{d_{\min}}\binom{d_{\min}}{\frac{d_{\min}}{2}}\alpha^{d_{\min}/2}}\right)^{-2/(md_{\min})}\right], \tag{A.20}$$

from which the spatial outage and related quantities can be calculated using (A.15).

Block Fading and Shadowing. In the second case, we observe that the BEPs are obtained from (A.3). We first deal with block coding, where we now insert (A.3) into (A.11), integrate over the fading distribution, the thus resulting expression into (A.6), and the finally resulting expression into (A.9). Thereupon, the invert the expression w.r.t. $\overline{\gamma}$ to be able to obtain the ATO under shadowing conditions. Following the same procedure as above, the average number of transmissions under block Nakagami-m fading conditions with block channel coder can be derived as:

$$\overline{N}_{tx}(\overline{\gamma}) \approx \left[1 - A \cdot \frac{\Gamma\left(m-\frac{t+1}{2}\right)}{\overline{\gamma}^{m}\left(\frac{t+1}{2}b+\frac{m}{\overline{\gamma}}\right)^{m-(t+1)/2}}\right]^{-1}, \tag{A.21}$$

where $A = c \cdot d \cdot N \cdot a^{t+1}/(2\pi b)^{(t+1)/2}/k$ and $d = m^{m}/\Gamma(m)$, and its ATO threshold value as

$$\overline{\gamma}^{*} \approx \sqrt[m]{\frac{A \cdot \Gamma\left(m-\frac{t+1}{2}\right)}{\left(1-\overline{N}_{tx}^{-1}\right)\left(\frac{t+1}{2}b\right)^{m-(t+1)/2}}}, \tag{A.22}$$

from which the spatial outage can be calculated using (A.15). Similarly, the average number of transmissions under block Nakagami-m fading conditions with convolutional channel coder can be derived as

$$\overline{N}_{tx}(\overline{\gamma}) \approx \left[1 - B \cdot \frac{\Gamma\left(m-\frac{d_{\min}}{4}\right)}{\overline{\gamma}^{m}\left(\frac{d_{\min}}{4}b+\frac{m}{\overline{\gamma}}\right)^{m-d_{\min}/4}}\right]^{-1}, \tag{A.23}$$

where $B = a \cdot d \cdot N \cdot \frac{1}{2T} a_{d_{\min}}\binom{d_{\min}}{\frac{d_{\min}}{2}}/(2\pi b)^{d_{\min}/4}$, and its ATO threshold value as

$$\overline{\gamma}^{*} \approx \sqrt[m]{\frac{B \cdot \Gamma\left(m-\frac{d_{\min}}{4}\right)}{\left(1-\overline{N}_{tx}^{-1}\right)\left(\frac{d_{\min}}{4}b\right)^{m-d_{\min}/4}}}, \tag{A.24}$$

from which the spatial outage can be calculated using (A.15).

Static Fading and Shadowing. Finally, in the third case, we observe that the BEPs are obtained from (A.3). We first deal with block coding, where we insert (A.3) again into (A.11), the thus resulting expression into (A.6), and the finally resulting expression into (A.9). Thereupon, the invert the expression w.r.t. $\overline{\gamma}$ to be able to obtain the ATO under shadowing conditions. Following the same procedure as above, the average number of transmissions for a given joint Nakagami-m and shadowing realization with block channel coder can be derived as:

$$\overline{N}_{tx}(\gamma) \approx \left[1 - c \cdot Q\left(\sqrt{b\gamma}\right)\right]^{-N/k} \tag{A.25}$$

and its ATO threshold value as

$$\gamma^* \approx \frac{2}{b}\left(\mathrm{erf}^{-1}\left[1 - \sqrt[t+1]{\frac{1-\overline{N}_{tx}^{-k/N}}{c\cdot(a/2)^{t+1}}}\right]\right)^2, \tag{A.26}$$

The spatial outage cannot be calculated using (A.15) since the joint fading and shadowing process is not Gaussian anymore. To obtain the outage on the average number of transmissions, we utilize the derivation of the cumulative distribution function (CDF) of the joint process [Atapattu et al., 2010], i.e.:

$$\mathrm{ATO}(\mu_{\mathrm{dB}}, \sigma_{\mathrm{dB}}) = D\sum_{i=1}^{M}\frac{a_i}{b_i^m}\gamma(m, b_i\gamma^*), \tag{A.27}$$

where $\gamma(x,y)$ is the lower incomplete Gamma function,

$$D = \frac{1}{2}\sqrt{\pi}\sum_{i=1}^{M} w_i$$

,

$$a_i = \frac{2m^m w_i e^{-m(\sqrt{2}\sigma_{\mathrm{dB}}t_i + \mu_{\mathrm{dB}})}}{\sqrt{\pi}\Gamma(m)}$$

,

$$b_i = m e^{-(\sqrt{2}\sigma_{\mathrm{dB}}t_i + \mu_{\mathrm{dB}})}$$

, and t_i and w_i are abscissas and weight factors of the Gaussian-Hermite integration which are available for different M values in [Abramowitz and Stegun, 1965, Table (25.10)].

The average number of transmissions and its ATO threshold value for a fixed joint fading and shadowing realization with convolutional channel coder can be derived following exactly the same procedure above and thus omitted here.

A.3 Network Layer

The role of the network layer is to chose the best route for a packet to reach its destination. Different routing protocols have been put forward to date, which are generally classified into

proactive and reactive routing protocols. The former establishes an optimum routing path between any source and its sink(s), independent whether a packet needs to be sent; the latter only does so when a packet is to be sent. Proactive routing protocols are very energy consuming since routes need to be updated continuously, at the advantage of always having an up-to-date route. Reactive routing protocols may chose a suboptimal path or take a while to establish a good path, at the advantage of being significantly more energy efficient. Due to the stringent energy constraints of embedded M2M systems, the latter is typically the choice of design.

To this end, the IETF ROLL group has designed a routing protocol for precisely this type of networks where nodes have limited resources and operate over lossy channels. The protocol design can be found under [IETF - Routing Over Low power and Lossy networks (Active WG), 2010] and the quantification of its benefits under [Watteyne et al., 2010]. The core of the protocol is the metric deciding on the choice of the actual route w.r.t. all possible routes available. Due to the distributed nature of the embedded system without central control, this decision is done locally where a transmitter chooses its target receiver (from all possible receivers) as the one which possesses minimum rank and costs least to communicate to. The latter, i.e. the transmission cost, is referred to as **ETX** and reflects the energy cost to transmit a packet over a given link. This energy cost is directly proportional to the average number of transmissions $\overline{N}_{tx}$ which has previously been derived. The former, i.e. the rank, is essentially the aggregated number of transmissions between the given node and the sink, and thus reflects the “distance" of the node to the sink including the channel conditions and choice of technology on the way.

Therefore, without going into the details of the actual protocol design, the best route from all available routes is the one which exhibits the minimum aggregated number of transmissions. The protocol of IETF ROLL is designed such that said path is found iteratively by updating the rank (through the **ETX**'s) at specific times, stipulated by the trickle timer. These considerations were taken into account for our studies data aggregation.

