Security and Performance Trade-off in KILAVI Wireless Sensor Network

MIKAEL SOINI, JUHA KUKKURAINEN, LAURI SYDÄNHEIMO
Rauma Research Unit, Department of Electronics
Tampere University of Technology
Kalliokatu 2, 26100 Rauma
FINLAND
mikael.soini@tut.fi

Abstract: Wireless sensor networks are typically used in different control and monitoring applications. In these applications resource-constrained sensor nodes gather information from the environment and possibly execute control commands based on the content of gathered information. The information transferred in a wireless sensor network can be very sensitive and therefore must be kept secret from outsiders. In this paper RC5 based encryption and CMAC authentication are used to obtain data confidentiality, freshness, replay protection, authentication, and integrity. These features enhance data security but can decrease sensor network operability because of added computation and communication load. This paper discusses the trade-offs between added security and sensor and sensor network performance. The focus is on how energy consumption and computation time are increased due to utilization of different security features and levels.

Key-Words: Energy consumption, KILAVI sensor network platform, performance tradeoffs, wireless sensor network security

1 Introduction

Wireless sensor networks (WSN) are based on physically small-sized sensor nodes exchanging mainly environment-related information with each other. WSNs have a very wide application area including home control, industrial sensing and environmental monitoring. Sensors typically have very limited power, memory, and processing resources and so interactions between sensors are limited to short distances and low data rates. Sensor node energy efficiency and sensor network data transfer reliability are the primary design parameters.

Security is one other vital aspect in WSN applications. The implementation of security policies is a complex and challenging issue because of resource-constrained nodes. Short transmission distances reduce some of the security threats, but there are risks, for example, related to spoofing, message altering and replaying, and flooding and wormhole attacks [1]. It is important therefore to consider security solutions that guarantee data authenticity, freshness, replay protection, integrity and confidentiality.

Security measures should not significantly affect WSN operation. In SPINS (Security Protocols for Sensor Networks) over 90% of security related energy consumption is caused by extra communication [2]. It is estimated that each extra bit transmitted consumes an equal amount of power to executing 800-1000 instructions in the processor [3]. The message size also affects reliability and scalability of the sensor network [4]. These support the idea that message size and the number of messages should be minimized in order to obtain low-power, simplicity and reliability.

There are methods that can be used to keep this sensitive information private. In asymmetric public key cryptosystems each node has a public key and a private key. The public key is published, while the private key is kept secret. Asymmetric public key cryptosystems such as the Diffie-Hellman key agreement or RSA signatures are typically too conservative in their security measures, adding too much complexity and protocol overhead to be usable in WSN solutions. The influence of public key cryptography to the lifetime of a sensor network node is evaluated in [15]. In symmetric cryptography the transmitter and the receiver of a message know and use the same secret key; the transmitter uses the secret key to encrypt the message, and the receiver uses the same secret key to decrypt the message. There are numerous key distribution mechanisms available for WSN applications, one presented in [16]. Symmetric solutions are therefore more suitable for resource-constrained sensor networks, though they need special solutions for security key pre-distribution. [5]

In this paper the well-known and well-understood RC5-based encryption and CMAC authentication is used to achieve security. RC5 is a symmetric block cipher that is used in counter mode (CTR). The same simple RC5 algorithm can be used for encryption and decryption. RC5 is reused in CMAC implementation. For these reasons, RC5 and CMAC are suitable for resource-constrained sensors.
This paper studies how these chosen security features affect sensor and sensor network performance. The focus is on security and performance trade-offs, especially in energy consumption and computation time. KILAVI platform [4, 10, 11, 12, 13] is used as a reference but all the results are easily exploited in other approaches as well. KILAVI is intended for low energy and low data rate device control

and monitoring in buildings. Compact KILAVI is comprehensive with regard to the different functions and devices needed to implement an operative building network, with dynamic network set-up and maintenance.

The rest of this paper is organized as follows. Section 2 concentrates on chosen encryption and authentication principles. Section 3 introduces the KILAVI platform and its security features. Section 4 presents the results on how much computation time the used security features take. Section 5 presents the results on how much time and energy the used security features consume in data transmission. Section 6 analyzes the obtained results. Finally, section 7 concludes the paper.

