(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
02 November 2017 (02.11.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/189590 A1

(51) International Patent Classification:
H04L 9/08 (2006.01) *H04L 9/32* (2006.01)

(21) International Application Number:
PCT/US2017/029436

(22) International Filing Date:
25 April 2017 (25.04.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/329,370 29 April 2016 (29.04.2016) US

(71) Applicant: **PCMS HOLDINGS, INC.** [US/US]; 200 Bellevue Parkway, Suite 300, Wilmington, Delaware 19809 (US).

(72) Inventor: **SCHMIDT, Andreas**; Dillenburger Strasse 13, 60439 Frankfurt am Main (DE).

(74) Agent: **STECK, Jeffrey Alan**; Invention Mine LLC, 216 South Jefferson Street, Suite 102, Chicago, Illinois 60661 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) **Title:** SYSTEM AND METHOD FOR PHYSICAL LAYER AUTHENTICATION AND KEY AGREEMENT

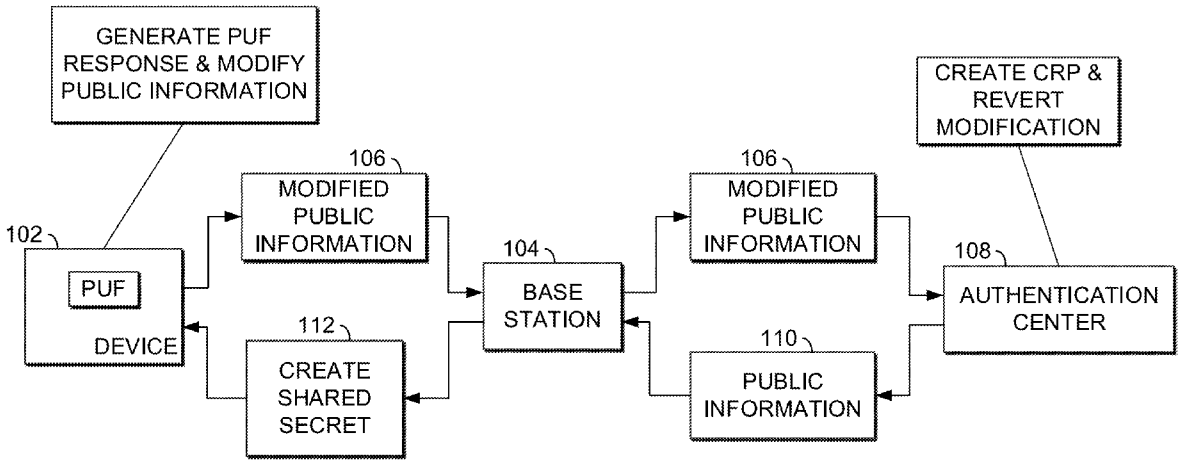

FIG. 1

(57) **Abstract:** Systems and methods are provided for authentication and key agreement between wireless communications devices. The devices measure dynamic channel conditions between them to generate random (but correlated) unreconciled channel data sets. In exemplary embodiments, challenge-response authentication is performed between the devices, where the challenge value is determined from least one of the channel data sets. Information reconciliation is performed between the two channel data sets to identify a shared secret for key generation. Communications used for information reconciliation are obfuscated at least in part by the response value in the challenge-response authentication.

WO 2017/189590 A1

[Continued on next page]

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

SYSTEM AND METHOD FOR PHYSICAL LAYER AUTHENTICATION AND KEY AGREEMENT

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present application is a non-provisional filing of, and claims benefit under 35 U.S.C. §119(c) from, U.S. Provisional Patent Application Serial No. 62/329,370, filed April 29, 2016, entitled "System and Method for Physical Layer Authentication and Key Agreement," which is incorporated herein by reference in its entirety.

BACKGROUND

[0002] Physical ("PHY") layer security methods have received attention in applied security research because they can provide methods for secure communication without necessitating conventional cryptographic algorithms and corresponding computational capabilities of devices. Two methods used in the PHY layer security field are physically unclonable functions ("PUF") and shared key generation from wireless PHY channel properties. PUFs have been shown to be practically implementable on devices as small as RFID tags. Stable outputs from PUFs may be obtained using techniques derived from error correction. The implementation of practical PUFs is described in, for example, the following disclosures: Soybali et al., "Implementation of a PUF Circuit on a FPGA," 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2011; Devadas et al., "Design and Implementation of PUF-Based 'Unclonable' RFID ICs for Anti-Counterfeiting and Security Applications," 2008 IEEE International Conference on RFID, Las Vegas, Nevada; NXP, "PUF – Physical Unclonable Functions: Protecting next-generation Smart Card ICs with SRAM-based PUFs".

SUMMARY

[0003] Systems and methods are presented to achieve joint entity authentication and key agreement ("AKA") between an authenticating device and a receiving device (base station) using principles and methods of the physical layer. Exemplary systems and methods can additionally use PUF on part of the authenticating device and properties of the fading wireless channel between authenticating device and base station. Exemplary systems and methods provide AKA and ultimately achieve a secret key shared privately between device and base station, without requiring specialized cryptographic methods at both entities.

[0004] Exemplary systems and methods may further be implemented using cryptographic (shared secret key) method for authentication and key agreement (wherein the AKA protocol is

still based entirely on the PHY layer). In this case, the secret keys may be stored in a protected domain of the device, such as an (embedded) SIM card. This may be less desirable for very small devices deployed in very large numbers. Therefore, using PUFs for authentication in this context may be advantageous.

[0005] Systems and methods are provided for authentication and key agreement between wireless communications devices. The devices both perform measurements of dynamic channel conditions between the devices to generate effectively random unreconciled channel data sets that are expected to be highly correlated with each other due to Lorentz reciprocity but to be effectively unknowable to an eavesdropper. Challenge-response authentication is performed between the devices, where the challenge value is determined based on at least one the channel data sets rather than being, e.g. randomly selected. In some embodiments, the response is generated using a physically unclonable function and is confirmed using an emulation of that function. Information reconciliation is performed between the two channel data sets to identify a shared secret for key generation. Communications used for information reconciliation are obfuscated in part by the response value in the challenge-response authentication. In this way, failure by an eavesdropper to obtain the correct channel measurements will defeat not only the generation of a shared key, but also the success of challenge-response authentication. Similarly, failure to emulate the correct challenge-response function will defeat not only the challenge-response authentication, but also the generation of a shared key.

[0006] In an exemplary method, a receiving device such as a base station operates to measure dynamic channel conditions between the receiving device and an authenticating device. The receiving device generates a first unreconciled data set, for example a bit sequence $(Y_i), i = 1, \ldots, N$, from the measurements. The receiving device receives a challenge value c and an obfuscated second unreconciled data set $\hat{L}$ from the authenticating device. The receiving device determines a valid challenge response $r=F^*(c)$ to the challenge value. The function F^* may be, for example, a function that emulates a physically unclonable function in the authenticating device. Using the valid challenge response r, the receiving device removes the obfuscation from the second unreconciled data set, e.g. by XORing the two together to recover an un-obfuscated second unreconciled data set $L = \hat{L} \oplus r$. The received challenge value c may then be validated by determining whether $c = \sigma(L)$ for a predetermined function σ. The first and second data sets may be reconciled to generate a reconciled data set $\tilde{L}$, and the reconciled data set may be used by both the receiving device and the authenticating device in the generation of a secret key for

communication between the devices. For example, the reconciled data set $\tilde{L}$ may be used to select bits from the bit sequence (Y_i) to use as a secret key.

[0007] In some exemplary methods, a receiving device operates to measure dynamic channel conditions between the receiving device and an authenticating device. The receiving device receives a challenge value and an obfuscated key derivation seed from the authenticating device and determines a valid challenge response to the challenge value. The valid challenge response may be determined based on a physically unclonable function of the authenticating device, among other techniques. Using the valid challenge response, the receiving device removes the obfuscation from the key derivation seed. Removing the obfuscation may include performing a bitwise XOR operation on the valid challenge response and the key derivation seed. The receiving device then generates a key for communication between the receiving device and the authenticating device based at least in part on (i) the measured channel conditions and (ii) at least a selected portion of the key derivation seed. The selected portion of the key derivation seed may be selected by a method comprising reconciling the received key derivation seed with a locally-generated key derivation seed.

[0008] In one such method, the receiving device generates a bit sequence from the dynamic channel measurements, and the key is generated based on values at bit positions in the bit sequence, the bit positions being identified by the selected portion of the key derivation seed.

[0009] In some embodiments, the receiving device validates the received challenge value based on the key derivation seed. The validation of the received challenge value may include applying a predetermined function to the key derivation seed and comparing the result to the received challenge value.

[0010] In some embodiments, the key is generated by selecting a portion of the channel condition measurements using the selected portion of the key derivation seed and generating a keyed hash message authentication code (HMAC) to a pairwise transient key (PTK) using the selected portion of the channel condition measurements as a hash key.

