APPLICATION NOTE 513

Robust EnOcean Networks

COMMUNICATION PATTERNS AND DETECTIONS OF DEVIATIONS

Table of contents

1. INTRODUCTION

Robustness of network is one of the reliability parameters of any network. We consider robustness in this application note as actions to deal with critical situations as external noise, attacks and malfunction with the target to ensure highest link quality and reliability. In general, we can see two major aspects of robustness:

- Actions to prevent critical situations – Here we conclude constraints for the network, so we prevent any critical actions to happen. This aspect goes more into the directions of protocol and product design. EnOcean Radio protocol has defined many measures in his design to prevent critical situations as listen before talk, transmission timing or redundant retransmission, for details please look up the specification itself [8].
- Actions to detect critical situations – Here we define processes to detect critical situations in a network. Detecting these situations is as important as preventing them, because after discovering a critical situation correcting or protecting actions can be taken. Detection is possible based on communication analysis and evaluation.

Preventing critical situations is situated in the design of the EnOcean Radio protocol itself and in network planning of future installations. For this purpose, we provide other Application notes, as for example AN001 EnOcean Wireless Systems - Installation Notes [10]. In this Application note we will focus on the actions to detect critical situations.

The described detection of critical situations is based on operations with network data and statics – gathering, analysing and reasoning. Network data are represented as communication log of sent and received telegrams. These tasks combined together are referenced also as data mining.

The essential part of detection of critical situations is defining communication patterns and detecting deviations based on network data evaluation. These deviations directly express a critical situation. In this App note we focus on the detection of deviations on the receiver.

Communication pattern is a predefined schedule of telegram transmission. The pattern is deduced from the transmitting application and its characteristics, e.g. the communication pattern on a current meter can be a static period of 10 seconds, this means every 10 seconds a telegram is expected.

When the communication deviates from the defined communication pattern, there is to assume:

- Device is not operating
  - Broken
  - Out of Range
  - Out of Energy
- Device is operating, but signals are not received:
  - Radio Link Budget is influenced by an external factor, not mean to harm
  - Denial of Service (DoS)
  - Delay Attack

Explaining about what the exact reason for the deviations is not trivial and highly dependent on the used application. Therefore we focus in this application note on the pure detection of critical and harmful situations – DoS and Delay Attack. The handling of these events is forwarded to the application.

The aim is to detect possible DoS and Delay attacks, therefore we assume the communication is protected with enhanced security features [9]. Without enhanced security features, the communication is not 100% protected against other attacks, like for example message forging or replay attacks. Please also consider that detecting DoS and Delay Attacks but not using enhanced security features would result in a non-consistent protection. Therefore, we consider this detection as an extension of the current Enhanced Security EnOcean Specification [9].

1.1. Definitions

Term / Abbr.	Description
µC	Microcontroller (external)
AES	Advanced Encryption Standard
API	Application Programming Interface
APP	Application
ASK	Amplitude Shift Keying
CBC	Cipher Block Chaining
CMAC	Cipher Based Message Authentication Code
CRC	Cyclic Redundancy Codes
DATA	Payload of a radio telegram
Device	Customer end-device with an integrated EnOcean radio module
DoS	Denial of service
EEP	EnOcean Equipment Profile
EHW	Energy Harvested Wireless protocol
ERP	EnOcean Radio Protocol (ERP1 = Version 1, ERP2 = Version 2)
ESP3	EnOcean Serial Protocol V3
FSK	Frequency Shift Keying
Gateway	Module with a bidirectional serial communication connected to a HOST
GP	Generic Profiles
ID	Unique module identification number
KEY	Specific parameter used to encrypt / decrypt / transform DATA
MAC	Message Authentication Code
MSB	Most Significant Byte
PSK	Pre-shared Key
PTM	Pushbutton Transmitter Module
RLC	Rolling Code
R-ORG	Message parameter identifying the message type
SLF	Security Level Format specifying which security parameters are used
TXID	ID of a transmitter
VAES	Variable AES