2 Security Issues

2.1 Encryption

RC5 (Rivest 5) is a block cipher type symmetric encryption algorithm that transforms a fixed-length block of unencrypted text into a block of encrypted text of the same length. This transformation takes place under the action of a user-provided secret key. Decryption is performed by applying the reverse transformation to the ciphertext block using the same secret key. [5]

RC5 is suitable for resource-constrained sensor nodes for the following reasons. RC5 is a simple and fast cipher using only common microprocessors operations; it has a low memory requirement making it suitable for sensor applications; the same lightweight algorithm can be used for both encryption and decryption; and heavy use of data-dependent rotations provides high security. RC5 block cipher has built-in parameter variability that provides flexibility at all levels of security and efficiency. Table 1 presents RC5 parameters. [6]

Table 1: RC5 parameters.

Parameters	Values
Word size (w)	16, 32, 64 bits
Block size ($2w$)	32, 64, 128 bits
The number of rounds (r)	0-255
Key length (b)	0-2040 bits

There are three basic routines in RC5: key expansion, encryption, and decryption. Encryption key expansion must be performed before the encryption can be initiated. In the key-expansion procedure, the user-provided secret key is expanded to fill a key table whose size depends on the number of rounds. The key table is then used in both encryption and decryption. The encryption routine is based on three primitive operations: integer addition, bitwise XOR- (exclusive-or) and variable rotation operations. Fig. 1 shows the first half of one RC5 encryption round. The encrypted plain text is divided into parts A and B. First, an XOR operation is performed on A and B. A bit rotation operation is executed with output of XOR and B. Finally, the output of the bit rotation operation is added to the extended encryption key (K). If the number of rounds used is 12, this operation is performed 24 times. [5, 6]

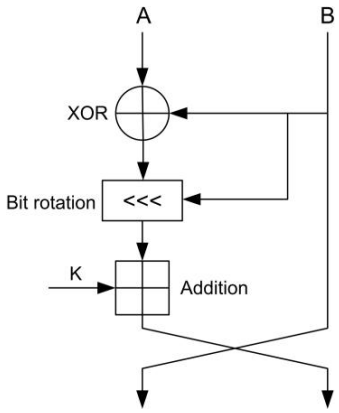

Figure 1: Basic operation of the RC5 algorithm.

In this work, the RC5 encryption algorithm is used in counter mode (CTR). RC5 guarantees a different character string each time and thus replay protection and data freshness qualities are obtained. CTR-based RC5 encryption does not increase the amount of transmitted bits or the number of sent messages in normal operation. In other words, the lengths of encrypted and plain messages are the same. In CTR mode the node maintains a counter that is increased by one after each successful transmission. The receiving node must be synchronized with the transmitter to be able to decrypt the messages. Fig. 2 presents the operation of encryption and decryption in CTR mode. K is the encryption key created in the key expansion procedure, C is the counter value and E is the RC5 encryption algorithm. CTR mode has the following benefits: high speed implementation, simplicity, arbitrary length of messages and a low rate of error propagation [7].

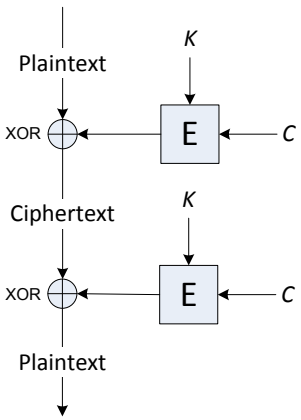

Figure 2: Encryption and decryption in CTR mode.

2.2 Authentication

Authentication may be used to check data integrity and authenticate communicating entities. A data integrity check makes sure that the message has not been altered by an adversary and an authentication check confirms the identity of the transmitter.

In this work a CMAC (Cipher-based Message Authentication Code) algorithm is used for authentication. CMAC is an authentication algorithm defined by the National Institute of Standards and Technology (NIST). Fig. 3 presents the operation principles of CMAC: a) message length is an integer multiple of block size, b) message length is not an integer multiple of block size.

