[0008] In LTE, the architecture of the UTRAN 100 will be changed. The RNC 130 no longer exists. An evolved Node-B (eNode-B) will assume medium access control (MAC) and some radio resource control (RRC) functionalities. Original RLC sub-layer and the data security, (or ciphering), entity in the RNC 130 will have to be re-located in LTE to maintain the necessary data encryption and data ARQ functionalities. Given this new LTE network architecture, the issue is where the outer ARQ entities and the data security entities shall be located and how the two formerly co-located entities cooperate to work in the LTE system.

[0009] Figure 2 shows a proposed LTE network 200 with respect to outer ARQ entities. The LTE network 200 includes a UE 210, an eNode-B 220 and an access gateway (aGW) 230. In the proposed LTE network 200, outer ARQ entities 212 and 222 are located in the UE 210 and the eNode-B 220, respectively. Placing the outer ARQ entity 222 in the eNode-B 220 would be optimal with respect to retransmission delay, retransmission PDU size, simple protocol complexity, low buffering requirements and possible hybrid ARQ (H-ARQ) and outer ARQ interaction. However, this approach does not have a user data security process in mind.

[0010] It would be optimal to place user data security entities in the UE 210 and the aGW 230, which is a network anchor node, for the following reasons. First, the security parameters of the UE 210 (or user), (such as UE security credentials, encryption key sets, or the like), may be kept in a safer place, (i.e., aGW 230), where the interaction of UE authentication with a home subscriber server (HSS) is administered. Second, user data may be protected all the way from the aGW 230 to the UE 210 without requiring an additional scheme to achieve at least the same level of security as in the conventional UTRAN 100. Third, eNode-B physical protection may be simplified, thus increasing the total system security protection and the system cost effectiveness, and simplifying the eNodeB functionality. ~~Forth~~, inter-Node-B handover and inter-aGW handover would be easier from less security context transfer, (between eNode-Bs if the data security entity is located on an eNode-

than a PDU size, multiple ciphered data blocks may be concatenated in one PDU.

[0034] Figure 6 shows an exemplary concatenated PDU 600 in accordance with the present invention. An optional ARQ SN 602 may be assigned by the outer ARQ entity 318, 322 to the PDU 600. The concatenated PDU 600 is generated from multiple ciphered data blocks and includes multiple segment headers 604a-604n. Each segment header 604a-604n indicates the ending position of the corresponding ciphered data blocks 608a-608n in the PDU 600. A different generic SN is used for each of the data blocks and the generic SNs 606a-606n are included in the PDU 600. The extension field 603 after the ARQ SN 602 indicates whether a segment header 604a is following or not. If the concatenation always supports in-sequence concatenated SDUs, the generic sequence number may only be included in the first concatenated SDU.

[0035] Alternatively, the outer ARQ entity 318, 322 may generate one PDU from one ciphered data block, (i.e., one-to-one mapping). When the ciphered data block size is close or same ~~to~~ the PDU size, the outer ARQ entity 318, 322 may generate one PDU from one data block. The one-to-one mapping may occur by coincidence or by configuration. If the one-to-one mapping is configured, the generic SN used by the cipher entity 316, 336 may include an ARQ SN, (either in higher or lower order bit positions in the generic SN). In this case, the generic SN is called a common SN. The common SN is a sequence number as a generic SN, but it embeds an ARQ SN. The common SN may be used when by configuration one data block is carried by one PDU. Since the ARQ SN is embedded in the common SN, the outer ARQ entity does not need to allocate another ARQ SN and a processing overhead is reduced.

[0036] Figure 7 shows an exemplary PDU 700 generated by one-to-one mapping. The PDU 700 includes a common SN 702 which embeds an ARQ SN 701. The segment header 704 indicates the last position of the ciphered data block 706. The data block size may be fixed, (by configuration), or may be flexible. The segment header 704 may be omitted if padding is zero or the size

[0089] 46. The method as in any of the embodiments 34-45, further comprising the step of re-segmenting one of the ciphered data block and the PDUs to multiple sub-level PDUs in smaller size than previous PDUs when transmission of at least one previous PDU fails.

[0090] 47. The method of embodiment 46, comprising the step of incrementing a segmentation version identifier.

[0091] 48. The method as in any of the embodiments 46-47, comprising the step of retransmitting the multiple PDUs.

[0092] 49. The method as in any of the embodiments 46-48, further comprising the step of terminating an H-ARQ process for the previous PDUs.

[0093] 50. The method as in any of the embodiments 46-49, further comprising the step of retransmitting a failed PDU when transmission of at least one previous PDU fails.

[0094] 51. The method as in any of the embodiments 34-45, wherein multiple ciphered data blocks are concatenated to one PDU.

[0095] 52. The method of embodiment 51, wherein a segment header is attached to the PDU.