1.2. References

[1] EnOcean Link - http://www.enocean.com/en/enocean-software/enocean-link/

[2] EEP Specification - http://www.enocean-alliance.org/eep/

[3] GP Specification - http://www.enocean-alliance.org/de/enocean_standard/

[4] EnOcean Radio Protocol 1 - http://www.enocean.com/fileadmin/redaktion/pdf/tec_docs/EnOceanRadioProtocol.pdf

[5] ESP 3 - http://www.enocean.com/esp

[6] Gateway Controller - http://www.enocean.com/en/enocean-software/gateway-controller/

[7] Dolphin V4 Gateway Controller http://www.enocean.com/en/enocean-software/

[8] EnOcean Wireless Standard http://www.enocean-alliance.org/en/enocean_standard/

[9] Security of EnOcean Radio Networks, http://www.enocean.com/en/security-specification/

[10] EnOcean Application Notes, http://www.enocean.com/en/application-notes/

1.3. Revision History

No	Major Changes
1.0.	First version

2. APPLICATION CONSTRAINTS OF A GATEWAY APPLICATION

To be able to define communication patterns on a receiver / gateway application, first we must define the application itself and its possibilities.

In the following chapter the use case of a generic receiver application is defined.

2.1. Use Case description

The critical situation detection is predestined to take part in a Gateway application (external controller) of an EnOcean network because:

- It has enough resources to execute needed computing tasks and store data;
- It has connection to interfaces and forward a security thread event to a corresponding handler (e.g. forward message to phone, visual signals)

The typical architecture of a Gateway Application is listed below:

Figure 1 Gateway receiver Application

The EnOcean Gateway (e.g. TCM 310) receives all EnOcean Radio signals and forwards those to the Customer Application placed on a secondary µC. Here, EnOcean Link (our Protocol Stack Implementation) [1] or customers security implementation takes the still encoded information and process it. The encoding and decoding tasks are in both case all executed in the customer µC. Below are listed tasks of a generic customer application which handles EnOcean UART signals forwarded by an EnOcean Gateway (e.g. TCM 310) in an overview.

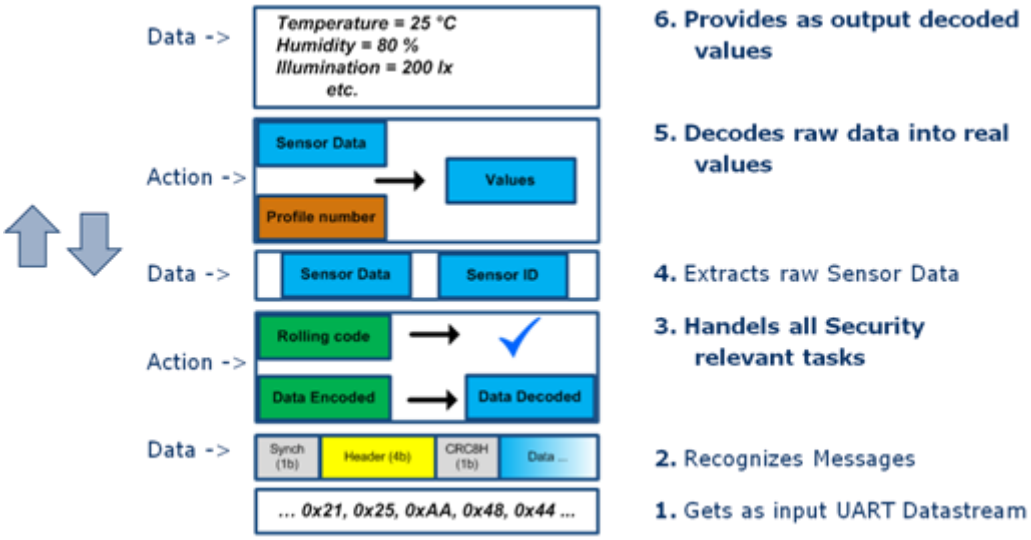

Figure 2 Generic reciever taks

The security relevant tasks are an essential part of the process. Security related tasks as also the DoS & Delay attack detection are handled in Step 3. The processing unit has all knowledge of the EnOcean network and the connected devices (especially COUNTERs values, KEYs, security format and application type).