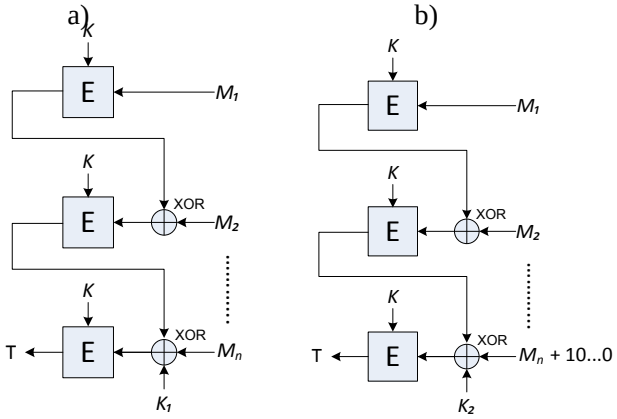

Figure 3: CMAC operation principles.

The CMAC algorithm depends on the choice of an underlying symmetric key block cipher, in this case RC5. The CMAC algorithm is thus a mode of operation of the block cipher (E). The CMAC key is the block cipher key (K); K is used to generate sub keys K_1 and K_2. The message M is divided into blocks where M_i is the block of the formatted message; M_n is the final block, possibly a partial block, of the formatted message. T is the authentication result. [8]

CMAC is a simple variant of the CBC-MAC (Cipher Block Chaining MAC). CMAC fixes security deficiencies of CBC-MAC [9]. Whereas the basic CBC-MAC is only secure for messages of one fixed length (and that length must be a multiple of the block size), CMAC is secure across messages of any bit length. Security of MAC is directly related to the length of MAC – a suitable value is 32 bits [14].

3 KILAVI Platform

3.1 Short introduction

Building control and monitoring is best performed with application specific sensor networking. KILAVI is an open manufacturer independent platform developed for low data rate and low-energy building control and monitoring applications. KILAVI defines a set of functions and messages which are needed to enable co-operative networking between different devices and the common means for data collection and device control tasks. Master controls the network operation and all network nodes are alike. Nodes operate either in sensing mode (Sensor nodes) or forwarding mode (Intermediate nodes) depending, for example, on battery state. The basis for an operational building control platform: centralized hierarchical architecture to enable resource concentration, compact messages to obtain robust networking, 433MHz operating frequency to gain necessary operating distances with low-power, a multihop communication to enable large scale networks and low-power sensors, and hybrid flooding to provide low overhead network management. KILAVI network architecture is shown in Fig. 4. Performance evaluation has shown that this platform is energy-efficient, reliable and low-power. [4, 10, 11, 12, 13]

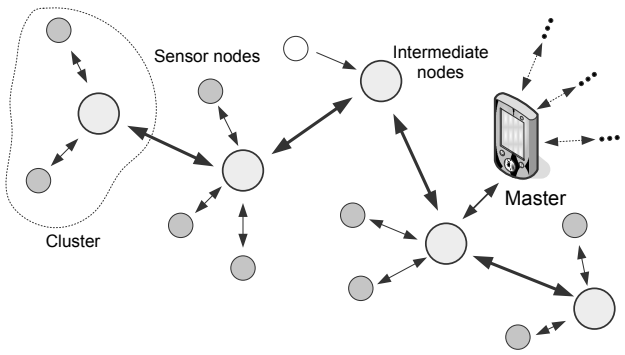

Figure 4: KILAVI network architecture.

3.2 Security in KILAVI

The dominating traffic pattern in WSN is many-to-one where sensors communicate with a central unit. Thus, centralized security architecture and symmetrical end-to-end keys between master and sensors are a natural choice. In KILAVI, the central node manages, delivers and updates keys with every node. Nodes have one authentication and one encryption key to enable secure communication with the central node. Nodes in forwarding mode do not interpret messages excluding the information related to forwarding or data storing. This is simple from the sensor node perspective and a low overhead from the networking perspective. [11, 13]

KILAVI uses RC5 encryption in counter mode and the CMAC authentication presented in Section 2. More KILAVI security features including security levels, secure device registration, key exchanging procedure and counter synchronization are presented in [10]. Table 2 summarizes KILAVI security parameters.

Table 2: KILAVI security parameters.

KILAVI security	Parameters
Word size (*w*)	16 bits
Block size (*2w*)	32 bits
The number of rounds (*r*)	12
Key length (*b*)	128 bits
Counter length	32 bits
MAC length	32 bits

Further, in KILAVI the increment due to the MAC field is only 32 bits in KILAVI if security features are used. These very small packet length increments do not significantly affect sensor network reliability as shown in [4].