[0096] 53. The method of embodiment 52, wherein a segment header is attached only if the PDU includes padding.

[0097] 54. The method as in any of the embodiments 51-53, further comprising the step of separating the PDU ~~to~~ multiple sub-PDUs when transmission of the PDU fails, each sub-PDU including at least one data block.

[0098] 55. The method of embodiment 54 further comprising the step of retransmitting the sub-PDUs.

[0099] 56. The method as in any of the embodiments 34-45, wherein one PDU is generated from one ciphered data block.

[00100] 57. The method of embodiment 56, wherein an ARQ SN is embedded in the generic SN.

[00101] 58. The method as in any of the embodiments 56-57, wherein a size of the data block is fixed.

[00127] 84. The WTRU as in any of the embodiments 81-83, wherein the outer ARQ entity is configured to separate the PDU ~~to~~ multiple sub-PDUs when transmission of the PDU fails, each sub-PDU including at least one ciphered uplink data block.

[00128] 85. The WTRU as in any of the embodiments 66-72, wherein the outer ARQ entity generates one PDU from one ciphered uplink data block.

[00129] 86. The WTRU of embodiment 85, wherein an ARQ SN is embedded in the generic SN.

[00130] 87. The WTRU as in any of the embodiments 85-86, wherein a size of the uplink data block is fixed.

[00131] 88. The WTRU as in any of the embodiments 85-86, wherein a size of the uplink data block is flexible.

[00132] 89. The WTRU as in any of the embodiments 85-88, wherein the outer ARQ entity is configured to segment one of the ciphered uplink data block and the PDU to multiple sub-level PDUs when transmission of the PDU fails, and increase a segmentation version identifier.

[00133] 90. The WTRU as in any of the embodiments 85-89, wherein an H-ARQ process for the previous PDU is terminated.

[00134] 91. A PDU for implementing data security and ARQ.

[00135] 92. The PDU of embodiment 91 comprising an ARQ SN for identifying a segment of a data block.

[00136] 93. The PDU of embodiment 92 comprising an extension for indicating whether a segment header is following.

[00137] 94. The PDU as in any of the embodiments 92-93, a generic SN used for ciphering the data block.

[00138] 95. The PDU of embodiment 94, comprising at least one data portion ciphered by using the generic SN.

[00139] 96. The PDU as in any of the embodiments 92-95, further comprising at least one segment header, the segment header including a length indicator for indicating an end point of corresponding data portion and

[0008] In LTE, the architecture of the UTRAN 100 will be changed. The RNC 130 no longer exists. An evolved Node-B (eNode-B) will assume medium access control (MAC) and some radio resource control (RRC) functionalities. Original RLC sub-layer and the data security, (or ciphering), entity in the RNC 130 will have to be re-located in LTE to maintain the necessary data encryption and data ARQ functionalities. Given this new LTE network architecture, the issue is where the outer ARQ entities and the data security entities shall be located and how the two formerly co-located entities cooperate to work in the LTE system.

[0009] Figure 2 shows a proposed LTE network 200 with respect to outer ARQ entities. The LTE network 200 includes a UE 210, an eNode-B 220 and an access gateway (aGW) 230. In the proposed LTE network 200, outer ARQ entities 212 and 222 are located in the UE 210 and the eNode-B 220, respectively. Placing the outer ARQ entity 222 in the eNode-B 220 would be optimal with respect to retransmission delay, retransmission PDU size, simple protocol complexity, low buffering requirements and possible hybrid ARQ (H-ARQ) and outer ARQ interaction. However, this approach does not have a user data security process in mind.

[0010] It would be optimal to place user data security entities in the UE 210 and the aGW 230, which is a network anchor node, for the following reasons. First, the security parameters of the UE 210 (or user), (such as UE security credentials, encryption key sets, or the like), may be kept in a safer place, (i.e., aGW 230), where the interaction of UE authentication with a home subscriber server (HSS) is administered. Second, user data may be protected all the way from the aGW 230 to the UE 210 without requiring an additional scheme to achieve at least the same level of security as in the conventional UTRAN 100. Third, eNode-B physical protection may be simplified, thus increasing the total system security protection and the system cost effectiveness, and simplifying the eNodeB functionality. Fourth, inter-Node-B handover and inter-aGW handover would be easier from less security context transfer, (between eNode-Bs if the data security entity is located on an eNode-

than a PDU size, multiple ciphered data blocks may be concatenated in one PDU.

[0034] Figure 6 shows an exemplary concatenated PDU 600 in accordance with the present invention. An optional ARQ SN 602 may be assigned by the outer ARQ entity 318, 322 to the PDU 600. The concatenated PDU 600 is generated from multiple ciphered data blocks and includes multiple segment headers 604a-604n. Each segment header 604a-604n indicates the ending position of the corresponding ciphered data blocks 608a-608n in the PDU 600. A different generic SN is used for each of the data blocks and the generic SNs 606a-606n are included in the PDU 600. The extension field 603 after the ARQ SN 602 indicates whether a segment header 604a is following or not. If the concatenation always supports in-sequence concatenated SDUs, the generic sequence number may only be included in the first concatenated SDU.

