**IEEE P802.15**

**Wireless Personal Area Networks**

|  |  |
| --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) |
| Title | **Supplementary text for secret key agreement protocol for IEEE 802.15.8 PAC** |
| Date Submitted | [November, 2015] |
| Source | [address] | Voice: [ ]Fax: [ ]E-mail: [bjkwak@etri.re.kr]1, [ssyun@kaist.ac.kr]2 |
| Re: | P802.15.8 D0.16.0 |
| Abstract | Text proposal for secret key agreement protocol for IEEE 802.15.8 PAC. |
| Purpose | Approval |
| Notice | This document has been prepared to assist the IEEE P802.15. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor acknowledges and accepts that this contribution becomes the property of IEEE and may be made publicly available by P802.15. |

***Note to Editor: Black texts represent the existing text in P802.15.8 PAC draft, and the proposed text changes are in blue.***

**15. Security**

**15.1 Secret key agreement using physical layer features**

IEEE 802.15.8 provides a secret key agreement protocol using physical layer features, by taking advantage of channel reciprocity and sequential key distillation. The secret key agreement protocol allows a pair of legitimate PDs to remotely share a secret key without resorting to a key management infrastructure. Figure 100 illustrates the three phase process of generating a shared secret key.



**Figure 100—The process of generating a shared secret key in three phases**

**15.1.1 Randomness sharing**

Since channel response is a location specific feature, a secret key can be extracted from the wireless channel that two PDs share. Figure YY illustrates the process of sharing common randomness from the wireless channel.

a) If PD1 and PD2 are engaged in packet exchange with each other, PD1 and PD2 can obtain a common random sequence by quantizing the channel responses estimated from the RTS and CTS packets. The RTS and CTS packets shall be exchanged within the channel coherent time. Any messages involving RTS/CTS handshaking, including peering messages, or data packets, can be used to obtain the common randomness.

b) Let be the frequency domain channel response corresponding to *N* non-null subcarriers estimated by PD1, then PD1 can obtain a random sequence of *N* bits by BPSK quantization of the channel response. Similarly, PD2 can obtain an *N* bit random sequence from the estimated frequency domain channel response .

c) PD1 and PD2 repeat this process until they obtain a common random sequence long enough to generate a 128 bit secret key. The required length of the random sequence *Nseq*, depends on the channel condition such as SNR, selectivity of the channel, etc. The mechanism of determining *Nseq* from channel condition is an implementation matter and is not specified in the standard.

d) After obtaining an *Nseq* bit common random sequence, PD1 and PD2 start post processing to extract a share secret key from the random sequence.



**Figure 101—Randomness sharing from RTS/CTS packets**

If PD1 and PD2 want to commence a secure communication, but the number of bits already obtained is not long enough to extract a 128 bit secret key, PD1 and PD2 exchange probe request and probe response messages for the purpose of obtaining additional common random bits, as illustrated in Figure 102.



**Figure 102—Randomness sharing from Probe Request/Response messages**

**15.1.2 Information reconciliation**

There may be a discrepancy between the random sequence obtained by PD1 and the random sequence obtained by PD2 during the randomness sharing step described in 15.1.1. The discrepancy can be removed by performing error correction code based information reconciliation.

1. Determine field size such that , where *Nseq* is the number of bits obtained in the randomness sharing step in 15.1.1.
2. Estimate the discrepancy between the random sequence obtained by PD1 and PD2 based on the channel condition. The estimation method is an implementation matter and is not specified in the standard.
3. Given and , calculate the necessary number of parity bits .
4. If , concatenate the *Nseq* bit random sequence and bit zero-padding sequence. If , repeat from b) with *m* increased by 1.
5. Encode the extended message with ~~systematic~~primitive narrow sense BCH (), where

, , and .

1. PD1 sends parity part of the codeword to PD2.
2. If the number of discrepancy is smaller than the error correction capability, i.e. , the discrepancy in the sequence can be corrected, and PD1 and PD2 will share an identical sequence . If the parity cannot correct the discrepancy, PD2 shall [the behavior is TBD].



**Figure 103—The size of the codeword for information reconciliation**

* + 1. **Privacy amplification**

The parity part of the codeword transmitted during the information reconciliation step can be overheard by an eavesdropper. The disclosed information is removed by privacy amplification. Since the number of disclosed bits during public discussion is and the parity check matrix of BCH code is of full rank (i.e., ), there will be an equivocation of bits in the bit codeword, as illustrated in Figure 104. Thus, the equivocation is reduced from bits to bits, which implies bits of information is leaked to an eavesdropper.



**Figure 104—The size of the codeword for information reconciliation**

Additional leakage of information can occur due to non-zero correlation between the wiretap channel and the channel between PD1 and PD2. Let be the correlation coefficient between random variables and , then the mutual information between and is given by

,

by data processing inequality, where represents an arbitrary post processing function. For example, if the correlation coefficient between the wiretap channel and the channel between PD1 and PD2 is , then up to bits of information can be leaked due the channel correlation, where . The calculation of is an implementation matter and is not specified in the standard, but the implementation shall assume the channel correlation is greater than or equal to 30 percent.

As a result, after information reconciliation, the total number bits of leaked information is given by

.

The leaked information can be removed by using a universal hash function

,

where is a security parameter. Note that, the remaining mutual information about the secret key after privacy amplification is less than bits. As illustrated in Figure 105, there is negligible remaining mutual information about the secret key when . Any implementation of the standard shall use greater than or equal to 10.



**Figure 105—Security parameter *Nsec* vs. remaining mutual information after privacy amplification**

The privacy amplification procedure is as follows:

1. PD1 transmits a randomly generated bit sequence to PD2.
2. PD1 and PD2 generate Toeplitz matrix T and eliminate the disclosed information by calculating . PD1 and PD2 have exactly the same bit secret key .

Let

be the bit sequence transmitted by PD1, then the Toeplitz matrix T is built as illustrated in Figure XX, where , .

Figure XX—Toeplitz matrix T for privacy amplification.

In wireless channel, channel observations between two adjacent subcarriers are correlated, and the correlation needs to be eliminated by applying entropy coding. Huffman coding with an alphabet size of 1 byte without reordering with a dictionary generated by empirical distribution of quantized bits shall be used. After correlation elimination, a key of length is obtained, where is compression efficiency. The first 128 bits of shall be used as a secret key.

***End of the proposed text.***