Memory space is usually very limited in sensor network nodes. The implemented solution with chosen parameters for encryption and authentication is lightweight: RC5 code size is 716 bytes and MAC size is 366 bytes in length. The total memory size is therefore 1082 bytes.

4 Effect of Enhanced Security on Computation

Section 2 presented the principles of chosen security measures. The encryption and authentication functions presented are implemented on an 8-bit Atmega644PV-10PU microprocessor. This section presents results, based on practical measurements, on how much time is consumed by key expansion, encryption and authentication computation operations with different security-related parameters. Section 5 also considers enhanced security from the data transmission perspective.

4.1 Key Expansion

The first routine of RC5 is key expansion. Encryption key expansion is executed before actual encryption and its operation was presented in section 2. Fig. 5 shows how much time the encryption key expansion routine consumes with different word lengths as a function of the number of rounds.

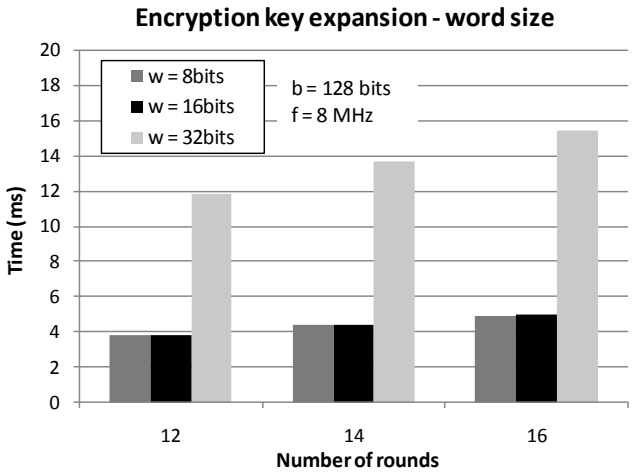

Figure 5: Encryption key expansion time with different word lengths as a function of the number of rounds (clock speed is 8MHz).

4.2 Encryption and decryption

After the key expansion, encryption and decryption procedures can be initiated in a manner presented in section 2. Fig. 6 and Fig. 7 present how much time it takes to encrypt variable size message with different word lengths and with different numbers of encryption rounds.

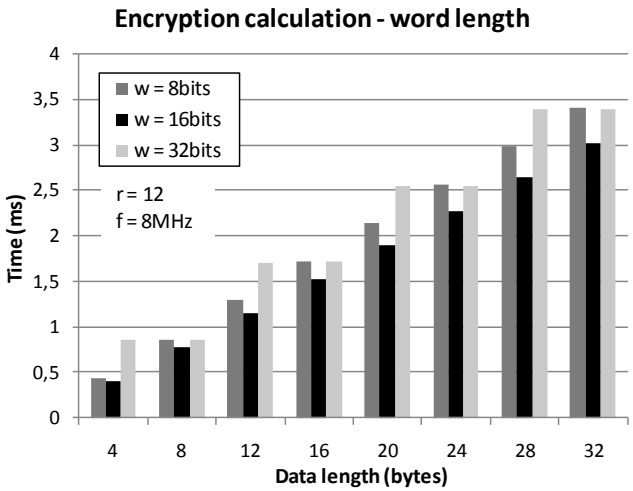

Figure 6: Encryption calculation time as a function of word length (clock speed is 8MHz and amount of encryption rounds is 12).

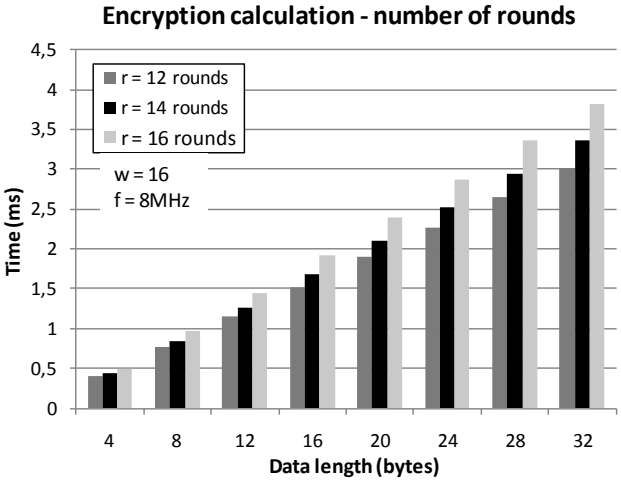

Figure 7: Encryption calculation time as a function of numbers of rounds (clock speed is 8MHz and word length is 16 bits).