[0035] Alternatively, the outer ARQ entity 318, 322 may generate one PDU from one ciphered data block, (i.e., one-to-one mapping). When the ciphered data block size is close or same as the PDU size, the outer ARQ entity 318, 322 may generate one PDU from one data block. The one-to-one mapping may occur by coincidence or by configuration. If the one-to-one mapping is configured, the generic SN used by the cipher entity 316, 336 may include an ARQ SN, (either in higher or lower order bit positions in the generic SN). In this case, the generic SN is called a common SN. The common SN is a sequence number as a generic SN, but it embeds an ARQ SN. The common SN may be used when by configuration one data block is carried by one PDU. Since the ARQ SN is embedded in the common SN, the outer ARQ entity does not need to allocate another ARQ SN and a processing overhead is reduced.

[0036] Figure 7 shows an exemplary PDU 700 generated by one-to-one mapping. The PDU 700 includes a common SN 702 which embeds an ARQ SN 701. The segment header 704 indicates the last position of the ciphered data block 706. The data block size may be fixed, (by configuration), or may be flexible. The segment header 704 may be omitted if padding is zero or the size

[0089] 46. The method as in any of the embodiments 34-45, further comprising the step of re-segmenting one of the ciphered data block and the PDUs to multiple sub-level PDUs in smaller size than previous PDUs when transmission of at least one previous PDU fails.

[0090] 47. The method of embodiment 46, comprising the step of incrementing a segmentation version identifier.

[0091] 48. The method as in any of the embodiments 46-47, comprising the step of retransmitting the multiple PDUs.

[0092] 49. The method as in any of the embodiments 46-48, further comprising the step of terminating an H-ARQ process for the previous PDUs.

[0093] 50. The method as in any of the embodiments 46-49, further comprising the step of retransmitting a failed PDU when transmission of at least one previous PDU fails.

[0094] 51. The method as in any of the embodiments 34-45, wherein multiple ciphered data blocks are concatenated to one PDU.

[0095] 52. The method of embodiment 51, wherein a segment header is attached to the PDU.

[0096] 53. The method of embodiment 52, wherein a segment header is attached only if the PDU includes padding.

[0097] 54. The method as in any of the embodiments 51-53, further comprising the step of separating the PDU into multiple sub-PDUs when transmission of the PDU fails, each sub-PDU including at least one data block.

[0098] 55. The method of embodiment 54 further comprising the step of retransmitting the sub-PDUs.

[0099] 56. The method as in any of the embodiments 34-45, wherein one PDU is generated from one ciphered data block.

[00100] 57. The method of embodiment 56, wherein an ARQ SN is embedded in the generic SN.

[00101] 58. The method as in any of the embodiments 56-57, wherein a size of the data block is fixed.

[00127] 84. The WTRU as in any of the embodiments 81-83, wherein the outer ARQ entity is configured to separate the PDU into multiple sub-PDUs when transmission of the PDU fails, each sub-PDU including at least one ciphered uplink data block.

[00128] 85. The WTRU as in any of the embodiments 66-72, wherein the outer ARQ entity generates one PDU from one ciphered uplink data block.

[00129] 86. The WTRU of embodiment 85, wherein an ARQ SN is embedded in the generic SN.

[00130] 87. The WTRU as in any of the embodiments 85-86, wherein a size of the uplink data block is fixed.

[00131] 88. The WTRU as in any of the embodiments 85-86, wherein a size of the uplink data block is flexible.

[00132] 89. The WTRU as in any of the embodiments 85-88, wherein the outer ARQ entity is configured to segment one of the ciphered uplink data block and the PDU to multiple sub-level PDUs when transmission of the PDU fails, and increase a segmentation version identifier.

[00133] 90. The WTRU as in any of the embodiments 85-89, wherein an H-ARQ process for the previous PDU is terminated.

[00134] 91. A PDU for implementing data security and ARQ.

[00135] 92. The PDU of embodiment 91 comprising an ARQ SN for identifying a segment of a data block.

[00136] 93. The PDU of embodiment 92 comprising an extension for indicating whether a segment header is following.

[00137] 94. The PDU as in any of the embodiments 92-93, a generic SN used for ciphering the data block.

[00138] 95. The PDU of embodiment 94, comprising at least one data portion ciphered by using the generic SN.

[00139] 96. The PDU as in any of the embodiments 92-95, further comprising at least one segment header, the segment header including a length indicator for indicating an end point of corresponding data portion and