Appendix B

Modes of Operation

"Verba volant, scripta manent" - "The words fly, the writings remain"
Caio Tito.

Contents

This Annex was partially inspired from NIST recommendations [CMAC, 2007, GCM, 2007, CCM, 2007] regarding modes of operation and it does not contain new technical contributions from our studies, such as simulations and/or mathematical results. We have decided to do not further explore this field of security because it was out-of-scope for our research regarding security in M2M networks; proposing and studying a new algorithm and/or modes of operation is a long procedure that may require years of analysis and results.

In general, modes of operation is the procedure of enabling the repeated and secure use of a block cipher with a single secret key. REF=("Block Cipher Modes". NIST Computer Security Resource Center.) Their functions is thus to decrease the probability of a successful cryptanalysis attacks providing mandatory security services to every kind of communication systems where block ciphers can be implemented. This annex is particulary interesting to justify our consideration regarding the WAVE2M packet formats that were proposed in Section 5.

To this end, this section is organized as follows: first, we present the AES symmetric-key block cipher introducing their main features and we treat the modes of operation that are recommended from NIST, and second, we focus on each mode of operation providing security recommendations. These modes of operation are able to increase the complexity of

AES and more specifically can provide: only authentication (CMAC mode of operation) and confidentiality and authentication (CCM and GCM modes of operation) security services. We have selected only these modes of operation because only confidentiality security service is not a requirement in M2M networks.

B.1 CIA with AES and modes of operation

In order to provide Confidentiality, Integrity and Authentication, WAVE2M should implement AES [Sanchez-Avila and Sanchez-Reillol, 2001]. AES is a symmetric block cipher that can be improved using several different modes of operation. The selection of the mode of operation depends on which security service is necessary for the desired communication.

On one hand, regarding AES, it is a symmetric-key cipher. It consists of an encryption and a decryption transformation that uses the same key value as input, or more accurately, a key input for encryption and a key input for decryption that are basically equal or “easy" to derive from the same seed. Since it is a block cipher, it operates on fixed length input blocks. Message Integrity Codes, that are basically arbitrary length hash function with additional key input, are also considered as symmetric primitives and are used to provide Autentication/Integrity with AES.

However, in general, the major advantage of symmetric-key primitives is their low computational complexity. The algorithms can be performed in reasonable time on small microcontrollers or small dedicated hardware modules that can be designed to increase performance or decrease energy consumption. Since security of encryption schemes using symmetric-key encryption is established as long as the key is, which represents the shared secret between the communications partners, such schemes are often referred to as secret-key crypto system. Delivery of the keys over insecure communication channels to all communication partners and management of all active keys in a system is the major concern for symmetric-key cryptography and thus of AES. This is the main reason why we have also dedicated part of our research to define a secure Key distribution protocol.

On the other hand, regarding the modes of operation, cryptographic primitives can be used in a multitude of ways in order to provide different security assurances. A method which defines a specific usage of one or more cryptographic primitives is referred to as mode of operation. For constrained devices whose limitations might only allow the support of a single primitive, the careful selection of the mode of operation may be the sole chance for providing the desired security services. Block ciphers are very popular building blocks and there exist a large number of modes of operation for them. The most common ones deal with providing either confidentiality or data interity/sender authentication. Some modes of operation offer both of these assurances. Here, we focus on: only authentication and both confidentiality and authentication modes of operation which are following described and analyzed.

B.2 CMAC, Authentication mode

Authentication modes of operation can be used to achieve data integrity and authentication. Generally, the input for such a mode is a message and the output is a Message Authentication Code or a Message Integrity Code. A verifier can then check whether the MAC has been generated by an entity which knows the correct key. The size of the MAC has to be chosen in such a way, that the chance of an attacker to guess a valid MAC is limited to an acceptable small degree. Another potential issue which is not addressed by the authentication mode is that an attacker can record authentication messages and replay them at a later time. If protection against such replay attacks is a concern, it must be provided by additional measures, e.g. inclusion of time-stamps or sequence numbers in messages. Cipher Block Chaining MAC (CBC-MAC) is a very basic method for generating a MAC base on CBC mode.

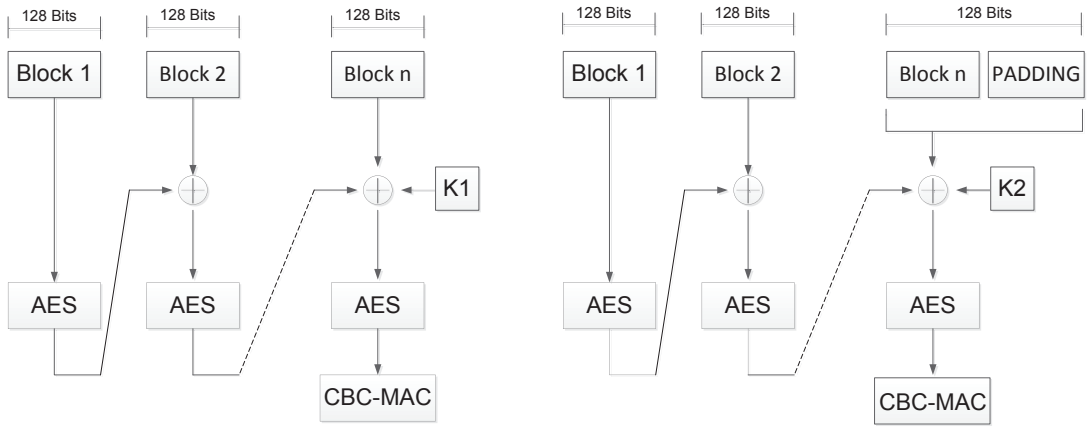

Figure B.1: The two cases of CBC-MAC generation.

CBC-MAC simply consists of CBC mode encryption with a zero IV, where the last resulting cipher text block is used as MAC. However, CBC-MAC has security deficiencies, e.g. it is not secure for variable-sized messages. There have been various extensions to solve the problems of CBC-MAC. The Cipher-based MAC (CMAC) [Liu et al., 2007] [CMAC, 2007] mode is one of the extensions of CBC-MAC which has been recommended by NIST [NIST, 2011]. It derives two sub keys (K_1 and K_2) from the key and divides the input message into blocks (conforming to the cipher's block size), padding the last block if necessary. Figure B.1 clearly illustrates these two cases. Depending on whether padding was necessary, K_1 or K_2 is exclusive-or'ed to the last block. Then CBC-MAC encryption is performed on the message blocks and the last block (or a subset of it) is returned as MAC.

B.2.1 Security recommendation for CMAC mode

The verification process determines whether the purported MAC on a message is the valid output of the MAC generation process applied to the message. The output of the MAC verification determines the assurance that the received message is authentic and it was not modified. However, the output result of the verification process can be:

- **INVALID**. The message is definitely not authentic, i.e., it did not originate from a source that executed the generation process on the message to produce the purported MAC.
- **VALID**. The design of the mode provides assurance that the message is authentic and, hence, was not corrupted in transit; however, this assurance, as for an MAC algorithm, is not absolute.