4.3 Authentication

Authentication is used to check data integrity and confirm the identity of a sender. Fig. 8 and Fig. 9 present MAC calculation times for variable message sizes with different word lengths when the microprocessor clock speed is set to 1 MHz and 8 MHz, respectively.

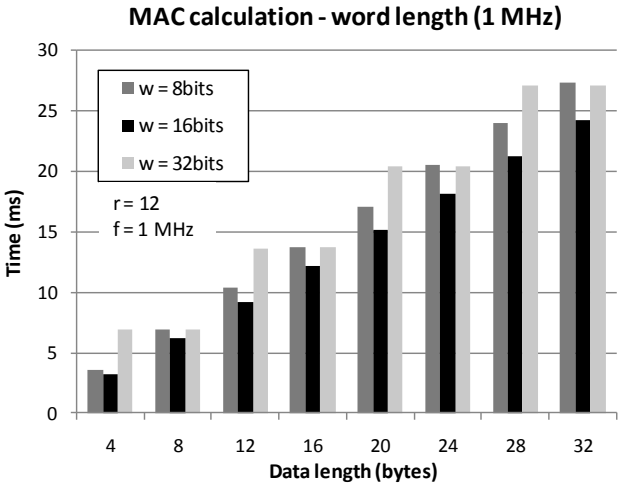

Figure 8: MAC calculation time as a function of word length (clock speed is 1MHz and amount of encryption rounds is 12).

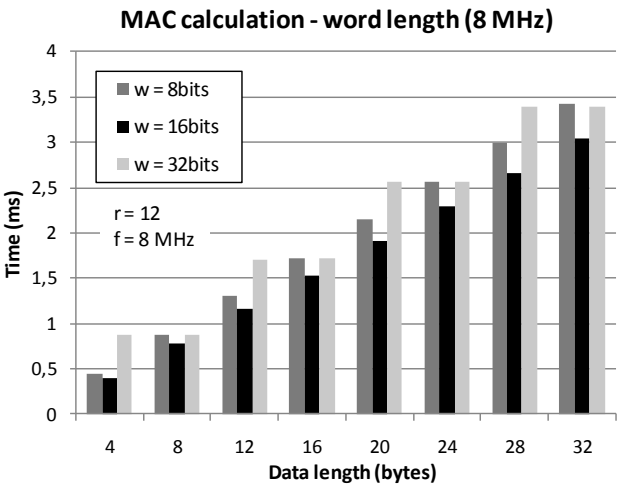

Figure 9: MAC calculation time as a function of word length (clock speed is set to 8MHz and amount of encryption rounds is 12).

5 Effect of Enhanced Security on Communication

This section considers how selected security features affect sensor data transmission including the computational parts presented in section 4. Computation and transmission time, and sensor energy consumption are studied. Message lengths used are 4-32 bytes which are typical in KILAVI.

5.1 Computation and Transmission Time

Fig. 10 and Fig. 11 show how time is distributed in typical message transmission (with nRF905) between encryption (encryption calculation), message authentication (MAC calculation, and SPI and RF transmission of 32-bit code), and actual data payload (SPI and RF transmission).

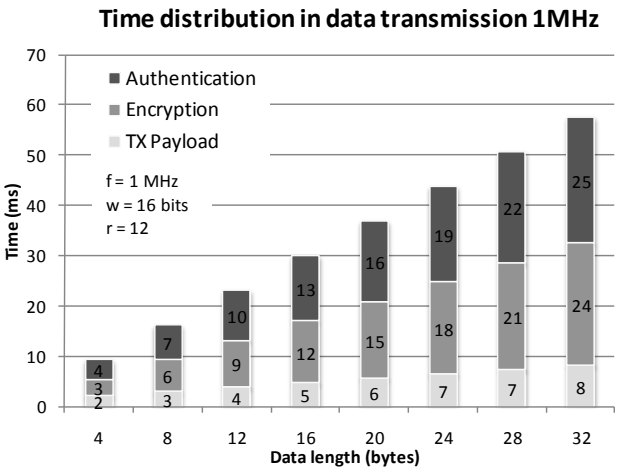

Figure 10: Time distribution in data transmission between MAC, encryption and data (clock speed is 1MHz, amount of encryption rounds is 12 and word length is 16 bits).