[0011] Some embodiments include method steps performed by the authenticating device. In some such embodiments, the authenticating device operates to measure the dynamic channel conditions between the receiving device and the authenticating device. The authenticating device generates a bit sequence from the dynamic channel measurements. The authenticating device generates the key derivation seed, wherein the key derivation seed identifies a plurality of bit positions in the bit sequence. The authenticating device generates the challenge value and determines the valid challenge response based on the challenge value. The authenticating device

obfuscates the key derivation seed using the valid challenge response and sends to the receiving device the obfuscated key derivation seed and the challenge value.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] FIG. 1 is a functional block diagram illustrating a high-level overview of an exemplary system employing a PHY AKA protocol, in accordance with some embodiments.

[0013] FIG. 2 is a message flow diagram illustrating a compact PHY AKA protocol, in accordance with some embodiments.

[0014] FIG. 3 is a message flow diagram illustrating another compact PHY AKA protocol in accordance with some embodiments.

[0015] FIG. 4 is a message flow diagram illustrating a compact PHY AKA protocol integrated into an 802.1x authentication procedure using an Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA), in accordance with some embodiments.

[0016] FIG. 5 illustrates an exemplary wireless transmit/receive unit (WTRU) that may be employed as an exemplary communications device (e.g. the device in FIGs. 1-3 or the supplicant device in FIG. 4) in some embodiments.

[0017] FIG. 6 illustrates an exemplary network entity that may be employed as an exemplary communications device (e.g. base station or access point) and/or as an authentication center or AAA server.

DETAILED DESCRIPTION

[0018] PUFs are functions which generate some secret key from the immutable properties of the underlying hardware. They come in two flavors, characterized by the number of challenge-response pairs ("CRP"), which they are able to sustain. Weak PUFs offer only a small number of responses to corresponding challenges, often only a single response. A typical example for a weak PUF is the power-up state of an SRAM. Since the output of a weak PUF is essentially a key which should be kept secret to be of sustained utility, weak PUFs are suitable for purposes of secret key derivation, but, in particular, cannot be directly used for entity authentication using CRPs. Strong PUFs, in contrast may support a large number, maybe even arbitrarily many, CRPs, and thus can be used for entity authentication directly. Examples of strong PUFs that can be used in exemplary embodiments include PUFs based on ring oscillators and arbiter circuits,

such as those described, for example, in C. Herder et al., "Physical Unclonable Functions and Applications: A Tutorial," Proc. IEEE 102(8), 1126-1140, 2014.

[0019] Shared key generation from PHY channel properties relies on channel characteristics as a source of randomness, using for example techniques described in C. Ye et al., "Information-Theoretically Secret Key Generation for Fading Wireless Channels," IEEE Trans. Inform. Forensics and Security 5(2), 240-254, 2010. One component in this type of key generation is exchange of public information between the communication partners, called public discussion. Since the channel characteristics are unique to a pair of communication partners, a key resulting from such a procedure is shared between and is private to the pair of partners. Although PHY layer secret key generation may be extended to provide channel authentication, it cannot directly be used to authenticate either of the partners as unique entities.

[0020] Systems and methods are presented herein to achieve joint entity AKA between an authenticating device and a receiving device (e.g. a base station or other access points) using properties of the PHY layer. An exemplary PHY AKA procedure includes an authenticating device, a base station ("BS"), and an authentication center ("AC"). The base station is a receiving device with which the authenticating device is attempting to establish a secure communication channel and from which the requesting device is seeking authentication. The authentication center supports the BS in authentication and key establishment.

[0021] The device and BS exchange public information to establish a secret from wireless channel characteristics. In an embodiment of a PHY AKA procedure, a challenge value is generated and provided to a PUF to generate a response. The response in turn is used to obfuscate (e.g. to encrypt) the data from the public exchange in such a way that the data is useless to an eavesdropper for generating a shared secret key between the device and the BS. Furthermore, the modification of the public information depends, in a preferred embodiment, on the properties of the PHY channel, by selecting the challenge for the PUF depending on said properties. It should be noted at this point that, instead of using a PUF, any challenge-response device can be used, in particular traditional cryptographic methods based on a secret key shared between the device (such as a secret key stored on a smart card and stored by an AC).

[0022] The PUF response is applied to the modified public information to revert the modified public information. Only an entity which is able to produce said response output, and concurrently experiences the same PHY channel properties as the device and the BS, will be able to eventually generate the shared secret key.

[0023] FIG. 1 depicts a high-level overview of the PHY AKA concept, in accordance with some embodiments. As depicted in FIG. 1, the authenticating device first obtains some information from the PHY channel between the device 102 and the BS 104 and prepares public information which would normally be sent to a BS to establish a shared secret. However, the device first generates a response from its PUF (or other secret function, such as an encryption function) and uses the response to modify the public information in a way that makes it unusable for the secret generation procedure directly (e.g. by using the latter as a one-time pad for encryption). However, the modification is reversible. The modified information 106 is sent to the BS 104 and handed further from the BS 104 to the AC 108. The AC 108 is able to re-create the PUF response and use it to revert the modification of the public information. The recovered public information 110 is sent back to the BS 104, which uses it to proceed with the shared secret generation procedure 112, interacting with the device. The key generation will only succeed with both parties sharing the same key if the modification of public information was reverted correctly by the AC, which is only possible if the PUF response of the AC is the same as that of the device. In this way, the device is authenticated as a unique entity known to the AC, as proven by the possession of the correct shared key by the device.

[0024] Exemplary embodiments include all features of an AKA protocol. For example, exemplary embodiments provide strong access control such that an encrypted channel can only be established between the BS and the device if the BS successfully authenticated the device. Additionally, the scheme can provide more privacy since the key generation process is completely private to the device and the BS, which is different from other mobile network AKA procedures. The binding of the authentication to the PHY key agreement increases security by making the process unique and difficult to subvert for an attacker. In particular, the binding entails the "strong access control" property that no secret communication channel can be established without successful entity authentication. The authentication, in turn, is bound to the properties of the physical channel which entails the mentioned protection against, for instance, replay attacks by an eavesdropper.

[0025] Exemplary embodiments provide a compact protocol which in some embodiments relies on physical channel properties and properties of the involved devices only. Use of specialized cryptographic methods is avoided. In at least one embodiment, the method has no additional communication steps beyond a PHY key agreement procedure that uses a single forth-back exchange of messages, such as the procedure described in Ye et al., *supra*.

[0026] Exemplary embodiments may be advantageous for devices that are deployed for a very long time in the field, e.g., they do not use cryptographic algorithms which could become "weak" by attackers' increasing capabilities. Exemplary embodiments may also be particularly advantageous for devices that operate on very low power, or even without power, e.g., using energy harvesting. Embodiments are also useful in devices that communicate highly sensitive data.

[0027] Exemplary embodiments augment the opportunity to use low-power, low-capability devices for communication of sensitive data in large-scale deployments.

[0028] Exemplary embodiments integrate authentication directly with the PHY layer key agreement procedure which may leads to more compact PHY AKA protocols. Exemplary embodiments are more suitable to very small footprint (non-cellular) devices.

[0029] Systems and methods of PHY layer security presented herein may employ strong binding of PHY layer key agreement and a strong PUF. The strong binding of the PHY layer key agreement procedures to an entity authentication operate such that, as a net effect, a secret shared key between communication partners is achieved if the device requesting the setup of the secure communications channel is authenticated as a unique entity. The entity authentication is based on a strong PUF, which avoids using a cryptographic infrastructure and secure elements for authentication and thus makes it possible to deploy embodiments in very small footprint devices deployed at a large scale. It should be noted that the concepts described above are modular in the sense that the second can be added to the first one as an additional variant.

[0030] In at least one embodiment, a classical challenge-response scheme is used which can be realized by secret shared keys at the device and the AC and cryptographic procedures such as in 3G AKA or using keyed-hash message authentication code (HMAC) function. Thus, in an exemplary embodiment, the device possesses logic for performing a function F, which yields a deterministic response r to any challenge c, $r = F(c)$, and which is hard to forge. Further, the AC possesses an equivalent function for verification $r = F^*(c)$, for each device. In the case a physical challenge-response scheme is used, F may be a strong PUF and F^* may be a PUF emulation model, or a database of known challenge-response pairs known only to the AC.

[0031] The device has a unique identifier known to the AC. The AC is configured to create valid responses to challenges of the PUF of the device. FIG. 2 depicts a compact PHY AKA protocol, in accordance with some embodiments. In particular, FIG. 2 gives an overview over the steps described below.