In the second case, an attacker, i.e., a party without access to the key or to the MAC generation process may have simply guessed the correct MAC for the message. In particular, if the attacker selects a MAC at random from the set of strings of length T_{len} bits, then the probability is 1 in $2^{T_{len}}$ that the MAC will be valid. Consequently, larger values of T_{len} provide greater protection against such an event. Of course, an attacker may attempt to systematically guess many different MACs for a message, or for different messages, and thereby increase the probability that one (or more) of them will be accepted as valid. For this reason, a system should limit the number of unsuccessful verification attempts for each key.

Selection of the MAC Length

Larger values of T_{len} provide greater assurance against guessing attacks. The performance tradeoff is that larger values of T_{len} require more bandwidth/storage for the MAC. For most applications, a value for T_{len} that is at least 64 bits should provide sufficient protection against guessing attacks. A value of T_{len} that is less than 64 bits shall only be used in conjunction with a careful analysis of the risks of accepting an inauthentic message as authentic.

In particular, a value of T_{len} smaller than 64 bits should not be used unless the controlling protocol or system sufficiently restricts the number of times that the verification process can return INVALID, across all implementations with any given key. For example, the short duration of a session or, more generally, the low bandwidth of the communication channel may preclude many repeated trials. This risk can be quantified in terms of the following two bounds: 1) the highest acceptable probability for an inauthentic message to pass the verification process, and 2) a limit on the number of times that the output is the error message INVALID before the key is retired, across all implementations of the verification process for the key. Given estimates of these quantities, denoted $RISK$ and $MAX - INV$, respectively, T_{len} should satisfy the following inequality:

$$Tlen \geq \lg(MAX - INV/RISK) \tag{B.1}$$

For example, suppose that the MAC verification process(es) within a system will not output INVALID for more than 1024 messages before the key is retired (i.e., $MAX - INV = 2^{10}$), and that the users can tolerate about a one in a million chance that the system will accept an inauthentic message (i.e., $RISK = 2^{-20}$). In this case, any value of T_{len} that is greater than or equal to 30 bits satisfies the inequality.

Protection against Replay of Messages

As described in above, the successful verification of a MAC for a message gives assurance that the source of the message executed the MAC generation algorithm to create the MAC; however, the party that presented the message and MAC for verification may not be the original source of the message. Therefore, the CMAC algorithm does not inherently prevent an attacker from intercepting a legitimate message and its MAC and "replaying" them for verification at a later time, for example, in an attempt to impersonate a party that has access to the key. In some protocols an attacker may even be able to present to a verifier a message-MAC pair that the verifier itself generated earlier in the protocol.

The controlling protocol or application may protect against such an event by incorporating certain identifying information into the initial bits of every message. Examples of such information include a sequential message number, a time-stamp, or a nonce. Upon successful verification of the message, this information may provide a means for the detection of replayed messages, out-of-sequence messages, or missing messages.

Message Span of the Key

The message span of a key is the total number of messages for which MACs are generated across all implementations of CMAC with that key. The message span of the key affects the security of the system against attacks that are based on the detection of a pair of distinct messages with the same MAC before its truncation. Such a pair is called a collision. As with other block cipher-based MAC algorithms, an attacker may be able to exploit a collision to produce the valid MAC for a new message, the content of which may be largely of the attacker's choosing. Such an event would be a fundamental breach of the expected authentication assurance.

In principle, collisions must exist because there are many more possible messages than MACs; in practice, however, collisions may not occur among the messages for which MACs are actually generated during the lifetime of the key. The probability that at least one collision will occur depends mostly on the message span of the key relative to the block size, 'b', of the underlying block cipher. For example, a collision is expected to exist among a set of $2^{b/2}$ arbitrary messages; in other words, 2^{64} messages for the AES algorithm. This property was one of the motivations to develop the AES with a block size of 128 bits.

For any system in which CMAC is implemented, the risk that an attacker can detect and exploit a collision shall be limited to a level that is appropriate to the value of the data. A simple

and prudent method to achieve this goal is to establish and enforce an appropriate limit on the message span of any CMAC key, which in turn limits the probability that a collision will even occur. For general purpose applications, the default recommendation is to limit the key to no more than 2^{48} messages when the block size of the underlying block cipher is 128 bits, as with the AES algorithm. Within these limits, the probability that a collision will occur is expected to be less than one in a billion for the AES algorithm. These are the considerations used as basis to properly define the secure crypto-period used on Section 4 where we focus on secure key management schemes.

In some cases, a limit on the message span of a key may be established and enforced within a key management infrastructure by an appropriate constraint on the time span during which the key remains in effect, i.e., its crypto-period.

B.3 CCM and GCM, Confidentiality and authentication mode

With the help of the aforementioned modes of operation, the provision of confidentiality and data integrity/ authentication requires the use of two separate modes. As a general rule, separate keys should be used if two (or more) modes of operation are employed to process the same message. This is because a common presumption of all these modes is the sole use of the key within the specified functions. Additional use of the same key in another mode may enable various forms of attacks. Another class of modes of operation can provide both authentication and confidentiality under the use of a single key. While an authenticated encryption (AE) scheme allows to protect authentication and confidentiality of a message, an authenticated encryption with associated data (AEAD) scheme aims to provide authenticity of additional data (associated data) as well. A good usage example for the latter is network packet protection, where the payload needs to be encrypted and authenticated, while it is sufficient to authenticate the packet header.

There have been a number of newly proposed modes of operations which address the AEAD problem. NIST [NIST, 2007] has recommended the Counter with CBC-MAC mode (CCM) [Huai et al., 2009] [CCM, 2007] and Galois with Counter mode (GCM) [Satoh et al., 2009] [GCM, 2007]:

- **CCM** mode is intended for use when the complete message is available for processing, i.e. it is not suited for online stream processing. It basically generates a MAC using the CBC-MAC mode (zero IV), and encrypts the message and the MAC in CTR mode. CTR mode is shown in Figure B.2. The associated data is just used for MAC generation but is not CTR encrypted.

- **GCM** mode uses a variant of CTR mode for encryption. Authentication is done over the resulting cipher text and the associated data with a universal hash function over binary finite (Galois) field. This construction allows faster implementation in hardware as well as in software under the use of look-up tables. However, when

software implementation cannot support large look-up tables, the performance degrades significantly. Implementation in nodes with limited memory could be hence less attractive if preventive studies are not well conducted.