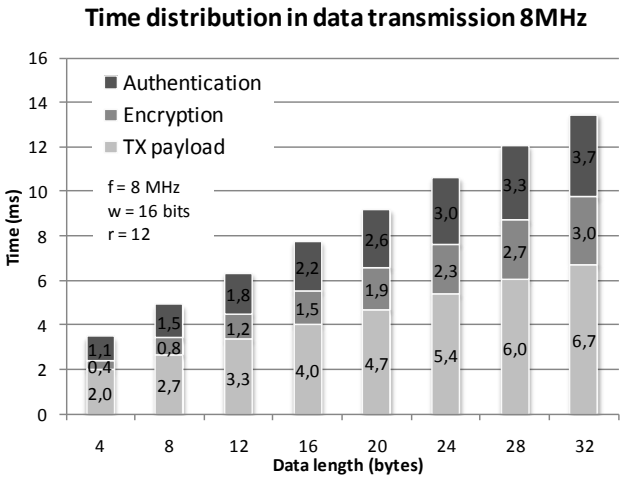

Figure 11: Time distribution in data transmission between MAC, encryption and data (clock speed is 8MHz, amount of encryption rounds is 12 and word length is 16 bits).

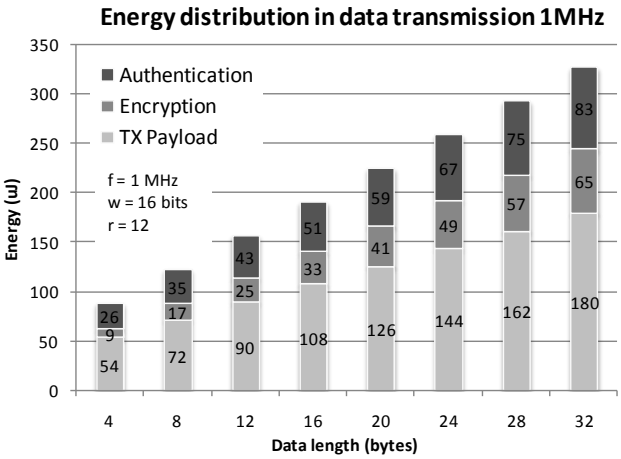

Figure 12: Energy distribution in data transmission between MAC, encryption and data (clock speed is 1MHz, amount of encryption rounds is 12 and word length is 16 bits)

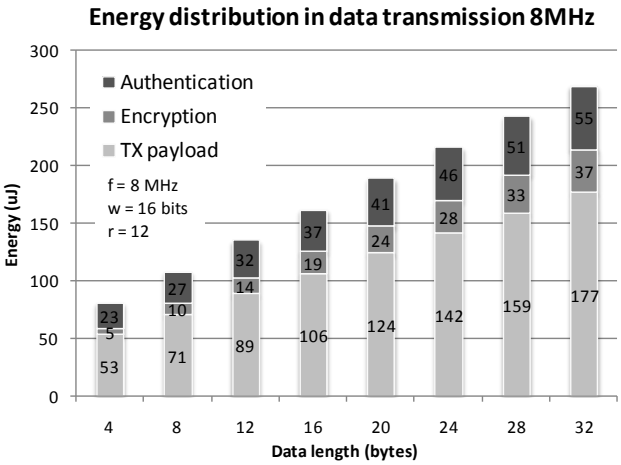

Figure 13: Energy distribution in data transmission between MAC, encryption and data (clock speed is 8MHz, amount of encryption rounds is 12 and word length is 16 bits).

5.2 Sensor Node Energy Consumption

Table 3 presents current consumption values that are used in following calculations. Current consumption values are measured with prototype sensors (Atmega644PV-10PU microprocessor and Nordic Semiconductors nRF905 radio used).

Table 3: Current consumption of prototype nodes.