[0032] In step 201, the device 222 and base station 224 exchange probe data over the PHY channel and obtain respective bit sequences representing measurements of the PHY channel. For example, the device 222 may obtain the bit sequence $(X_i), i = 1, \ldots, N$, and the base station 224 may obtain the bit sequence $(Y_i), i = 1, \ldots, N$. In some embodiments, these bit sequences are generated as follows. A series of measurements of the channel between device D and base station BS is taken by those devices. The measurements of the channel are low-pass filtered such that the mean of the channel measurements is approximately zero. A thresholding function is applied to each of the measurements such that a measurement is assigned a bit value of "1" if it is above a threshold, a bit value of "0" if it is below a negative threshold, and is otherwise undefined. One exemplary technique that can be used to generate a bit sequence from channel measurements is described in Ye et al., *supra*. Other techniques of generating a bit sequence from channel measurements may alternatively be used. Due to Lorentz reciprocity, the bit sequence (X_i) generated by the device D and the sequence (Y_i) generated by the base station BS can be expected to demonstrate a high level of correlation, although the correlation need not be perfect. Preferably, for each value of i, a channel measurement by device D that is used to generate value X_i is taken within a short time interval of the measurement by base station BS that is used to generate value Y_i. The time interval is preferably less than the coherence time of the channel.

[0033] In step 202, the device 222 selects a sequence L of numbers that are indices in the range $1, \ldots, N$. Each number in the sequence L identifies a position in the sequence (X_i). The sequence L operates as a key derivation seed in that both the device 222 and the base station 224 use at least a portion of L, together with the respective channel measurements they have obtained, to generate a key. Different techniques may be used to select the sequence L. One such technique operates as follows. A parameter m is selected (m may be a predetermined value). A plurality of excursions are identified in the sequence (X_i), where an excursion refers here to a series of at least m consecutive identical values of X_i (e.g. at least m consecutive zeros or m consecutive ones). From among the excursions, a subset of excursions is selected (e.g. by random selection from among all the identified excursions), and each number in the sequence L represents one of the excursions. For example, each number in the sequence L may represent the index j of the value X_j that lies at the center of the excursion. An exemplary technique that may be used for generating a sequence L of numbers from channel measurements is described in Ye et al., *supra*. Other techniques may alternatively be used.

[0034] The numerical sequence L is a random sequence of numbers in $\{1, \ldots, N\}$ but the numbers in the sequence are not necessarily independent and identically distributed (i.i.d.). In an exemplary embodiment, in step 203a, a selection function σ is applied to the sequence L to obtain the challenge value $c = \sigma(L)$. The function σ may be public and may for instance include binary encoding of L, such as taking the bit parity of each of the numbers in the list. The function σ may include bit permutation. Thus, the function σ is selected so as to extract additional entropy from the sequence L and to approximate a desired distribution (such as i.i.d.). In some embodiments, the challenge value c is not derived from the numerical sequence L.

[0035] In step 203b, the device 222 applies a challenge-response function F to the challenge value c to create a response value r, where $r = F(c) = F(\sigma(L))$. In some embodiments, the challenge-response function F is a physically unclonable function.

[0036] In step 203c, the device 222 obfuscates the key derivation seed, sequence L in this embodiment, by applying the response value r as a one-time-pad (OTP) to L to generate obfuscated key derivation seed $\hat{L}$. In some embodiments, a bitwise XOR function is used by device 222 such that $\hat{L} = L \oplus r$. The length of the OTP may be adjusted to the length of L, for instance by repeating the sequence r and pruning.

[0037] In step 204, the device 222 sends the values $(id, c, \hat{L})$ to the base station 224, where id is an identifier of device 222. In step 205, the base station 224, in turn, sends the values (id, c) to the authentication center 226. In step 206, the authentication center 226 uses the id value to select a verification function F^* associated with the device 222. Using the selected function F^*, the authentication center 226 recreates the response value r from (id, c) by calculating $r = F^*(c)$. In step 207, the authentication center 226 sends r to the base station.

[0038] In step 208a, the base station removes the OTP obfuscation on $\hat{L}$ by applying r. For example, the base station may apply a bitwise XOR operation to recover $L = \hat{L} \oplus r$. To verify that the device 222 correctly followed the procedure and derived c from the PHY channel-dependent L, the base station 224 may, in some implementations, additionally verify that $c = \sigma(L)$ in step 208b.

[0039] In step 209, the base station 224 uses the recovered list L and the sequence (Y_i) (generated from the base station's own channel measurements) to create a reconciled list $\tilde{L}$. In some embodiments, the reconciled list $\tilde{L}$ is a selected portion of the recovered list L. In some embodiments, the base station determines which of the indices in the list L correspond to values in the sequence (Y_i) that fall within excursion of length $(m - 1)$ or longer, and the base station

includes only those indices in the reconciled list $\tilde{L}$. The reconciled list $\tilde{L}$ thus represents a list indices that the device 222 and base station 224 both agree are within an excursion. (It should be noted that the reconciled list $\tilde{L}$ does not necessarily identify all such excursions on which device 222 and base station 224 agree.) One technique that may be used for the generation of a reconciled list $\tilde{L}$ is the method described in Ye et al., *supra*. In some embodiments, the base station determines a number (or proportion) of excursions identified in list L that also correspond to excursions in sequence (Y_i). If this number (or proportion) is too low, e.g. below a predetermined threshold, the base station 224 may determine that the list L was sent by an entity other than device 222.

[0040] In step 210, the base station sends the reconciled list $\tilde{L}$ to the device 222. At this point, the device 222 and base station 224 have exchanged information regarding the location of excursions that are found in both sequences (X_i) and (Y_i). However, the device 222 and base station 224 have not exchanged information indicating whether each of those excursions is an excursion of ones or an excursion of zeros. Nevertheless, due to Lorentz reciprocity, device 222 and base station 224 are expected to have the same information as to whether each excursion is an excursion of ones or of zeros. (The probability of agreement may be increased if desired by, for example, increasing the value of the parameter m.) In steps 211a and 211b, device 222 and base station 224 use this shared information as the basis for generating a shared key k.

[0041] In some embodiments, the device 222 may generate the shared key by concatenating the bit values of X_i for every value of i listed in the reconciled list $\tilde{L}$. Similarly, the base station 224 may generate the secret key by concatenating the bit values of Y_i for every value of i listed in the reconciled list $\tilde{L}$. One technique that may be used for the generation of a shared key k is the method described in Ye et al., *supra*.

[0042] In exemplary embodiments such as the method of FIG. 2, the AKA challenge is generated from the PHY channel properties themselves. This provides a strong binding of the two procedures and uniqueness, which helps to prevent of replay attacks for both authentication and key agreement. Therefore, an eavesdropper Eve listening to the communication between the device 222 and the BS 224 cannot perform the PHY channel key generation, since she first experiences different channel properties. Similarly, Eve cannot subvert the authentication procedure for instance by emulating the device 222 and sending a recorded challenge and her own list L_{Eve} to the base station. For this, note that the information sent from the device 222 to the base station 224 is secret, since it is obfuscated by an OTP operation, and thus Eve can also not use the correct L to authenticate.

[0043] In the above protocol as depicted in FIG. 2, device 222 is the party initiating the PHY key generation procedure by sending the list L, and concurrently transmits the self-generated challenge c to BS. An analogous protocol can be created as an essential variant of the above method, in which said roles of the device and BS are exchanged.

[0044] FIG. 3 depicts a compact PHY AKA protocol in which the roles of the device and base station are exchanged, in accordance with some embodiments. FIG. 3 shows this exchanged protocol using the same notation used with respect to FIG. 2. In the embodiment of FIG. 3, unlike in FIG. 2, the authenticator base station controls the creation of the challenge c and in consequence also binding of the challenge to the PHY channel properties.

[0045] In step 301, the device 222 and base station 224 probe the properties of the physical channel. Based on its measurements of the physical channel the base station 224 in step 302 creates list L (analogous to step 202). In step 303, a selection function σ is applied to the sequence L to obtain the challenge value $c = \sigma(L)$. In step 304, the base station 224 receives an identifier *id* from the device 222, and in step 305, the base station 224 sends the identifier along with challenge value c to the authentication center 226. In step 306, the authentication center 226 uses the id value to select a verification function F^* associated with the device 222. Using the selected function F^*, the authentication center 226 generates the response value r from (id, c) by calculating $r = F^*(c)$. In step 307, the authentication center 226 sends r to the base station.

[0046] In step 308, the base station applies OTP obfuscation on L by applying r. For example, the base station may apply a bitwise XOR operation to generate $\hat{L} = L \oplus r$. In step 309, the base station transmits the obfuscated list $\hat{L}$ and the challenge value c to the device 222. In step 310a, the device 222 applies a challenge-response function F to the challenge value c to create a response value r, where $r = F(c)$. In step 310b, the device 222 removes the OTP obfuscation on $\hat{L}$ by applying r. For example, the device may apply a bitwise XOR operation to recover $L = \hat{L} \oplus r$.