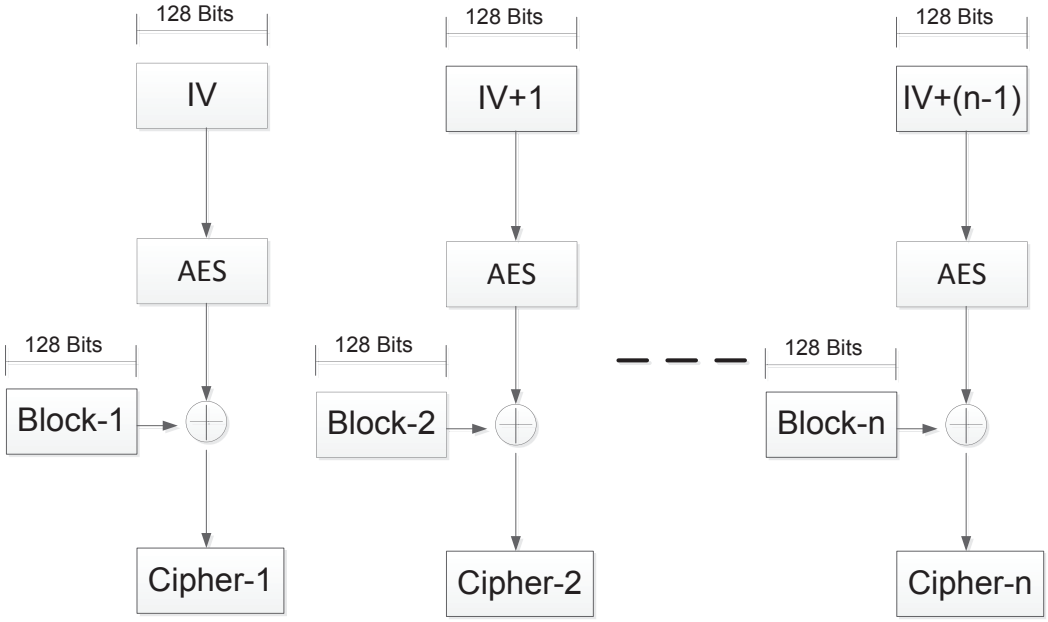

Figure B.2: CTR Encryption mode.

B.3.1 Security recommendations for CCM mode

CCM is intended for use in a packet environment, i.e., when all of the data is available in storage before CCM is applied; CCM is not designed to support partial processing or stream processing. The input to CCM includes three elements: 1) data that will be both authenticated and encrypted, called the payload; 2) associated data, e.g., a header, that will be authenticated but not encrypted; and 3) a unique value, called a nonce, that is assigned to the payload and the associated data.

CCM consists of two related processes: generation-encryption and decryption-verification. In generation-encryption, cipher block chaining is applied to the payload, the associated data, and the nonce to generate a message authentication code; then, counter mode encryption is applied to the MAC and the payload to transform them into an unreadable form, called the ciphertext. In decryption-verification, counter mode decryption is applied to the purported ciphertext to recover the MAC and the corresponding payload; then, cipher block chaining is applied to the payload, the received associated data, and the received nonce to verify the correctness of the MAC. Successful verification provides assurance that the payload and the associated data originated from a source with access to the key.

With that said, is clear that CCM is a generic authenticate-and-encrypt block cipher mode that needs several parameters to provide the requested security services. These security parameters are i) the secret key to provide confidentiality with AES or others block cipher; ii) the counter or initialization vector size that is used from the CTR mode as input data to encrypt and decrypt the blocks; and finally iii) the nonce that is used as part of the first block to compute the MAC with CBC-MAC mode.

Threats analysis

We claim that this block cipher mode is secure against attackers limited to 2^{128} steps of operation if the key K is 256 bits or larger. There are fairly generic pre-computation attacks against all block cipher modes that allow a meet-in-the-middle attack on the key K. If these attacks can be made, then the theoretical strength of this, and any other, block cipher mode is limited to $2^{(n/2)}$ where n is the number of bits in the key. The strength of the authentication is of course limited by the authentication field size.

Users of smaller key sizes (such as 128-bits) should take precautions to make the pre-computation attacks more difficult. Repeated use of the same nonce value (with different keys of course) ought to be avoided. One solution is to include a random value within the nonce. Of course, a packet counter is also needed within the nonce. Since the nonce is of limited size, a random value in the nonce provides a limited amount of additional security.

Nevertheless, the main difficulty in specifying this mode is the trade-off between nonce size and counter size. Some applications use only small messages, but would rather have a larger nonce. The authentication field size gives the traditional trade-off between message expansion and probability of forgery. For most applications, we recommend choosing this parameter at least equal to 8 octets.

Rationale

CCM encryption is a straightforward application of CTR mode. In this mode of operation the CBC-MAC collision attacks is avoided by encrypting the MAC value. If the block cipher behaves as a pseudo-random permutation, then the key stream is indistinguishable from a random string. Thus, the attacker gets no information about the CBC-MAC results. The only avenue of attack that is left is a differential-style attack, which has no significant chance of success if the block cipher is a pseudo-random permutation.

To simplify implementation, CCM mode uses the same block cipher key for the encryption and authentication functions. If the block cipher behaves like a random permutation, then the outputs are independent of each other, up to the insignificant limitation that they are all different. The only cases where the inputs to the block cipher can overlap are an intermediate value in the CBC-MAC and one of the other encryptions.

As all the intermediate values of the CBC-MAC computation are essentially random

(because the block cipher behaves like a random permutation) the probability of such a collision is very small. Even if there is a collision, these values only affect the MAC value, which is encrypted so that an attacker cannot deduce any information, or detect any collision.

In this mode of operation, particular care has been taken to ensure that the blocks used by the authentication function match up with the blocks used by the encryption function. This should simplify hardware implementations, and reduce the amount of byte-shifting required by software implementations.

Nonce Suggestions

The main requirement for CCM mode it is that, within the scope of a single key, the nonce values are unique for each message. A common technique is to number messages sequentially, and to use this number as the nonce. Sequential message numbers are also used to detect replay attacks and to detect message reordering, so in many situations the sequence numbers are already available.

Users of CCM, and all other block cipher modes, should be aware of pre-computation attacks. These are effectively collision attacks on the cipher key. Let suppose the key K is 128 bits, and the same nonce value is used with many different keys. The attacker chooses a particular nonce at she fixes it as test case. She chooses 264 different keys at random and computes a table entry for each K value, generating a pair of the form (K, S_1) where S_1 is the first key stream block. Given the key and the nonce is possible computing S_1. She then waits for messages to be sent with nonce. We will assume the first 16 bytes of each message are known so that she can compute S_1 for each message. She looks in her table for a pair with a matching S_1 value. She can expect to find a match after checking about 2^{64} messages. Once a match is found, the other part of the matched pair is the key in question. The total workload of the attacker is only 2^{64} steps, rather than the expected 2^{128} steps. Similar pre-computation attacks exist for all block cipher modes.

The main weapon against pre-computation attacks is to use a larger key. Using a 256-bit key forces the attacker to perform at least 2^{128} pre-computations, which is infeasible. In situations where using a large key is not possible or desirable (for example, due to the resulting performance impact), users can use part of the nonce to reduce the number of times any specific nonce value is used with different keys. If there is room in the nonce, the sender could add a few random bytes, and send these random bytes along with the message. This makes the pre-computation attack much harder, as the attacker now has to pre-compute a table for each of the possible random values. An alternative is to use something like the sender's Ethernet address. Including the Ethernet address forces the attacker to perform the pre-computation specifically for a specific source address, and the resulting table could not be used to attack anyone else. Although these solutions can all work, they need careful analysis and almost never entirely prevent these attacks. Where possible, we recommend using a larger key, as this solves all the problems.