2.2. Defining communication patterns on Receiver Applications

Key essence is to define communication patterns and detect suspicious behavior. Communication patterns define the schedule of message transmission of a device (e.g. when a message was expected to be received). Basically, it is a prediction of when the next message should be received.

The communication pattern should not consider the actual application payload and is situated with the encryption algorithms on the Data Link layer. Suspicious behavior is a defined deviation from the communication pattern and given tolerances. Based on detected suspicious behavior, a thread event is raised to the handler.

To define a communication pattern, the following information of the transmitting application must be known:

- Transmission behaviour:
  - Event triggered only
  - Periodical transmission only
  - Periodical & Event triggered transmission behavior
- Device transmission interval (periodicity)
- Dynamics of device transmission interval – events which can change the transmission interval

En example of a communication pattern definition of a device, which is a window contact, could look like this.

- Transmission behaviour: Periodical & Event triggered transmission behaviour
- Communication Interval: 15 minutes

- Dynamics:
  - If an event triggered transmission was realized, then the communication period was reset from the beginning.

For pattern definition, mostly only devices with a periodical transmission are suitable. Pure event based transmissions are hard to predict on the gateway application and therefore out of scope of this application.

The communication pattern must be defined for every device separately. During the definition manufacturer specific aspects must be considered and included in the definition.

2.2.1. Detecting deviations from defined communication patterns

The following deviation from the communication pattern can be detected – critical situation:

1. Message was not received in the defined time
2. Message was received in an undefined time & device is only periodical based - not event triggered
3. Received telegrams per defined time interval ratio is higher than defined
4. Count of unsuccessfully performed CMAC validations has reached a defined level

3. DETECTION OF DOS

We define these types of DoS attacks, which can be detected on the defined gateway application:

A. Intruder makes the gateway "not receiving" a telegram (by e.g. jamming the signal, noise generation etc.)
B. Intruder forges telegrams with correct telegram format and an EnOcean Sender ID to:
   - Desynchronize the COUNTER values on the receiver and make so the application non-functional
   - Brute force attack the system – try to guess the COUNTER state

Critical situation (see 2.2.1) 1 is the indicator of a possible DoS attack A.

Critical situation (see 2.2.1) 2, 3 and 4 is the indicator of a possible DoS attack B.

4. DETECTION OF DELAY ATTACKS

In our application we define Delay Attacks as a process of:

1. Intercepting a telegram X from a device A.
2. At the same time, making the gateway B „not receiving" the telegram X.
3. Transmitting the telegram X unchanged at a later time.
   Constraint: Step 3 must be executed before transmission of any further telegram from device A, or make the gateway B „not receiving" any further telegram.

If any further telegram was be transmitted by device A before step. 3 would be performed, then the COUNTER on gateway B for device A would be updated to the next following value and the telegram X would lose its validity.

The detection algorithm of Delay Attacks is based on these steps:

ASSUMPTION:
DoS attack A is suspected (see 3). The expected telegram would be having the COUNTER value M.

DETECTION:

1. A message from this particular device was received later (regardless of the actual pattern).
2. The received telegram was carrying the previous expected value M.

If the received telegram would be having the counter M+1, then it is to be assumed that the previous telegram with value M just got lost.

If multiple DoS attacks A were detected continuously e.g. more transmission intervals are missed, then all missed COUNTER values must be considered in the evaluation step.

5. RAISING THREAD EVENTS

The detection of possible DoS or Delay Attacks, the gateway application should raise a thread event and forward it to the defined handler. The handler is not part of this specification and is free for the gateway manufacturer how he decides to define it. The handler can have:

1. An informative character - the human user is informed by this event
2. A restrictive character – any other functionality of application is locked
3. A forwarding character – event information is used as input for more complex processing which is beyond the scope of this specification

6. CONCLUSION

The description above defines a possible way of detecting a DoS and Delay Attack of the specified kind in an EnOcean network with use of communication patterns. Implementing this algorithm in the gateway with combined EnOcean Enhanced security features will raise the security level of any receiving application and make it harder for intruders to perform attacks.