Some Security Considerations over Contiki-based Sensor Network

Yan Yan

December 10, 2015

Contents

Chapter 1

Introduction

This paper discusses two security measurements, namely Link Layer Security (LLSEC) and Datagram TLS (DTLS), within Contiki OS.

1.1 Related Work

[1] discusses some security concerns in 802.15.4. LLSEC[2] is the implementation of 802.15.4 security in Contiki.

tinydtls[3] is the implementation of DTLS we used in DTLS related experiments.

1.2 Experiment Setup

All experiments are done within the Cooja simulator.

The setup is as described in Figure 1.1.

- **Adversary** is the malicious party that tries to recover information from the encrypted traffic.
- **Border Router**, or BR, is a device that connects the adversary to the sensor network. However, **BR is not allowed when LLSEC is enabled** as the adversary does not have the key and hence cannot connect into the network.
- **Sniffer** is a device that passively captures all traffics in the sensor network.
- **Target** and **Nodes** are sensors deployed in the sensor network. They communicates to each other through encrypted channels.
- **Sensor Network** discussed in this paper is a 6LowPAN network.

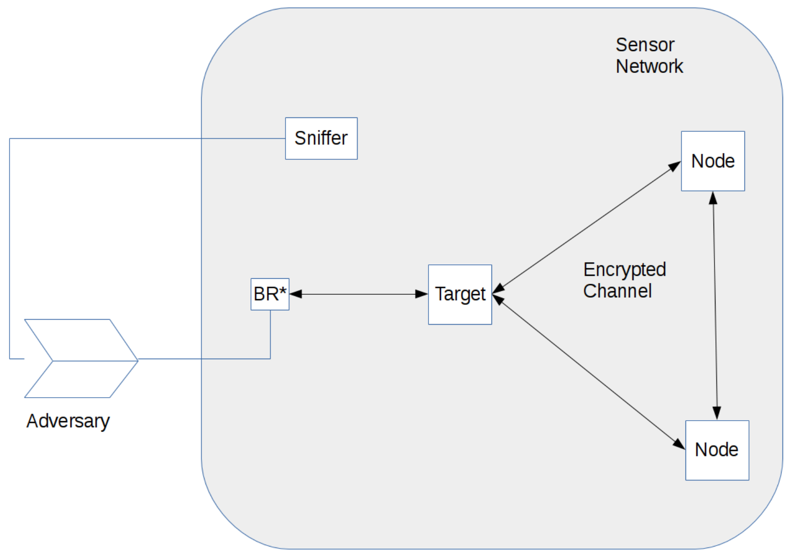

Figure 1.1: Experiment setup

1.3 Adversary Power

The powers assumed in the experiments are considered to be practical in real life.

When LLSEC is enabled, all traffic, including RPL[1] messages, are encrypted; therefore no external nodes can connect to the network. The only power for the adversary is to passively sniff all the traffic.

In other cases where LLSEC is disabled, the adversary will be enabled to join the sensor network through a BR and hence is also capable to send ICMP messages to the target(s).

1.4 Types of Packets

We simply categorise the packets into two types:

- **Network Management Packets**: These are the packets generated by the protocols those maintains the network, such as MAC ACKs, RPL messages or ICMP messages.
- **Data Packets**: These are those packets generated by the applications running on the nodes., such as a CoAP packet.

[1] Routing Procol for Low-power and Lossy Networks

This is only a subjective rough categorisation and may not be precise. For example an TCP data packet may also serves as an ACK, or DTLS handshake packets could fall into both categories. However, we ignore this ambiguity as it is not our focus.

Chapter 2

Link Layer Security

Link Layer Security, or LLSEC, is a security measure that implements cryptography just above the physical layer.

Introducing cryptography at a lower level has several benefits. Firstly, more data being encrypted reduces the observable packet features to an adversary, such as SRC[1] and DST[2] field in the IP header which are very likely to be exploited by the adversary. Secondly, authentication at lower level also prevents an active adversary from joining the network which therefore weakens his power.

Imposing cryptography at a lower level also brings more challenge to the design of sensor network architecture. The first problem is its overhead. Even for a node that only tries to retransmits the packet to its next hop, it must decrypt the whole packet to extract its routing information, and then re-encrypt it before retransmission. This is particularly problematic in a mesh wireless sensor network as it could potentially lead to performance and energy consumption problems. Key management is also challenging due to the lossy and power optimised nature of wireless sensor network.

It is also noticeable that some packet features are not hidden even with LLSEC enabled, such as packet length, timing information and part of the MAC header.

2.1 802.15.4 Security: *noncoresec*

noncoresec[2] is the current implementation of LLSEC in Contiki. It corresponds to the AES_CCM_16 ciphersuite in 802.15.4 standard. This section briefly describes how it works.

- **Key Management**: All nodes share a network wide AES key for both encryption and authentication. The key is hardcoded during the setup stage.