Restrictions

To preserve security, implementations need to limit the total amount of data that is encrypted with a single key; the total number of block cipher encryption operations in the CBC-MAC and encryption together cannot exceed 2^{61}. This allows nearly 2^{64} octets to be encrypted and authenticated using CCM. This is roughly 16 million tera-bytes, which should be more than enough for most applications. In an environment where this limit might be reached, the sender MUST ensure that the total number of block cipher encryption operations in the CBC-MAC and encryption together does not exceed 2^{61}. Receivers that do not expect to decrypt the same message twice MAY also check this limit.

The recipient MUST verify the CBC-MAC before releasing any information such as the plain-text. If the CBC-MAC verification fails, the receiver MUST destroy all information, except for the fact that the CBC-MAC verification failed.

B.3.2 Security recommendations for GCM mode

The following are three important design considerations for GCM modules:

1. The freshness of keys shall be assured.

2. The IV implemented from the two functions used from CGM mode, shall be a critical security parameter as until the authenticated encryption function is invoked with the IV. Prior to this invocation, the IV shall be provided the same protection as other critical security parameters.

3. A loss of power to the module shall not cause the repetition of IVs. If the generation unit cannot recover from a loss of power, then the authenticated encryption function shall enter a failure state until a fresh key can be established.

The IV construction that is implemented affects the options for recovery from a loss of power. For the deterministic construction, all of the deterministic elements that are necessary to construct the IV would have to be available when power is restored. For example, these elements could be stored in non-volatile memory.

Key Establishment

The following norm, which in general is valid for every secret key cryptographic algorithms, takes on explicit importance for GCM to support the uniqueness requirement:

- Any GCM key that is established among its intended users shall, with high probability, be fresh.

In practice, the requirement should ensure that a key is fresh when it is generated, if the generation mechanism is resistant to tampering. Achieving such resistance usually imposes requirements on the management of the key generation mechanism.

In particular, if the key generation mechanism is deterministic, then the management of the mechanism shall provide strong assurance that no outside entity can induce the repetition of a previous set of inputs to the mechanism, or otherwise cause the repetition of a previous output. For example, GCM keys may be established using the key derivation functions of the following protocols: Transport Layer Security, Internet Key Exchange V_1 and V_2, and Secure Shell.

Similarly, if a new key must be transported to its intended recipient(s), the method of transport/distribution shall provide strong assurance against “replay", so that no party can induce the substitution of a previous key for the intended key. GCM keys should be established within the framework of an approved key management structure to assure their freshness, as well as their confidentiality and authenticity.

IV Requirement

In GCM the IVs must fulfill the following “uniqueness" requirement:

- The probability that the authenticated encryption function ever will be invoked with the same IV and the same key on two (or more) distinct sets of input data shall be no greater than 2^{-32}.

Compliance with this requirement is crucial to the security of GCM. Across all instances of the authenticated encryption function with a given key, if even one IV is ever repeated, then the implementation may be vulnerable to the forgery attacks. In practice, this requirement is almost as important as the secrecy of the key.

GCM weaknesses

Niels Ferguson [Ferguson, 2005] shows two weaknesses in the authentication functionality of GCM when it is used with a short authentication tag. The first weakness raises the probability of a successful forgery significantly. The second weakness reveals the authentication key if the attacker manages to create successful forgeries.

B.4 Overview Table

The following table summarizes the aforementioned modes of operation, specifying the algorithms used, the security services provided, the keying data, the maximum number of key invocations, field length, etc.

Table B.1: Mode of operation characteristics.

Mode of operation	**CMAC**	**CCM**	**GCM**
Algorithms	CBC-MAC variant.	CTR and CMC-MAC.	CTR variant and CBC-MAC variant.
Security services	Authentication	Authentication and Confidentiality	Authentication and Confidentiality
Key parameters	Key and Sub-keys (K1 and K2).	Key, Nonce and Associated data	IV (96 bits recommended), Key and Authentication Tag.
Maximum recommended invocation	2^{48}	2^{61}	2^{32}
Against replay attack	Need a timestamp.	Nonce sequence number.	Need a time stamp.
Input data	Plain-text.	Nonce (N), Plain-text (P) and Associated data (A).	Plain-text, Additional Authenticated Data and IV.
Output data	MIC or MAC.	Cipher-text and MIC or MAC.	Cipher-text and Authentication Tag.
Information protected from the algorithms	Authenticity of plain-text	Plain-text (encrypted) and Associated data (optional and in clear)	Plain-text (encrypted) and Additional Authenticated Data (in clear)
Authentication field recommended	32bit	64bit	64bit

Bibliography

[100.11a, 2010] 100.11a, I. (2010). The ISA 100.11a website.

[802, 2010] 802 (2010). The IEEE 802.X website.

[802.X, 2010] 802.X (2010). The ieee 802.x website.

[Abramowitz and Stegun, 1965] Abramowitz, M. and Stegun, I. A. (1965). *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*. Dover Publications.

[Ahn and Heidemann, 2002] Ahn, J.-S. and Heidemann, J. (2002). An adpative FEC algorithm for mobile wireless networks. Technical Report ISI-TR-555, USC/Information Sciences Institute.

[Akyildiz et al., 2002] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102 – 114.

[Atapattu et al., 2010] Atapattu, S., Tellambura, C., and Jiang, H. (2010). Representation of composite fading and shadowing distributions by using mixtures of gamma distributions. In *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, pages 1 –5.

[Bachir et al., 2010] Bachir, A., Dohler, M., Watteyne, T., and Leung, K. (2010). MAC essentials for wireless sensor networks. *Communications Surveys Tutorials, IEEE*, 12(2):222 –248.

[Baig, 2011] Baig, Z. (2011). Pattern recognition for detecting distributed node exhaustion attacks in wireless sensor networks. *Computer Communications*, 34(3):468–484. Special Issue of Computer Communications on Information and Future Communication Security.

[Barker, 2010] Barker, R. (2010). Security aspects in 6lowpan networks. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, page 660.

[Bartoli et al., 2011] Bartoli, A., Hernandez-Serrano, J., Soriano, M., Dohler, M., Kountouris, A., and Barthel, D. (2011). Secure lossless aggregation over fading and shadowing channels for smart grid m2m networks. *Smart Grid, IEEE Transactions on*, 2(4):844 –864.

[Bouhenguel et al., 2008] Bouhenguel, R., Mahgoub, I., and Ilyas, M. (2008). Bluetooth security in wearable computing applications. In *High Capacity Optical Networks and Enabling Technologies, 2008. HONET 2008. International Symposium on*, pages 182 –186.

[Castelluccia et al., 2009] Castelluccia, C., Chan, A. C.-F., Mykletun, E., and Tsudik, G. (2009). Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM Trans. Sen. Netw.*, 5(3):20:1–20:36.