Operation	RC5/MAC	SPI	Radio
MCU	active	active	power save
SPI	non-active	active	non-active
Radio	power down	power down	active
I@1MHz	0,9mA	1,1mA	9mA
I@8MHz	4,15mA	4,3mA	9mA

By using these measured current consumption (*I*) values, measured voltage (*U*) of 2,989V and previous measured time values (*t*), energy consumption (*E*) can be calculated with (1)

$$E = P \cdot t = U \cdot I \cdot t \qquad (1)$$

Fig. 12 and Fig. 13 present energy distribution in typical message transmission between encryption, message authentication, and actual data payload.

6 Analysis

6.1 Encryption and MAC calculations

From Fig. 8 and Fig. 9 it can be seen that crystal oscillator frequency is inversely proportional to computation time. Therefore, increasing the oscillator frequency from 1MHz to 8MHz decreases the computation times (t) 87.5%. At the same time, energy consumption decreases by about 42% independent of message length. Energy consumption (*E=P*t*) does not decrease by the same amount because power consumption is about 4-times higher with an 8MHz oscillator speed. The *number of rounds* affects on security strength. From Fig. 7 it can be observed that additional rounds (*n*) increase the computation time by around *n**5%. Optimal *word length* can save 10% to 25% of computation time depending on message length (see Fig. 6 and Fig. 8). In

the encryption key expansion procedure (see Fig. 5), computation time can decrease by as much as 68% if optimal *word length* is used (in this case 16-bits).

6.2 Message transmission

Here the results are considered from a message transmission perspective. This means that in addition to security calculations there is also a message transmission to be considered. From Fig. 10 and Fig. 11 it can be seen that by increasing the oscillator frequency from 1MHz to 8MHz, the computation time decreases by 63% to 77% depending on message length (4B to 32B). At the same time energy consumption decreases by 8.7% to 18%. The data transmission time does not vary as a function of the oscillator frequency (excluding SPI transmission from microprocessor to radio).

6.3 Security versus No-Security: Transmission of a Single Message

In this subchapter, 8MHz oscillator frequency is used for the reasons given above. From Fig. 11, it can be seen that transmission of a single message without security takes 2.0ms to 6.7ms depending on message length (4B to 32B). With the security features operating this time increases by 74% to 100% (3.5ms to 13.4ms) again depending on message length. From Fig. 13, it can be seen that from the energy perspective transmission of a single message without security consumes 53 to 177μJ depending on message length (4B to 32B). With the security features this energy consumption increases around by 52% with all message lengths tested (4B to 32B).

6.4 Security versus No-Security: Sensor Network Operation

In this subchapter, the above results are mirrored in a real-world KILAVI-based data collection system where a sensor wakes up and transmits measured data at specific intervals and stays in sleep mode (here, I_{SLEEP} is 10μA) most of the time. If a sensor transmits messages at a rate of 1 message per second, and security features are used then the total duration of one operation cycle increases by 0.15% to 0.67% depending on message length (4B to 32B) and the energy consumption increases 33% to 45% depending on message length (4B to 32B). If a sensor transmits messages at a rate of 1 message per minute, then the total time taken for one operation cycle increases by 0.003% to 0.011% depending on message length (4B to 32B) if security features are used and the increase in energy consumption is 1.5% to 4.7% depending on message length (4B to 32B). With longer transmission intervals increase in energy consumption cause by security features is negligible as seen in Fig. 14.

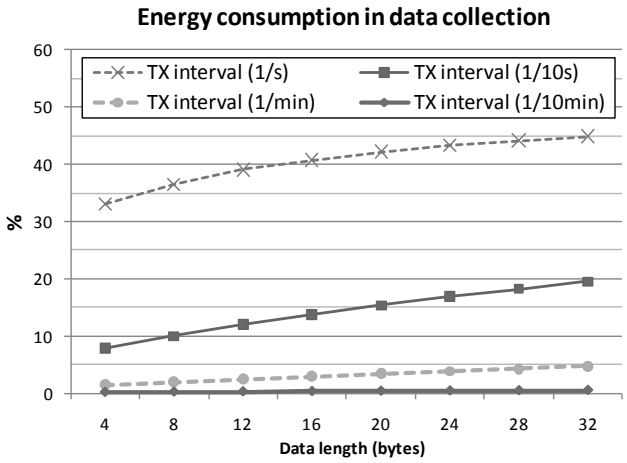

Figure 14: The increase in energy consumption caused by security feature utilization in a data collection system where status information is sent periodically (TX interval).

