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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
15 December 2016 (15.12.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/200659 A1

(51) International Patent Classification:
*H04L 9/08* (2006.01) *H04L 29/06* (2006.01)
*H04L 29/08* (2006.01)

(21) International Application Number:
PCT/US2016/035282

(22) International Filing Date:
1 June 2016 (01.06.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

| | | |
|---|---|---|
| 62/172,893 | 9 June 2015 (09.06.2015) | US |
| 62/195,409 | 22 July 2015 (22.07.2015) | US |
| 14/968,125 | 14 December 2015 (14.12.2015) | US |


(71) Applicant: **INTEL CORPORATION** [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95054 (US).

(72) Inventor: **SMITH, Ned M.**; 375 SW Delta Drive, Beaverton, Oregon 97006 (US).

(74) Agents: **RICHARDS, Edwin E.** et al.; Trop, Pruner & Hu, P.C., 1616 S. Voss Rd., Ste. 750, Houston, Texas 77057-2631 (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, 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, KN, KP, KR, 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, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

**Declarations under Rule 4.17**:

— *as to the identity of the inventor (Rule 4.17(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

*[Continued on next page]*

(54) **Title**: A SELF-CONFIGURING KEY MANAGEMENT SYSTEM FOR AN INTERNET OF THINGS NETWORK

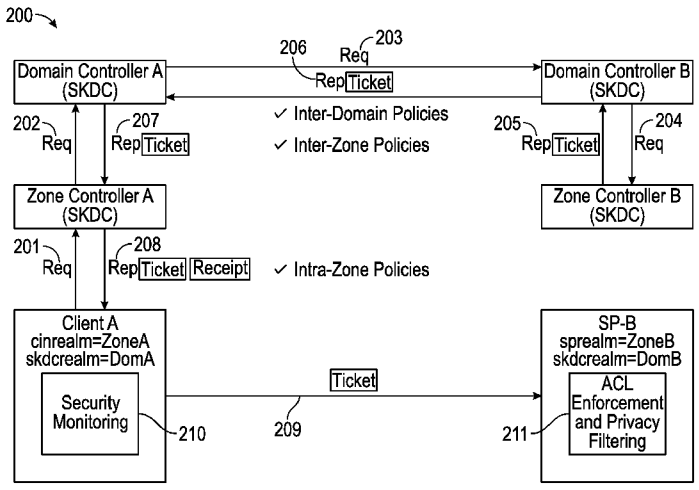



FIG. 2

(57) **Abstract**: In one embodiment, a method includes receiving, in a first device, at least one of a first symmetric key and a first asymmetric key in a common key management structure, the common key management structure to accommodate asymmetric keys and symmetric keys, and further including security policy information to enable communication between the first device of a first domain of an Internet of Things (IoT) network and a second device of a second domain of the IoT network according to an inter-domain security policy; and sending a first message directly from the first device to the second device according to the security policy information of the common key management structure. Other embodiments are described and claimed.

WO 2016/200659 A1

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

**Published**:

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

# A Self-Configuring Key Management System for an Internet of Things Network

## Background

[0001] Internet of Things (IoT) networks may range dramatically in size and complexity from a single sensor connected to a smart device to smart city networks having hundreds of thousands of interconnected nodes. Interoperability between these IoT networks is an issue. For example, different IoT frameworks may incorporate differing device management schemes. Those schemes may include extreme variances in approaches to network and device security. Those schemes may also use different mechanisms for scaling up or down (i.e., adding or subtracting nodes from the IoT networks) according to site specific deployment expectations.

## Brief Description of the Drawings

[0002] Features and advantages of embodiments of the present invention will become apparent from the appended claims, the following detailed description of one or more example embodiments, and the corresponding figures. Where considered appropriate, reference labels have been repeated among the figures to indicate corresponding or analogous elements.

[0003] FIG. 1 is a timing diagram illustrating a key management system in accordance with an embodiment.

[0004] FIG. 2 is a block diagram of a portion of an IoT network in accordance with an embodiment.

[0005] FIG. 3 is a common key management context structure in accordance with an embodiment.

[0006] FIG. 4 is an example key management context structure that may be used in connection with an asymmetric key.

[0007] FIG. 5 is a block diagram of an example system with which embodiments can be used.

[0008] FIG. 6 is a block diagram of a system in accordance with another embodiment of the present invention.

[0009] FIG. 7 is a block diagram of a system in accordance with another embodiment of the present invention.

Detailed Description

[0010] In the following description, numerous specific details are set forth but embodiments of the invention may be practiced without these specific details. Well-known circuits, structures and techniques have not been shown in detail to avoid obscuring an understanding of this description. “An embodiment”, “various embodiments” and the like indicate embodiment(s) so described may include particular features, structures, or characteristics, but not every embodiment necessarily includes the particular features, structures, or characteristics. Some embodiments may have some, all, or none of the features described for other embodiments. “First”, “second”, “third” and the like describe a common object and indicate different instances of like objects are being referred to. Such adjectives do not imply objects so described must be in a given sequence, either temporally, spatially, in ranking, or in any other manner.

[0011] In various embodiments, techniques are provided to enable scalable key management in an IoT network. This allows for secure IoT networks even as those networks grow in size (e.g., number of nodes), complexity (e.g., hierarchies of command among the nodes), and diversity (e.g., IoT network having different nodes from different vendors). More specifically, embodiments provide a key management system that is integrated with the dynamics of IoT network evolution. Given that IoT devices can have widely disparate compute and other capabilities (e.g., processing power and/or storage capacity), and may further have multiple dissimilar transport and network technologies, embodiments provide a common key management system that can implement symmetric cryptography, asymmetric cryptography, or both. To this end, a key management system as described herein can issue both symmetric and asymmetric keys under a common convention for organizing IoT devices and for evolving the IoT network.