[Castelluccia et al., 2005] Castelluccia, C., Mykletun, E., and Tsudik, G. (2005). Efficient aggregation of encrypted data in wireless sensor networks. In *The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, MobiQuitous 2005.*, pages 109 – 117.

[Castro and Liskov, 1999] Castro, M. and Liskov, B. (1999). Practical byzantine fault tolerance. In *Proceedings of the third symposium on Operating systems design and implementation*, OSDI '99, pages 173–186, Berkeley, CA, USA. USENIX Association.

[CCM, 2007] CCM (2007). Nist special publication 800-38c.

[Chan et al., 2007] Chan, H., Perrig, A., Przydatek, B., and Song, D. (2007). SIA: Secure information aggregation in sensor networks. *J. Comput. Secur.*, 15(1):69–102.

[Chan et al., 2006] Chan, H., Perrig, A., and Song, D. (2006). Secure hierarchical in-network aggregation in sensor networks. In *Proceedings of the 13th ACM conference on Computer and communications security*, CCS '06, pages 278–287, New York, NY, USA. ACM.

[Chen et al., 2010] Chen, F., German, R., and Dressler, F. (2010). Towards ieee 802.15.4e: A study of performance aspects. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*, pages 68 –73.

[Chung-Wei Phan, 2002] Chung-Wei Phan, R. (2002). Cryptanalysis of full skipjack block cipher. *Electronics Letters*, 38(2):69 –71.

[CMAC, 2007] CMAC (2007). Nist special publication 800-38b.

[Dingledine et al., 2004] Dingledine, R., Mathewson, N., and Syverson, P. (2004). Tor: The second-generation onion router. In *In Proceedings of the 13th USENIX Security Symposium*, pages 303–320.

[Docherty and Koelmans, 2011] Docherty, J. and Koelmans, A. (2011). A flexible hardware implementation of SHA-1 and SHA-2 hash functions. In *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, pages 1932–1935.

[Douceur, 2002] Douceur, J. R. (2002). The sybil attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, pages 251–260, London, UK, UK. Springer-Verlag.

[Du et al., 2003] Du, W., Deng, J., Han, Y., and Varshney, P. (2003). A witness-based approach for data fusion assurance in wireless sensor networks. In *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, volume 3, pages 1435 – 1439 vol.3.

[Dust Smart Mesh Network, 2012] Dust Smart Mesh Network (2012). Dn6000/m600, dust corporation.

[Fan et al., 2009] Fan, C., Tan, J., and Zheng, P. (2009). Low-speed wireless networks research and simulation based on RC5. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1 –4.

[Ferguson, 2005] Ferguson, N. (2005). Authentication weaknesses in gcm.

[Gagneja, 2012] Gagneja, K. (2012). Pairwise post deployment key management scheme for heterogeneous sensor networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1 –2.

[GCM, 2007] GCM (2007). Nist special publication 800-38d.

[Gradshteyn and Ryzhik, 2000] Gradshteyn, I. S. and Ryzhik, I. M. (2000). *Table of Integrals, Series, and Products*. Academic Press, San Diego, USA, sixth edition.

[Haas et al., 2011] Haas, J., Hu, Y.-C., and Laberteaux, K. (2011). Efficient Certificate Revocation List Organization and Distribution. *Selected Areas in Communications, IEEE Journal on*, 29(3):595 –604.

[Harbor Research, 2011] Harbor Research (2011). Next generation platform innovation in m2m.

[Hong and Landay, 2004] Hong, J. I. and Landay, J. A. (2004). An architecture for privacy-sensitive ubiquitous computing. In *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, MobiSys '04, pages 177–189, New York, NY, USA. ACM.

[Hu and Evans, 2003] Hu, L. and Evans, D. (2003). Secure aggregation for wireless networks. In *Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT'03 Workshops)*, SAINT-W '03, page 384, Washington, DC, USA. IEEE Computer Society.

[Hu et al., 2002] Hu, Y., Perrig, A., and Johnson, D. B. (2002). Wormhole detection in wireless ad hoc networks. Technical report, Rice University Department of Computer Science.

[Huai et al., 2009] Huai, L., Zou, X., Liu, Z., and Han, Y. (2009). An energy-efficient aes-ccm implementation for ieee802.15.4 wireless sensor networks. In *Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC '09. International Conference on*, volume 2, pages 394 –397.

[Hwang et al., 2011] Hwang, M.-S., Chong, S.-K., and Ou, H.-H. (2011). On the security of an enhanced UMTS authentication and key agreement protocol. *European Transactions on Telecommunications*, 22(3):99–112.

[IBM, 2010] IBM (2010). Smarter planet initiative.

[IEEE Computer Society, 2010] IEEE Computer Society (2010). IEEE Std 802.15.4-2006.

[IETF - Routing Over Low power and Lossy networks (Active WG), 2010] IETF - Routing Over Low power and Lossy networks (Active WG) (2010). Roll status pages. Last accessed October 2010.

[Ishmanov et al., 2011] Ishmanov, F., Malik, A. S., and Kim, S. W. (2011). Energy consumption balancing (ECB) issues and mechanisms in wireless sensor networks (wsns): a comprehensive overview. *European Transactions on Telecommunications*, 22(4):151–167.

[Jadia and Mathuria, 2005] Jadia, P. and Mathuria, A. (2005). Efficient secure aggregation in sensor networks. In Bouge, L. and Prasanna, V., editors, *High Performance Computing - HiPC 2004*, volume 3296 of *Lecture Notes in Computer Science*, pages 111–119. Springer Berlin, Heidelberg.

[Jyh-Cheng Chen et al., 2005] Jyh-Cheng Chen, Ming-Chia Jiang, and Yi-wen Liu (2005). Wireless LAN security and IEEE 802.11i. *Wireless Communications, IEEE*, 12(1):27 – 36.

[Karlof et al., 2004a] Karlof, C., Sastry, N., and Wagner, D. (2004a). TinySec: a link layer security architecture for wireless sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, pages 162–175, New York, NY, USA. ACM.

[Karlof et al., 2004b] Karlof, C., Sastry, N., and Wagner, D. (2004b). Tinysec: a link layer security architecture for wireless sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 162–175, New York, NY, USA. ACM Press.

[Karnouskos et al., 2012] Karnouskos, S., Da Silva, P., and Ilic, D. (2012). Energy services for the smart grid city. In *Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on*, pages 1–6.

[Kim et al., 2010] Kim, B. H., Ahn, H.-J., Kim, J. O., Yoo, M., Cho, K., and Choi, D. (2010). Application of M2M technology to manufacturing systems. In *Information and Communication Technology Convergence (ICTC), 2010 International Conference on*, pages 519–520.

[Krontiris et al., 2008] Krontiris, I., Dimitriou, T., Soroush, H., and Salajegheh, M. (2008). *Wireless Sensors Networks Security*, chapter WSN Link-layer Security Frameworks, pages 142–163. IOS Press.

[Labraoui et al., 2011] Labraoui, N., Gueroui, M., and Aliouat, M. (2011). Secure DV-Hop localization scheme against wormhole attacks in wireless sensor networks. *European Transactions on Telecommunications*, pages n/a–n/a.