7 Conclusions

KILAVI uses Rivest's nominal version (RC5-32/12/16) for encryption and decryption, and CMAC for authentication. In this paper, the effects of these security features on WSN operation were studied.

It can be concluded that high oscillator frequencies should be used if security features are implemented in WSN solutions. The fast computation speed causes shorter buffer lengths on sensor nodes and shorter end-to-end delays in multihop communication. It can be seen from the results that the increase in computation time caused by added security is negligible in sensor networks when using typical transmission intervals. Therefore KILAVI network nodes in forwarding mode do not become congested due to increased computation and delay. Further, energy-scarce nodes in sensing mode typically operate with large transmission intervals and therefore the energy consumption increase caused by added security is tolerable and does not substantially affect sensor lifetime.

KILAVI uses very short messages and in building control applications messages are sent quite rarely and therefore the utilization of security features presented neither causes congestion in network operation nor increases sensor node energy consumption significantly.

References:

[1] C. Karlof and D. Wagner, Secure Routing in Wireless Sensor Networks: Attacks and countermeasures," *Proc. IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 113-127, May 2003.

[2] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, SPINS: Security Protocols for Sensor Networks, *Proc. 7th Annual ACM International Conference on Mobile Computing and Networks*, Rome, Italy, pp. 189-199, July 2001.

[3] Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister, System architecture directions for networked sensors, *Proc. 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, Cambridge, MA, USA, pp. 93-104, November 2000.

[4] M. Soini, L. Sydanheimo, and M. Kivikoski, Reliability and Scalability of Kilavi Building Control Platform, *Proc. International Symposium of Consumer Electronics*, Dallas, TX, USA, June 2007.

[5] RSA Laboratories, RSA Laboratories: Frequently Asked Questions About Today's Cryptography, *RSA Security* Inc., Version 4.1, 2000.

[6] R. L. Rivest, The RC5 Encryption Algorithm, *Proc. 2nd International Workshop on Fast Software Encryption*, Leuven, Belgium, pp. 86–96, December 1994.

[7] H. Lipmaa, P. Rogaway, and D. Wagner, Comments to NIST concerning AES Modes of Operations: CTR-Mode Encryption, *Symmetric Key Block Cipher Modes of Operation Workshop*, Baltimore, MD, USA, October 2000.

[8] National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, *National Institute of Standards and Technology*, Special Publication 800-38B, May 2005.

[9] A. Menezes, P. van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, *CRC Press, Inc.*, Boca Raton, FL, USA, 1996.

[10] H. Sikkila, M. Soini, L. Sydanheimo, and M. Kivikoski, KILAVI Wireless Communication Protocol for the Building Environment – Security Issues, *Proc. 10th IEEE International Symposium of Consumer Electronics*, St. Petersburg, Russia, June-July 2006.

[11] P. Oksa, M. Soini, L. Sydänheimo, and M. Kivikoski, Kilavi platform for wireless building automation, *Energy and Buildings Journal*, Vol. 40, No. 9, pp. 1721–1730, 2008.

[12] M. Soini, H. Sikkila, P. Oksa, L. Sydanheimo, and M. Kivikoski, KILAVI Wireless Communication Protocol for the Building Environment – Network Issues, *Proc. 10th International Symposium of Consumer Electronics*, St. Petersburg, Russia, June-July 2006.

[13] M. Soini, J. van Greunen, J. Rabaey, and L. Sydanheimo, Beyond Sensor Networks: ZUMA Middleware, *Proc. IEEE Wireless Communication and Networking Conference*, Hong Kong, China, March 2007.

[14] C. Karlof, N. Sastry, and D. Wagner, TinySec: A Link Layer Security Architecture for Wireless Sensor Networks, *Proc. 2nd ACM Conference on Embedded Networked Sensor Systems*, November 2004.

[15] K. Piotrowski, P. Langendoerfer, and Steffen Peter, How public key cryptography influences wireless sensor node lifetime, *Proc. 4th ACM workshop on Security of ad hoc and sensor networks*, October 2006.

[16] B. Arazi, I. Elhanany, O. Arazi, and H. Qi, Revisiting Public-Key Cryptography for Wireless Sensor Networks, *IEEE Computer Magazine*, Vol. 38, No. 11, November 2005.