[0047] In step 311 (analogous to step 209), the device 222 uses the recovered list L and the sequence (X_i) (generated from the device's own channel measurements) to create a reconciled list $\tilde{L}$. The creation of a reconciled list is described in greater detail above. In step 312, the device 222 sends the reconciled list $\tilde{L}$ to the base station 224. In steps 313a and 313b, the device 222 and base station 224 create a key k as described above with respect to steps 211a, 211b.

[0048] In at least one embodiment, the methods presented herein are applicable when the strong PUF at the device is replaced with another suitable method such as the combination of

shared key and HMAC, wherein the shared key may be a secret shared key on secure hardware (e.g., a smart card) or a weak PUF.

[0049] It has been noted that strong PUFs may be vulnerable to modeling attacks if an attacker is able to collect enough CRPs, which will eventually enable the attacker to emulate the PUF. In at least one embodiment, only the base station is in possession of CRPs in the absence of eavesdropping. To avoid that a third party obtains CRPs, the channel between the base station BS and authentication center AC may be secured, in which case the security of the CRP becomes identical to that in common 2G/3G mobile network AKA. A variant of the method which restricts the possession of CRPs to the authentication center 226 may be used in which the base station 224 additionally sends $\hat{L}$ to the authentication center in step 205; authentication center 226 additionally removes the obfuscation in step 206, and the authentication center 226 sends L (and not r) back to the base station 224 in modified step 207. In such an embodiment, the execution of step 208a is essentially moved from the base station 224 to the authentication center 226.

[0050] In at least one embodiment, the authentication center 226 is configured to create a valid PUF response to every challenge. This may be done by providing the authentication center 226 with a complete computational model of the PUF. The latter class of PUFs is suspected to be slightly weaker regarding security, for instance they might enable an attacker to model the PUF from sufficiently many CRPs. On the other hand, a PUF which cannot be modeled has other, more practical drawbacks. In particular, the authentication center 226 may be equipped with a database of valid CRPs and may discard CRPs once used. When such a non-model PUF is used, the device 222 obtains a valid challenge from the authentication center 226, or the device confirms that its own challenge is valid for the authentication center 226. This can be achieved by sending id (or (id, c)) from the device to the authentication center 226 via the base station 224 before step 203a, upon which the authentication center 226 responds with a valid challenge c (or confirms that the device's c is valid).

[0051] It should also be noted that in at least one embodiment, the described methods are fully applicable to any key agreement procedure which relies on the exchange of public information for key reconciliation. This holds in particular for quantum cryptographic key agreement as in the BB84 and similar protocols.

[0052] Examples of the devices and deployments which can benefit from the presented method include ultra-low-power devices in M2M communication and the Internet of Things

(IoT). Mobile devices, such as cargo transponders and non-mobile devices such as thermostats, industrial sensors, etc., are equally fit targets for the technology.

[0053] The methods disclosed herein may be used in some embodiments to secure communication in mesh networks. For example, a PHY AKA procedure can be applied through multiple hops of the network, starting from the base station to the first hop, then from BS to second hop through the secure channel to the first hop, and so on.

[0054] In an exemplary embodiment, the above described method may be integrated in the standard protocol for 802.1X authentication, e.g. for the secure attachment of a wireless client at a base station. As a concrete example for an authentication method usable within the 802.1x framework, Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA) is chosen. For clarity of explanation, not all message parameters are illustrated in FIG. 4.

[0055] FIG. 4 depicts a compact PHY AKA protocol integrated in an 802.1x authentication procedure using EAP-AKA authentication method, in accordance with some embodiments. In particular, FIG. 4 depicts the following steps. In step 401 the device, called a supplicant in EAP terminology, discovers the Access Point AP, which sets its state to 'Unauthorized Client'. In step 402, the AP and supplicant probe the PHY channel to obtain (not yet reconciled) bit sequences J', and J, respectively. In step 403, the AP sends an EAP Request/Identity to the supplicant.

[0056] In step 404, the supplicant creates a value C=s(J) using a public function s (analogous to step 203a of FIG. 2). In step 405, the supplicant sends an EAP Response/Identity message. The format of the identity may be compliant with the Network Access Identifier (NAI) format specified in 3GPP TS 23.003. NAI contains either a pseudonym allocated to the supplicant in previous authentication or, in the case of first authentication, the IMSI. '0' (ASCII 0x30) will be pre-pending to the IMSI, resulting in 0IMSI@realm. The supplicant appends the value C to the identity.

[0057] In step 406, the AP sends EAP ID and the value C to the AAA server using a RADIUS Access Request. In step 407 The AAA server fetches an authentication vector containing values RAND*, XRES*, AUTN, for the supplicant with the aforementioned EAP ID. The normal values of the client challenge RAND and the comparison response XRES are herein modified to new values RAND* and XRES* using the value C in an appropriate way, for instance using an HMAC or another one-way function. In this, the normal comparison response XRES is first created using a function F* (analogous to step 206 of FIG. 2). The creation of XRES may for instance be performed in accordance with the AKA procedures, such that F* is

the appropriate AKA challenge-response algorithm. The modification of RAND to RAND* and according creation of XRES* may for instance be performed by an HLR/HSS entity.

[0058] In step 408, the AAA Server sends RAND*, XRES* AUTN, and other parameters as required to the AP which forwards it to the supplicant in step 409, using and EAP-Request/AKA-Challenge message. In step 410, the supplicant checks the network authentication AUTN. The supplicant derives its authentication response value RES using a function F on the value RAND*, in analogy to step 310a of FIG. 3. The function F may correspond to the challenge-response algorithm of AKA. The supplicant derives the Pairwise Master Key PMK.

[0059] In step 411, the supplicant provides RES to the AP with an EAP-Response message. In step 412, the AP sends the EAP Response/AKA-Challenge packet to the AAA Server. In step 413, the AAA Server compares XRES* to the received RES and, on success, derives the PMK.

[0060] In step 414, if all checks are successful, the AAA Server sends the message Access-Accept to AP and includes the keying material PMK for the AP. In step 415, the AP informs the supplicant about the successful authentication with the EAP Success message. In step 416, the supplicant and AP perform 802.1x key negotiation (4-way handshake) and derive the Pairwise Transient Key PTK at both ends. In step 417, the AP sends a message EAP-Request/Reconcile to the supplicant, indicating that it wishes to reconcile the previously collected PHY channel bit sequences and add physical channel based protection to secure the channel. In step 418, the supplicant responds sending the bit sequence J, protected by PTK, to AP.

[0061] In step 419, the AP checks that C=s(J) (analogous to step 208b in FIG. 2), and, if successful, creates an indication ind (analogous to step 209 of FIG. 2) and a reconciled sequence J*. The indication provides the information used for the reconciliation of the bit sequences. Then, AP modifies PTK to PTK* using J*, for instance using an HMAC or other one-way function. In some embodiments, PTK*=HMAC(PTK,J*) In step 420, the AP sends the indication ind, protected by PTK, to the supplicant.

[0062] In step 421, the supplicant uses the indication ind to reconcile its own bit sequence and obtain J*. The supplicant uses J* to modify PTK into PTK* in the same way as the AP. In step 422, the supplicant sends a success message to the AP.

[0063] The supplicant and the AP can now communicate on the secure channel protected by PTK*. One method that may be used to modify the wireless keys (PMK) using information-theoretically secret bits derived from the physical channel as in steps 419, 421 above is the technique described in S. Mathur, A. Reznik, C. Ye, R. Mukherjee, A. Rahman, Y. Shah, W.

Trappe and N. Mandayam, "Exploiting the Physical Layer for Enhanced Security," in IEEE Wireless Communications Magazine, vol. 17, pg. 63-70, Oct. 2010.

[0064] Note that various hardware elements of one or more of the described embodiments are referred to as "modules" that carry out (i.e., perform, execute, and the like) various functions that are described herein in connection with the respective modules. As used herein, a module includes hardware (e.g., one or more processors, one or more microprocessors, one or more microcontrollers, one or more microchips, one or more application-specific integrated circuits (ASICs), one or more field programmable gate arrays (FPGAs), one or more memory devices) deemed suitable by those of skill in the relevant art for a given implementation. Each described module may also include instructions executable for carrying out the one or more functions described as being carried out by the respective module, and it is noted that those instructions could take the form of or include hardware (i.e., hardwired) instructions, firmware instructions, software instructions, and/or the like, and may be stored in any suitable non-transitory computer-readable medium or media, such as commonly referred to as RAM, ROM, etc.

[0065] Exemplary embodiments disclosed herein are implemented using one or more wired and/or wireless network nodes, such as a wireless transmit/receive unit (WTRU) or other network entity.