[Lai et al., 2002] Lai, Y.-K., Chen, L.-G., Lai, J.-Y., and Parng, T.-M. (2002). VLSI architecture design and implementation for twofish block cipher. In *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, volume 2, pages II–356 – II–359 vol.2.

[Lee et al., 1995] Lee, W.-B., Chang, C.-C., and Harn, L. (1995). Comment on &idquo: digital signature with (t,n) shared verification based on discrete logarithms rdquo. *Electronics Letters*, 31(3):176 –177.

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

[Liu et al., 2007] Liu, S., Fan, K.-W., and Sinha, P. (2007). CMAC: An energy efficient mac layer protocol using convergent packet forwarding for wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on*, pages 11 –20.

[Lopez et al., 2009] Lopez, J., Roman, R., and Alcaraz, C. (2009). Analysis of security threats, requirements, technologies and standards in wireless sensor networks. In Aldini, A., Barthe, G., and Gorrieri, R., editors, *Foundations of Security Analysis and Design V*, volume 5705 of *Lecture Notes in Computer Science*, pages 289–338. Springer Berlin / Heidelberg.

[Loree et al., 2009] Loree, P., Nygard, K., and Du, X. (2009). An efficient post-deployment key establishment scheme for heterogeneous sensor networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1 –6.

[Lu et al., 2011] Lu, R., Li, X., Liang, X., Shen, X., and Lin, X. (2011). GRS: The green, reliability, and security of emerging machine to machine communications. *Communications Magazine, IEEE*, 49(4):28–35.

[Luk et al., 2007] Luk, M., Mezzour, G., Perrig, A., and Gligor, V. (2007). Minisec: A secure sensor network communication architecture. In *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, pages 479 –488.

[Lv et al., 2012] Lv, C., Li, H., Ma, J., and Zhang, Y. (2012). Vulnerability analysis of elliptic curve cryptography-based rfid authentication protocols. *Transactions on Emerging Telecommunications Technologies*, pages n/a–n/a.

[M2M communication, 2010] M2M communication (2010). The ICT EXALTED project.

[Mahimkar and Rappaport, 2004] Mahimkar, A. and Rappaport, T. (2004). Securedav: a secure data aggregation and verification protocol for sensor networks. In *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, volume 4, pages 2175 – 2179 Vol.4.

[Maleysson and Dugas, 2005] Maleysson, L. and Dugas, C. (2005). Configuring and managing a large-scale monitoring network solving real world challenges for ultra low powered and long-range wireless mesh networks. In *sOc-EUSAI '05: Proceedings of the 2005 joint conference on Smart objects and ambient intelligence*, pages 225–230, New York, NY, USA. ACM.

[Manji and Mandayam, 2000] Manji, S. and Mandayam, N. B. (2000). Block Error Probability using List Viterbi Decoding with Hard Decisions.

[Mary, 2008] Mary, P. (2008). Performance of Wireless Systems over Fading and Shadowing Channels. *PhD Thesis, INSA-Lyon*.

[Mirkovic and Reiher, 2004] Mirkovic, J. and Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *SIGCOMM Comput. Commun. Rev.*, 34:39–53.

[Muhaidheen, 2007] Muhaidheen, M. (2007). New application of M2M in railway protection. In *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on*, volume 4, pages 110–114.

[Nath et al., 2009] Nath, S., Yu, H., and Chan, H. (2009). Secure outsourced aggregation via one-way chains. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, SIGMOD '09, pages 31–44, New York, NY, USA. ACM.

[Network layer, 2007] Network layer (2007). Rfc 4944.

[NIST, 2005] NIST (2005). Recommendation for block cipher modes of operation: The CMAC mode for authentication.

[NIST, 2007] NIST (2007). Using aes-ccm and aes-gcm authenticated encryption in the cryptographic message syntax (cms).

[NIST, 2011] NIST (2011). National Institute of Standards and Technology (NIST). Last accessed August 2011.

[Ozdemir, 2007] Ozdemir, S. (2007). Secure and reliable data aggregation for wireless sensor networks. In *Proceedings of the 4th international conference on Ubiquitous computing systems*, UCS'07, pages 102–109, Berlin, Heidelberg. Springer-Verlag.

[Ozdemir, 2008] Ozdemir, S. (2008). Functional reputation based reliable data aggregation and transmission for wireless sensor networks. *Comput. Commun.*, 31(17):3941–3953.

[Papadopoulos et al., 2012] Papadopoulos, S., Kiayias, A., and Papadias, D. (2012). Exact in-network aggregation with integrity and confidentiality. *Knowledge and Data Engineering, IEEE Transactions on*, PP(99):1.

[Park and Shin, 2004] Park, T. and Shin, K. G. (2004). LiSP: a lightweight security protocol for wireless sensor networks. *ACM Transactions on Embedded Computing Systems*, 3:634–660.

[Parra-Arnau et al., 2012] Parra-Arnau, J., Rebollo-Monedero, D., and Forné, J. (2012). A privacy-protecting architecture for collaborative filtering via forgery and suppression of ratings. In *Proceedings of the 6th international conference, and 4th international conference on Data Privacy Management and Autonomous Spontaneus Security*, DPM'11, pages 42–57, Berlin, Heidelberg. Springer-Verlag.

[Perrig et al., 2002] Perrig, A., Szewczyk, R., Tygar, J. D., Wen, V., and Culler, D. E. (2002). Spins: Security protocols for sensor networks. *Wireless Networks*, pages 521–534.

[Privacy, 2010] Privacy (2010). Guidelines for smart grid cyber security: Vol. 1, smart grid cyber security strategy, architecture, and high-level requirements, and vol. 2, privacy and the smart grid.

[Proakis, 1995] Proakis, J. G. (1995). *Digital Communications*. McGraw-Hill, New York, USA, third edition.

[Rebollo-Monedero and Forné, 2010] Rebollo-Monedero, D. and Forné, J. (2010). Optimized query forgery for private information retrieval. *Information Theory, IEEE Transactions on*, 56(9):4631 –4642.

[Reiter and Rubin, 1998] Reiter, M. K. and Rubin, A. D. (1998). Crowds: anonymity for web transactions. *ACM Trans. Inf. Syst. Secur.*, 1(1):66–92.

[Ren et al., 2005] Ren, J., Li, T., and Aslam, D. (2005). A power efficient link-layer security protocol (llsp) for wireless sensor networks. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 1002 –1007 Vol. 2.

[Ren et al., 2006] Ren, K., Lou, W., and Zhang, Y. (2006). Leds: Providing location-aware end-to-end data security in wireless sensor networks. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1 –12.

[Sallinen, 2010] Sallinen, M. (2010). Applications of wireless M2M communication. In *Information and Communication Technology Convergence (ICTC), 2010 International Conference on*, pages 384–385.

[Samarati, 2001] Samarati, P. (2001). Protecting respondents identities in microdata release. *Knowledge and Data Engineering, IEEE Transactions on*, 13(6):1010 –1027.