[1] Source Address
[2] Destination Address

Flags(1)	Addresses(8)	Frame Counter(4)	Security Level(1)	Block Counter(2)

Table 2.1: IV of 802.15.4 Frame with Security

- **AEAD**[3]: *noncoresec* implements AES_CCM_16 [4] as described in 802.15.4[4]. CCM mode turns AES into a stream cipher. The same key is used for both encryption and authentication.
- **Initial Vector (IV, or nonce)**: The IV for each packet is constructed from certain fields of unencrypted MAC frame header and therefore is public.

An adversary without the knowledge cannot join the sensor network as he cannot sent out a valid RPL message.

2.2 Weak IV

One problem within the *noncoresec* implementation is the low variance of IV. The IV is a 16 byte bit-string constitutes of the following fields(Table 2.1):

- **Flags (1 byte)**: This field contains part of the MAC frame header. It is identical to most (basically all) of the data packets.
- **Source Address (8 bytes)**: This is mapped from the source address field of the frame.
- **Frame Counter (4 bytes)**: This field increases by 1 for each frame sent.
- **Security Level (1 byte)**: This field indicates which ciphersuite to be used for this frame. In the case of AES_CCM_16, this is constantly 0x7.
- **Block Counter (2 bytes)**: This field begins from 0x0 and increases by 0x1 for each block in CCM mode. The block length for AES-128 is 16 bytes. The 2 bytes counter is sufficient as it supports up to 2^{32} bytes of data whereas the minimum MTU[5] required by 6lowPAN standard[5] is 127 bytes.

In the current *noncresec* implementation, **Flags** and **Security Level** are constant. **Block Counter** always begins from 0x0 and the **Source Address** is also constant for a specific device. Such design leaves the 4 bytes **Frame Counter** the only field that is variable. This indicates that only $2^3 2$ messages are allowed which is cryptographically considered to be inappropriate.

[3] Authenticated Encryption with Associated Data
[4] CCM mode of AES-128 with 16 bytes MAC
[5] Maximum Transmit Unit, simply speaking this is the maximum length of a packet.

Figure 2.1: Captured packets with *noncoresec* enabled

2.2.1 Reset Problem

The low variance of IV leads to a plaintext leakage problem which only requires the adversary to reboot the target node.

The idea is that rebooting the device resets the **Frame Counter** to 0x0; hence once a pair of packets with same **Frame Counter** is found, the difference of their plaintext can be computed by their ciphertext:

$$\Delta p = c_1 \oplus c_2$$

where Δp is the difference of plaintexts. c_1 and c_2 are their ciphertext respectively.

Example 2.2.1. Figure 2.1 demonstrates some packet captured[6] with *noncoresec* enabled. These packets are captured with a sensor broadcasting a 4 byte integer with left being $[00000000]_{16}$ and right $[12345678]_{16}$. The marked are the corresponding ciphertexts which are $[00127401]_{16}$ and $[12262279]_{16}$ respectively.

[6]The duplicated packets are caused by the retransmission of ContikiMAC[6].

As we can see, the difference of ciphertext is exactly the difference of plaintext:

$$\Delta p = [00127401]_{16} \oplus [12262269]_{16} = [12345678]_{16}$$

2.3 Distinctive packet length for RPL packets

Some RPL packets are shorter than the minimum length of data packets which can be used to distinguish the packets.

2.4 Performance issue

The header overhead with LLSEC enabled is 20 bytes which is relatively a large overhead comparing to the 127 bytes MTU requirement of 6LowPAN standard[5].

Chapter 3

DTLS

3.1 Conflicting MTU between DTLS and 6lowPAN

The abandoned CoDTLS.

3.2 Overloading DTLS with LLSEC

Chapter 4

Application Detection

4.1 Packet Length

4.2 Response Time

4.3 Pingload: Ping side-channel for Payload

Bibliography

[1] Naveen Sastry and David Wagner. “Security Considerations for IEEE 802.15.4 Networks”. In: *Proceedings of the 3rd ACM Workshop on Wireless Security*. WiSe ’04. Philadelphia, PA, USA: ACM, 2004, pp. 32–42. ISBN: 1-58113-925-X. DOI: 10.1145/1023646.1023654. URL: http://doi.acm.org/10.1145/1023646.1023654.

[2] URL: https://github.com/kkrentz/contiki/wiki.

[3] URL: http://sourceforge.net/projects/tinydtls/.

[4] *IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*. Tech. rep. 2006, 0_1–305. DOI: 10.1109/ieeestd.2006.232110. URL: http://dx.doi.org/10.1109/ieeestd.2006.232110.

[5] G. Montenegro et al. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. RFC 4944 (Proposed Standard). Updated by RFCs 6282, 6775. Internet Engineering Task Force, Sept. 2007. URL: http://www.ietf.org/rfc/rfc4944.txt.

[6] Adam Dunkels. *The ContikiMAC Radio Duty Cycling Protocol*. 2011.