[0066] FIG. 5 is a system diagram of an exemplary WTRU 502, which may be employed as a communications device in embodiments described herein. As shown in FIG. 5, the WTRU 502 may include a processor 518, a communication interface 519 including a transceiver 520, a transmit/receive element 522, a speaker/microphone 524, a keypad 526, a display/touchpad 528, a non-removable memory 530, a removable memory 532, a power source 534, a global positioning system (GPS) chipset 536, and sensors 538. It will be appreciated that the WTRU 502 may include any sub-combination of the foregoing elements while remaining consistent with an embodiment.

[0067] The processor 518 may be a general purpose processor, a special purpose processor, a conventional processor, a digital signal processor (DSP), a plurality of microprocessors, one or more microprocessors in association with a DSP core, a controller, a microcontroller, Application Specific Integrated Circuits (ASICs), Field Programmable Gate Array (FPGAs) circuits, any other type of integrated circuit (IC), a state machine, and the like. The processor 518 may perform signal coding, data processing, power control, input/output processing, and/or any other functionality that enables the WTRU 502 to operate in a wireless environment. The processor 518 may be coupled to the transceiver 520, which may be coupled to the

transmit/receive element 522. While FIG. 5 depicts the processor 518 and the transceiver 520 as separate components, it will be appreciated that the processor 518 and the transceiver 520 may be integrated together in an electronic package or chip.

[0068] The transmit/receive element 522 may be configured to transmit signals to, or receive signals from, a base station over the air interface 516. For example, in one embodiment, the transmit/receive element 522 may be an antenna configured to transmit and/or receive RF signals. In another embodiment, the transmit/receive element 522 may be an emitter/detector configured to transmit and/or receive IR, UV, or visible light signals, as examples. In yet another embodiment, the transmit/receive element 522 may be configured to transmit and receive both RF and light signals. It will be appreciated that the transmit/receive element 522 may be configured to transmit and/or receive any combination of wireless signals.

[0069] In addition, although the transmit/receive element 522 is depicted in FIG. 5 as a single element, the WTRU 502 may include any number of transmit/receive elements 522. More specifically, the WTRU 502 may employ MIMO technology. Thus, in one embodiment, the WTRU 502 may include two or more transmit/receive elements 522 (e.g., multiple antennas) for transmitting and receiving wireless signals over the air interface 516.

[0070] The transceiver 520 may be configured to modulate the signals that are to be transmitted by the transmit/receive element 522 and to demodulate the signals that are received by the transmit/receive element 522. As noted above, the WTRU 502 may have multi-mode capabilities. Thus, the transceiver 520 may include multiple transceivers for enabling the WTRU 502 to communicate via multiple RATs, such as UTRA and IEEE 802.11, as examples.

[0071] The processor 518 of the WTRU 502 may be coupled to, and may receive user input data from, the speaker/microphone 524, the keypad 526, and/or the display/touchpad 528 (e.g., a liquid crystal display (LCD) display unit or organic light-emitting diode (OLED) display unit). The processor 518 may also output user data to the speaker/microphone 524, the keypad 526, and/or the display/touchpad 528. In addition, the processor 518 may access information from, and store data in, any type of suitable memory, such as the non-removable memory 530 and/or the removable memory 532. The non-removable memory 530 may include random-access memory (RAM), read-only memory (ROM), a hard disk, or any other type of memory storage device. The removable memory 532 may include a subscriber identity module (SIM) card, a memory stick, a secure digital (SD) memory card, and the like. In other embodiments, the processor 518 may access information from, and store data in, memory that is not physically located on the WTRU 502, such as on a server or a home computer (not shown).

[0072] The processor 518 may receive power from the power source 534, and may be configured to distribute and/or control the power to the other components in the WTRU 502. The power source 534 may be any suitable device for powering the WTRU 502. As examples, the power source 534 may include one or more dry cell batteries (e.g., nickel-cadmium (NiCd), nickel-zinc (NiZn), nickel metal hydride (NiMH), lithium-ion (Li-ion), and the like), solar cells, fuel cells, and the like.

[0073] The processor 518 may also be coupled to the GPS chipset 536, which may be configured to provide location information (e.g., longitude and latitude) regarding the current location of the WTRU 502. In addition to, or in lieu of, the information from the GPS chipset 536, the WTRU 502 may receive location information over the air interface 516 from a base station and/or determine its location based on the timing of the signals being received from two or more nearby base stations. It will be appreciated that the WTRU 502 may acquire location information by way of any suitable location-determination method while remaining consistent with an embodiment.

[0074] The processor 518 may further be coupled to other peripherals 538, which may include one or more software and/or hardware modules that provide additional features, functionality and/or wired or wireless connectivity. For example, the peripherals 538 may include sensors such as an accelerometer, an e-compass, a satellite transceiver, a digital camera (for photographs or video), a universal serial bus (USB) port, a vibration device, a television transceiver, a hands free headset, a Bluetooth® module, a frequency modulated (FM) radio unit, a digital music player, a media player, a video game player module, an Internet browser, and the like.

[0075] FIG. 6 depicts an exemplary network entity 690 that may be used in embodiments of the present disclosure, for example as an exemplary authentication center or AAA server. As depicted in FIG. 6, network entity 690 includes a communication interface 692, a processor 694, and non-transitory data storage 696, all of which are communicatively linked by a bus, network, or other communication path.

[0076] Communication interface 692 may include one or more wired communication interfaces and/or one or more wireless-communication interfaces. With respect to wired communication, communication interface 692 may include one or more interfaces such as Ethernet interfaces, as an example. With respect to wireless communication, communication interface 692 may include components such as one or more antennae, one or more transceivers/chipsets designed and configured for one or more types of wireless (e.g., LTE)

communication, and/or any other components deemed suitable by those of skill in the relevant art. And further with respect to wireless communication, communication interface 692 may be equipped at a scale and with a configuration appropriate for acting on the network side—as opposed to the client side—of wireless communications (e.g., LTE communications, Wi-Fi communications, and the like). Thus, communication interface 692 may include the appropriate equipment and circuitry (perhaps including multiple transceivers) for serving multiple mobile stations, UEs, or other access terminals in a coverage area.

[0077] Processor 694 may include one or more processors of any type deemed suitable by those of skill in the relevant art, some examples including a general-purpose microprocessor and a dedicated DSP.

[0078] Data storage 696 may take the form of any non-transitory computer-readable medium or combination of such media, some examples including flash memory, read-only memory (ROM), and random-access memory (RAM) to name but a few, as any one or more types of non-transitory data storage deemed suitable by those of skill in the relevant art could be used. As depicted in FIG. 6, data storage 696 contains program instructions 697 executable by processor 694 for carrying out various combinations of the various network-entity functions described herein.

[0079] Although features and elements are described above in particular combinations, one of ordinary skill in the art will appreciate that each feature or element can be used alone or in any combination with the other features and elements. In addition, the methods described herein may be implemented in a computer program, software, or firmware incorporated in a computer-readable medium for execution by a computer or processor. Examples of computer-readable storage media include, but are not limited to, a read only memory (ROM), a random access memory (RAM), a register, cache memory, semiconductor memory devices, magnetic media such as internal hard disks and removable disks, magneto-optical media, and optical media such as CD-ROM disks, and digital versatile disks (DVDs). A processor in association with software may be used to implement a radio frequency transceiver for use in a WTRU, UE, terminal, base station, RNC, or any host computer.

CLAIMS

1. A method comprising:

operating a receiving device to measure dynamic channel conditions between the receiving device and an authenticating device;

receiving a challenge value and an obfuscated key derivation seed from the authenticating device;

determining a valid challenge response to the challenge value;

using the valid challenge response, removing the obfuscation from the key derivation seed;

generating a key for communication between the receiving device and the authenticating device based at least in part on (i) the measured channel conditions and (ii) at least a selected portion of the key derivation seed.

2. The method of claim 1 further comprising generating a bit sequence from the dynamic channel measurements, wherein the key is generated based on values at bit positions in the bit sequence, the bit positions being identified by the selected portion of the key derivation seed.

3. The method of claim 1, further comprising validating the received challenge value based on the key derivation seed.

4. The method of claim 3, wherein validating the received challenge value includes applying a predetermined function to the key derivation seed and comparing the result to the received challenge value.

5. The method of claim 4, wherein the predetermined function includes bit permutation.

6. The method of claim 1, wherein the authenticating device has an identifier, and wherein the valid challenge response depends on the received challenge value and the identifier of the authenticating device.

7. The method of claim 6, wherein determining a valid challenge response includes selecting a function associated with the identifier and applying the selected function to the challenge value.

8. The method of claim 7, wherein the selected function emulates a physically unclonable function of the authenticating device.

9. The method of claim 6, wherein determining a valid challenge response includes:

sending the identifier and the challenge value to an authentication center; and

receiving the valid challenge response from the authentication center.

10. The method of claim 1, wherein removing the obfuscation includes performing a bitwise XOR operation on the valid challenge response and the key derivation seed.