[Sanchez-Avila and Sanchez-Reillol, 2001] Sanchez-Avila, C. and Sanchez-Reillol, R. (2001). The rijndael block cipher (AES proposal) : a comparison with des. In *Security Technology, 2001 IEEE 35th International Carnahan Conference on*, pages 229 –234.

[Sanli et al., 2004] Sanli, H., Ozdemir, S., and Cam, H. (2004). Srda: secure reference-based data aggregation protocol for wireless sensor networks. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 7, pages 4650 – 4654 Vol. 7.

[Satoh et al., 2009] Satoh, A., Sugawara, T., and Aoki, T. (2009). High-performance hardware architectures for galois counter mode. *Computers, IEEE Transactions on*, 58(7):917 –930.

[Schuba et al., 1997] Schuba, C., Krsul, I., Kuhn, M., Spafford, E., Sundaram, A., and Zamboni, D. (1997). Analysis of a denial of service attack on tcp. In *Security and Privacy, 1997. Proceedings., 1997 IEEE Symposium on*, pages 208 –223.

[Serjantov et al., 2003] Serjantov, A., Dingledine, R., and Syverson, P. F. (2003). From a trickle to a flood: Active attacks on several mix types. In *Revised Papers from the 5th International Workshop on Information Hiding*, IH '02, pages 36–52, London, UK, UK. Springer-Verlag.

[Shin and Lee, 2004] Shin, H. and Lee, J. H. (2004). On the Error Probability of Binary and M-ary Signals in Nakagami-m Fading Channels. *IEEE Transactions on Communications*, 52(4):536–539.

[Smith et al., 2012] Smith, J., Clark, A., Staggemeier, A., and Serpell, M. (2012). A genetic approach to statistical disclosure control. *Evolutionary Computation, IEEE Transactions on*, 16(3):431–441.

[Sodagari et al., 2012] Sodagari, S., Attar, A., Leung, V. C., and Bilen, S. G. (2012). Combating channel eviction triggering denial-of-service attacks in cognitive radio networks. *Transactions on Emerging Telecommunications Technologies*, pages n/a–n/a.

[Song et al., 2005] Song, N., Qian, L., and Li, X. (2005). Wormhole attacks detection in wireless ad hoc networks: A statistical analysis approach. In *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 17 - Volume 18*, IPDPS '05, pages 289.1–, Washington, DC, USA. IEEE Computer Society.

[Sweeney, 1000] Sweeney, L. (1000). Uniqueness of Simple Demographics in the U.S. Population.

[Tsao et al., 2012] Tsao, T., Alexander, R., Dohler, M., Daza, V., and Lozano, A. (2012). A security framework for routing over low power and lossy networks. Internet Draft draft-ietf-roll-security-framework-07. ROLL - Networking WG.

[Tse and Viswanath, 2005] Tse, D. and Viswanath, P. (2005). *Fundamentals of Wireless Communication*. Cambridge University Press.

[Venkitasubramaniam and Anantharam, 2008] Venkitasubramaniam, P. and Anantharam, V. (2008). On the anonymity of chaum mixes. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, pages 534–538.

[Wagner, 2004] Wagner, D. (2004). Resilient aggregation in sensor networks. In *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, SASN 04, pages 78–87, New York, NY, USA. ACM.

[Wander et al., 2005] Wander, A., Gura, N., Eberle, H., Gupta, V., and Shantz, S. (2005). Energy analysis of public-key cryptography for wireless sensor networks. In *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, pages 324–328.

[Wang and Wyglinski, 2011] Wang, L. and Wyglinski, A. (2011). A combined approach for distinguishing different types of jamming attacks against wireless networks. In *Communications, Computers and Signal Processing (PacRim), 2011 IEEE Pacific Rim Conference on*, pages 809–814.

[Wang et al., 2007] Wang, P., Li, C., and Zheng, J. (2007). Combined data aggregation and encryption using clustered slepian-wolf coding for wireless sensor networks. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, pages 920 –925.

[Wang et al., 2006] Wang, Y., Attebury, G., and Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks. *Communications Surveys Tutorials, IEEE*, 8(2):2 –23.

[Watteyne et al., 2010] Watteyne, T., Barthel, D., Dohler, M., and Auge-Blum, I. (2010). Sense&sensitivity: A large-scale experimental study of reactive gradient routing. *Measurement Science and Technology, Special Issue on Wireless Sensor Networks - designing for real-world deployment and deployment experiences*.

[WAVE2M Community, 2012] WAVE2M Community (2012). WAVE2M community web page.

[Wei et al., 2009] Wei, J., Guo, S., and Xu, Q. (2009). Secure homomorphic aggregation algorithm of mixed operations in wireless sensor networks. In *E-Business and Information System Security, 2009. EBISS '09. International Conference on*, pages 1 –5.

[Westhoff et al., 2006] Westhoff, D., Girao, J., and Acharya, M. (2006). Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation. *Mobile Computing, IEEE Transactions on*, 5(10):1417 –1431.

[Wiki, 2010] Wiki (2010). Error function. Last accessed October 2010.

[WirelessHART, 2010] WirelessHART (2010). The WirelessHART website.

[WirelessHart, 2010] WirelessHart (2010). The wirelesshart website.

[Wood and Stankovic, 2006] Wood, A. D. and Stankovic, J. A. (2006). AMSecure: secure link-layer communication in tinyos for ieee 802.15.4-based wireless sensor networks. In *SenSys'06*, pages 395–396.

[Wu et al., 2006] Wu, K., Dreef, D., Sun, B., and Xiao, Y. (2006). Secure data aggregation without persistent cryptographic operations in wireless sensor networks. In *Performance, Computing, and Communications Conference, 2006. IPCCC 2006. 25th IEEE International*, pages 6 pp. –640.

[Xie and Pan, 2010] Xie, J. and Pan, X. (2010). An improved RC4 stream cipher. In *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, volume 7, pages V7–156 –V7–159.

[Yang et al., 2006] Yang, Y., Wang, X., Zhu, S., and Cao, G. (2006). SDAP: A secure hop-by-hop data aggregation protocol for sensor networks. In *In ACM MOBIHOC 06*, pages 356–367. ACM Press.

[Zahariadis et al., 2010] Zahariadis, T., Leligou, H. C., Trakadas, P., and Voliotis, S. (2010). Trust management in wireless sensor networks. *European Transactions on Telecommunications*, 21(4):386–395.

[Zhang, 2009] Zhang, R. (2009). Analysis of energy-delay performance in multi-hop wireless sensor networks. PhD Thesis.

[Zhang et al., 2007] Zhang, W., Liu, Y., and Das, S. K. (2007). Aggregation supportive authentication in wireless sensor networks: A watermark based approach. In *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pages 1 –11.

[Zhu et al., 2003] Zhu, S., Setia, S., and Jajodia, S. (2003). LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *Proceedings of the 10th ACM conference on Computer and communications security*, CCS '03, pages 62–72, New York, NY, USA. ACM.

[Zigbee, 2010] Zigbee (2010). The Zigbee website.