11. The method of claim 1, wherein the selected portion of the key derivation seed is selected by a method comprising reconciling the received key derivation seed with a locally-generated key derivation seed.

12. The method of claim 1, wherein the received key derivation seed identifies a plurality of values derived from the measurement of channel conditions.

13. The method of claim 1, wherein generating the key comprises:

selecting a portion of the channel condition measurements using the selected portion of the key derivation seed; and

generating a keyed hash message authentication code (HMAC) to a pairwise transient key (PTK) using the selected portion of the channel condition measurements as a hash key.

14. The method of claim 1, further comprising, at the authenticating device:

operating the authenticating device to measure the dynamic channel conditions between the receiving device and the authenticating device;

generating a bit sequence from the dynamic channel measurements;

generating the key derivation seed, wherein the key derivation seed identifies a plurality of bit positions in the bit sequence;

generating the challenge value;

determining the valid challenge response based on the challenge value;

obfuscating the key derivation seed using the valid challenge response; and

sending to the receiving device the obfuscated key derivation seed and the challenge value.

15. The method of claim 14, wherein generating the challenge value comprises applying a predetermined function to the key derivation seed.

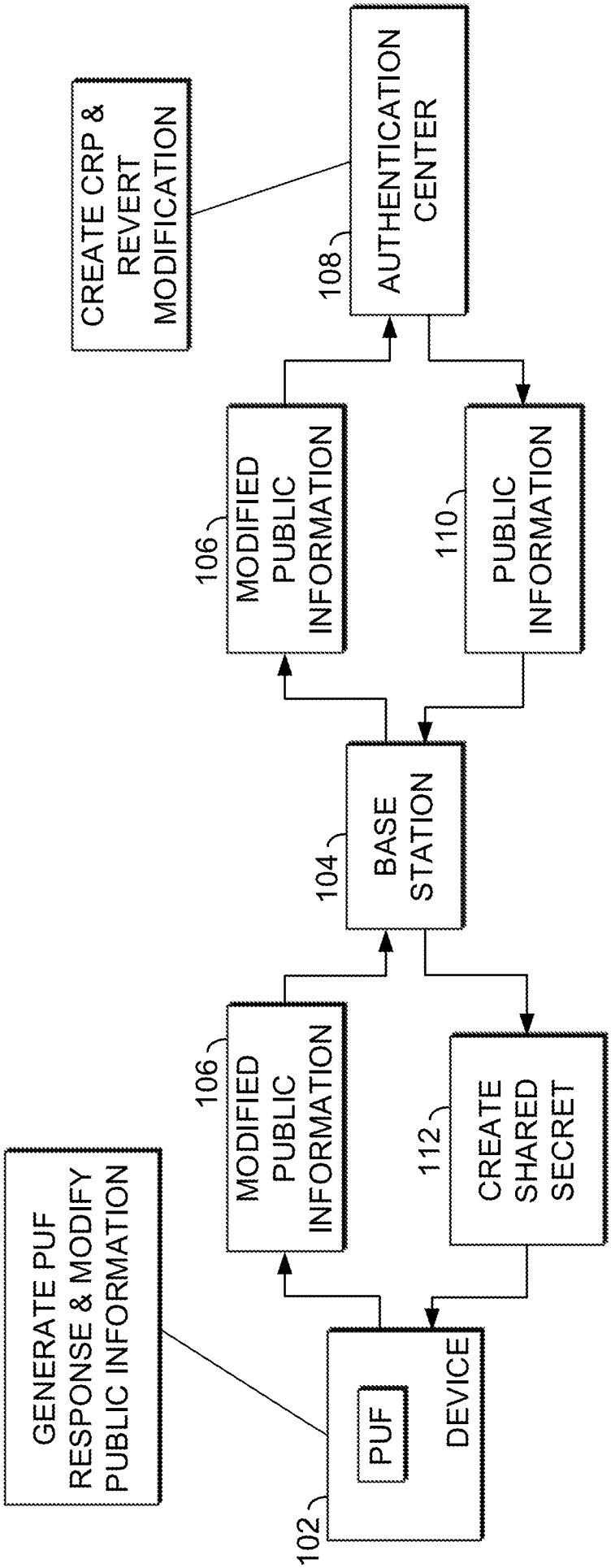
CREATE CRP & REVERT MODIFICATION
108
AUTHENTICATION CENTER
106
MODIFIED PUBLIC INFORMATION
110
PUBLIC INFORMATION
104
BASE STATION
106
MODIFIED PUBLIC INFORMATION
112
CREATE SHARED SECRET
GENERATE PUF RESPONSE & MODIFY PUBLIC INFORMATION
PUF
DEVICE
102

FIG. 1

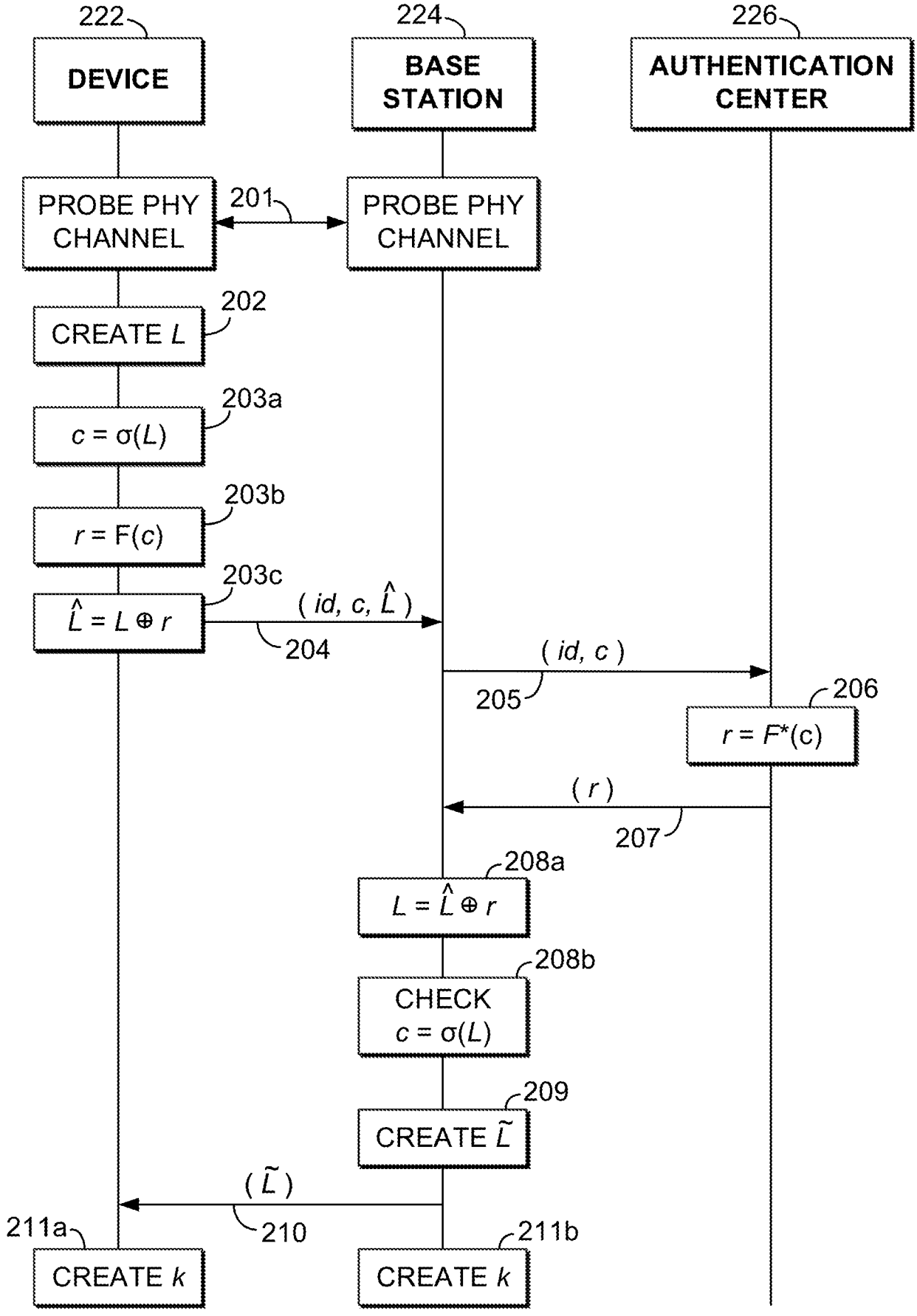
222
DEVICE
224
BASE STATION
226
AUTHENTICATION CENTER
PROBE PHY CHANNEL
201
PROBE PHY CHANNEL
CREATE L
202
c = σ(L)
203a
r = F(c)
203b
L̂ = L ⊕ r
203c
(id, c, L̂)
204
(id, c)
205
r = F*(c)
206
(r)
207
L = L̂ ⊕ r
208a
CHECK c = σ(L)
208b
CREATE L̃
209
(L̃)
210
CREATE k
211a
CREATE k
211b

FIG. 2

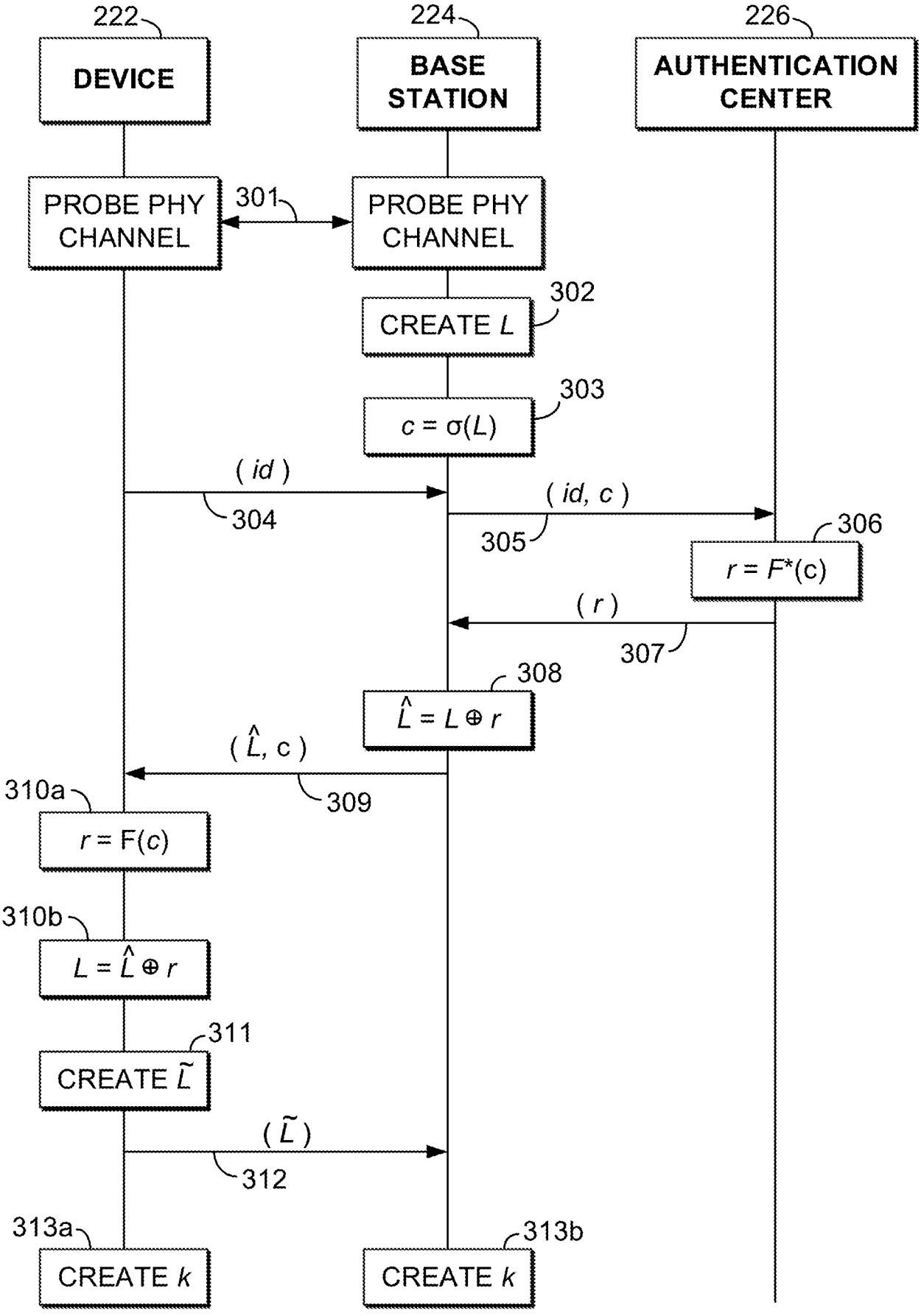
222
DEVICE
224
BASE STATION
226
AUTHENTICATION CENTER
PROBE PHY CHANNEL
301
PROBE PHY CHANNEL
302
CREATE L
303
c = σ(L)
(id)
304
(id, c)
305
306
r = F*(c)
(r)
307
308
L̂ = L ⊕ r
(L̂, c)
309
310a
r = F(c)
310b
L = L̂ ⊕ r
311
CREATE L̃
(L̃)
312
313a
CREATE k
313b
CREATE k

FIG. 3

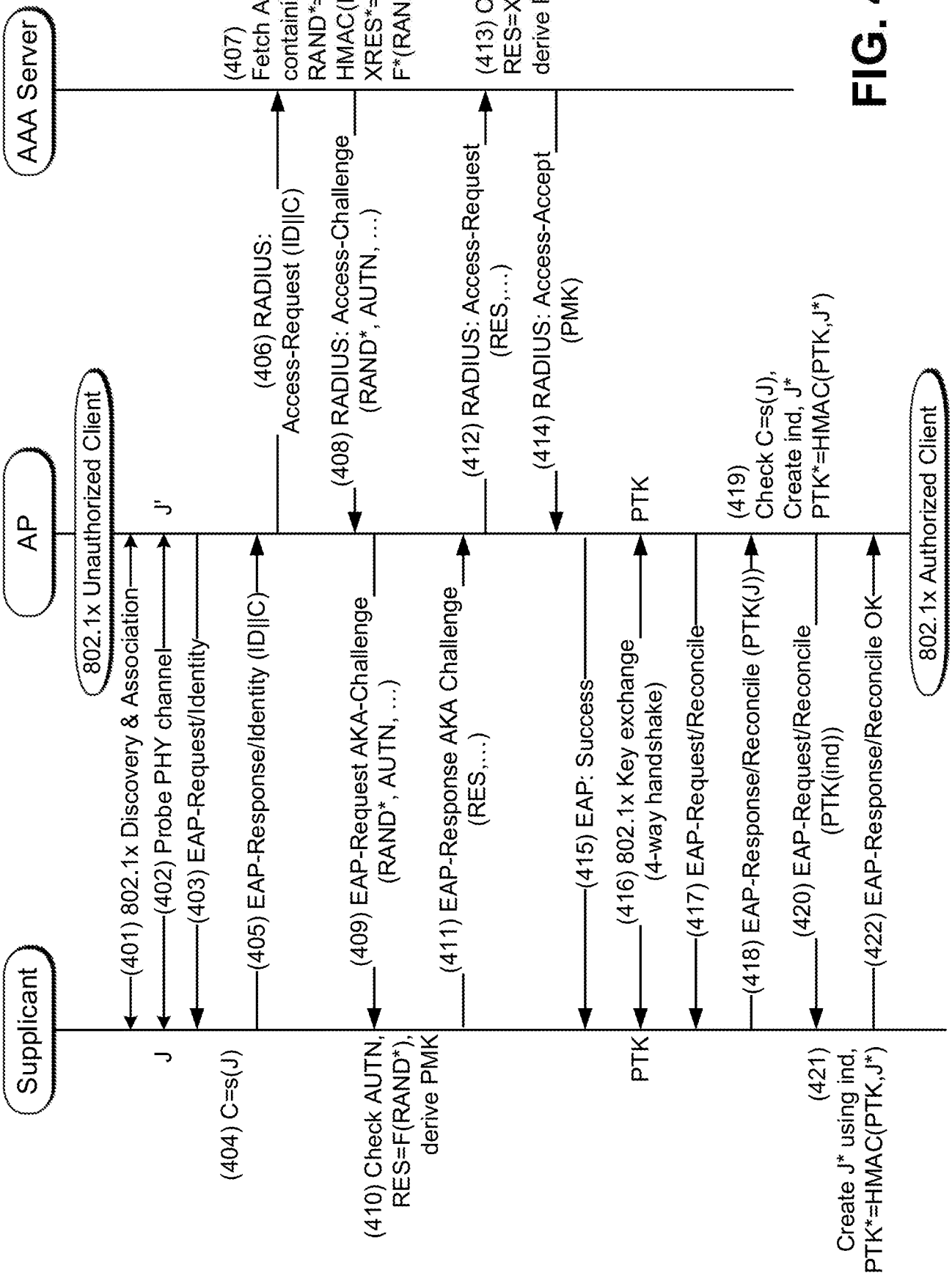
Supplicant
AP
802.1x Unauthorized Client
AAA Server
(401) 802.1x Discovery & Association
(402) Probe PHY channel
J'
J
(403) EAP-Request/Identity
(404) C=s(J)
(405) EAP-Response/Identity (ID||C)
(406) RADIUS: Access-Request (ID||C)
(407) Fetch AV for ID||C containing AUTN RAND*= HMAC(RAND,C), XRES*= F*(RAND*)
(408) RADIUS: Access-Challenge (RAND*, AUTN, ...)
(409) EAP-Request AKA-Challenge (RAND*, AUTN, ...)
(410) Check AUTN, RES=F(RAND*), derive PMK
(411) EAP-Response AKA Challenge (RES,...)
(412) RADIUS: Access-Request (RES,...)
(413) Check RES=XRES*; derive PMK
(414) RADIUS: Access-Accept (PMK)
(415) EAP: Success
(416) 802.1x Key exchange (4-way handshake)
PTK
PTK
(417) EAP-Request/Reconcile
(418) EAP-Response/Reconcile (PTK(J))
(419) Check C=s(J), Create ind, J* PTK*=HMAC(PTK,J*)
(420) EAP-Request/Reconcile (PTK(ind))
(421) Create J* using ind, PTK*=HMAC(PTK,J*)
(422) EAP-Response/Reconcile OK
802.1x Authorized Client

FIG. 4

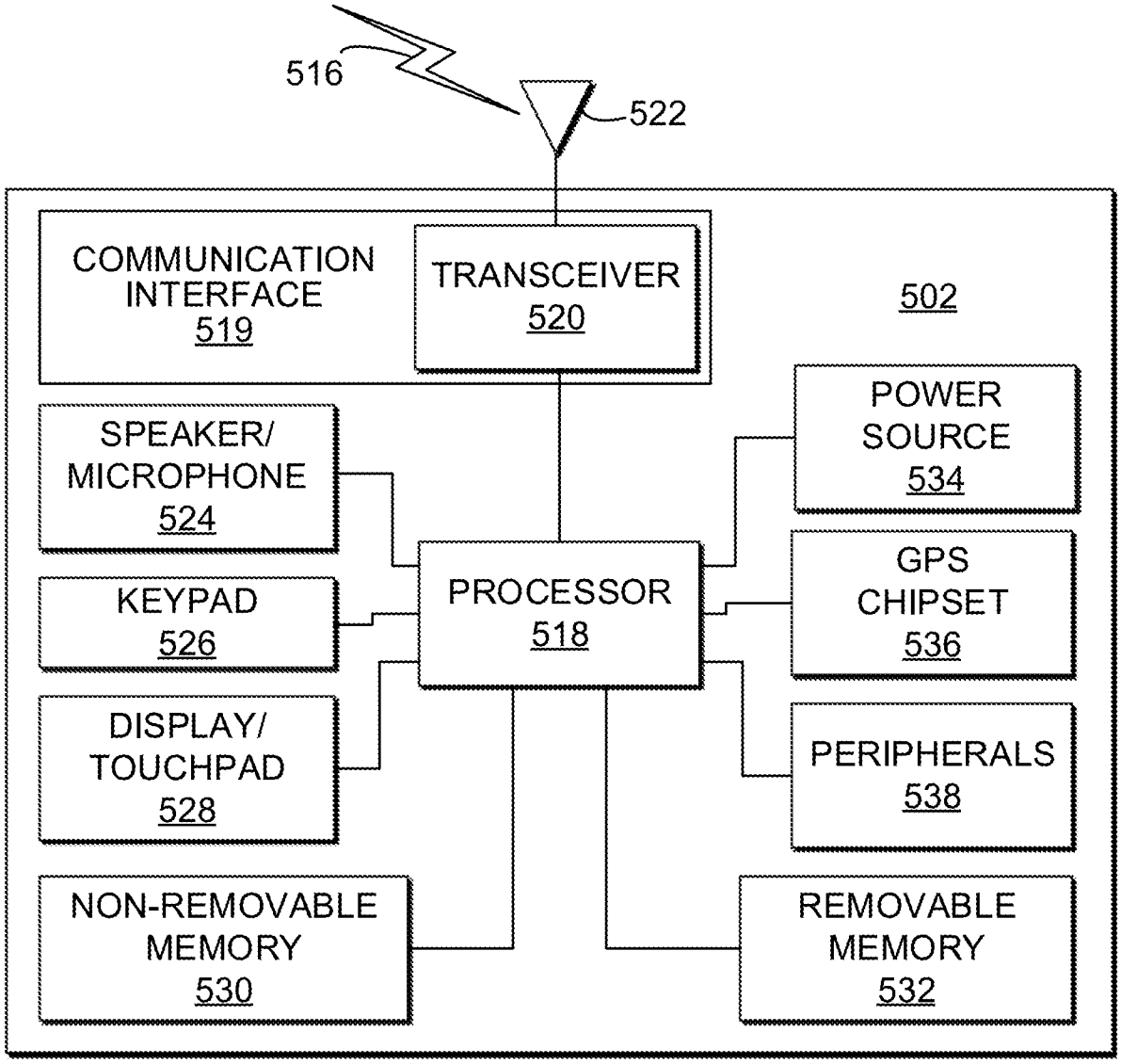
516
522
COMMUNICATION INTERFACE 519
TRANSCEIVER 520
502
SPEAKER/ MICROPHONE 524
KEYPAD 526
DISPLAY/ TOUCHPAD 528
NON-REMOVABLE MEMORY 530
PROCESSOR 518
POWER SOURCE 534
GPS CHIPSET 536
PERIPHERALS 538
REMOVABLE MEMORY 532

FIG. 5

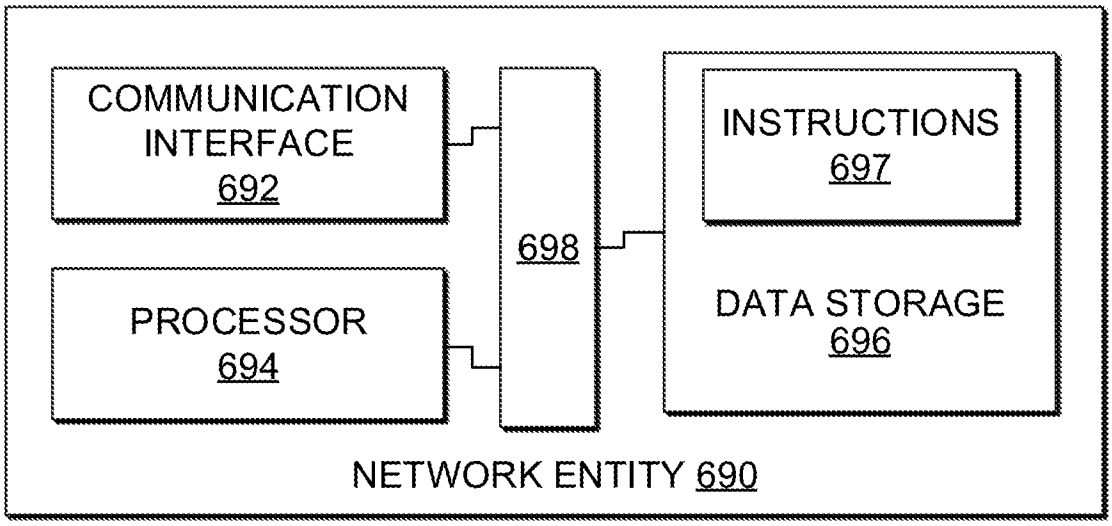
COMMUNICATION INTERFACE 692
PROCESSOR 694
698
INSTRUCTIONS 697
DATA STORAGE 696
NETWORK ENTITY 690

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2017/029436

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L9/08 H04L9/32
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	HUTH CHRISTOPHER ET AL: "Securing systems on the Internet of Things via physical properties of devices and communications", 2015 ANNUAL IEEE SYSTEMS CONFERENCE (SYSCON) PROCEEDINGS, IEEE, 13 April 2015 (2015-04-13), pages 8-13, XP032782394, DOI: 10.1109/SYSCON.2015.7116721 [retrieved on 2015-06-02] abstract page 3, left-hand column, line 17 - page 4, right-hand column, line 40; figures 3-6 ----- -/--	1-15

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
14 July 2017	26/07/2017

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Spranger, Stephanie

2

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CHUNXUAN YE ET AL: "Information-theoretically Secret Key Generation for Fading Wireless Channels", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 27 October 2009 (2009-10-27), XP080374058, DOI: 10.1109/TIFS.2010.2043187 abstract page 7, line 1 - page 23, line 22 -----	1-15
A	JORGE GUAJARDO ET AL: "Anti-counterfeiting, key distribution, and key storage in an ambient world via physical unclonable functions", INFORMATION SYSTEMS FRONT, KLUWER, DORDRECHT, NL, vol. 11, no. 1, 1 March 2009 (2009-03-01), pages 19-41, XP007909023, ISSN: 1387-3326, DOI: 10.1007/S10796-008-9142-Z abstract page 22, right-hand column, line 1 - page 34, left-hand column, line 37; figure 8 -----	1-15