[0012] In an embodiment IoT devices perform key management functions in proportion to the complexity of the network. As network complexity increases, key management duties expand to accommodate need. Key management duties can be

assignable by role privileges that may be assigned as part of on-boarding a device into the network. This key management model allows both ad-hoc and introducer models for extending a network. Nevertheless, both models share the same mechanisms for assigning context (e.g., administrative realm, publish-subscribe topic) to keys. This allows a credential that originated using an ad-hoc key agreement protocol (e.g. Diffie-Hellman or PAKE) to be used with a device that understands introducer methods (e.g. PKI, Kerberos) and vice versa.

[0013] In some embodiments, both symmetric and asymmetric keys may be included in a credential so that the constraints of the underlying transport and network layers may be accommodated if security is applied at lower layers. If security is applied at the application layer, the same credentials may be used having consistent security, scope and use semantics.

[0014] For example, IoT frameworks such as the Open Interconnect Consortium (OIC) and PC networking framework WinSOCK define a multi-protocol transport or session layering scheme where one type of security is assumed by a first transport and a second type of security is assumed by a second transport. In a scenario where a first IoT device using a first transport connects to a second IoT device supporting both a first and a second transport, the second IoT device will select the credential (symmetric) that enables secure communication with the first device. Whereas, a third IoT device using a second transport will require the 2nd IoT device to select the credential (certificate) suited for the second transport. If a system of devices supporting multiple transports have occasion to connect where there are multiple transports in common, a selection or negotiation of which transport and which credential to use is made. This leads to the case where devices will have multiple credentials that enable secure communication to the same peer device which may lead to a negotiation over which credential type shall be used for a given connection. In such an environment, consistency of security context across differing credential types is necessary to ensure security policies are uniformly applied.

[0015] Note that an embodiment of a key management system can be instantiated on a single network node and expand to be shared across multiple nodes as new nodes are added to a network. The key management system (or a given one of

multiple instantiated key management devices of the system) can issue credentials that may be used to authenticate and protect devices, data, network services, network zones and various grouping constructs.

[0016] Embodiments may integrate multiple approaches under a common infrastructure and provide a mechanism for delegating various IoT behaviors including publish-subscribe, peer-peer, network management and key management function to its members. As such, embodiments can enable use of asymmetric keys (such as typically managed using Public Key Infrastructure (PKI)), symmetric keys (such as typically managed using Kerberos), and further enable ad hoc approaches to key management such as using Diffie-Hellman key agreement to produce symmetric keys between network peers. Thus, embodiments provide a common framework for interoperability and migration between these different approaches to key management.

[0017] Referring now to FIG. 1, shown is a process 100 illustrating a key management system in accordance with an embodiment. As seen in FIG. 1, a key management system D1, which may be a server computer or any other type of computing device, bootstraps an IoT network using both ad hoc and third party introduction techniques. More specifically, FIG. 1 shows a timing illustration of key management operations for a plurality of devices in an IoT network accordance with an embodiment.

[0018] In the particular implementation shown, assume an initial key management system D1 (e.g., Smartphone) possesses all key management roles at time t1. Thereafter, at time t2 a new device D2 is introduced and an ad hoc key management process (block 101) is performed to provide one or more keys (a symmetric key, an asymmetric key, or both) to device D2 in a common key management context structure. This ad hoc key establishment may occur via a Diffie-Hellman key exchange, and the like.

[0019] At time t3 a new device D3 is introduced to D1 and an ad hoc key management process (block 102) is performed to provide one or more keys (a symmetric key, an asymmetric key, or both) to device D3 in a common key

management context structure. At time t3, device D3 is also introduced (as a zone controller) to device D2 (block 103). For example, an embodiment is based on the Fluffy key management system. For example, D3 may be introduced to the first domain, to which D1 is the domain controller, by sending a request "PSK-REQ" to a key manager (e.g., device D1) for a ticket. As used herein, an example of a "request" and a "reply" include "GSK-REQ"/"PSK-REQ" and "GSK-REP"/"PSK-REP", however "GSK-FET"/"PSK-FET" may also be considered a request and "GSK-DEL"/"PSK-DEL" may also be considered a reply. Further details regarding such requests and replies are located at, for example, https://tools.ietf.org/id/draft-hardjono-ace-fluffy-00***txt (see also (see ***https://datatracker.ietf.org/doc/draft-hardjono-ace-fluffy/ for version 01). The reply to such a request may use a client permissions field (cpac: the permissions and access control (PAC) structure containing permissions granted to a client associated with a key enclosed key in receipt sent to the client). The same tools (e.g., access controls structures) may be used to designate D2 as a mere client in zone controlled by Zone Controller D3. Afterwards, D2 and D3 can communicate directly (block 104). Therefore, blocks 101, 102 include ad hoc exchanges while block 103 includes an introducer method to introduce two nodes. This intruder method will be further explained with an embodiment addressed in Figure 2.

[0020] Then at a still later time t4, another device D4 is introduced into the IoT network (block 105). Specifically, device D3, which is a zone controller, performs an ad hoc key establishment process with this device such that thereafter devices D2 and D4 can interact directly (block 107), after device D3 introduces device D4 to D2 (block 106). D4 and D2 are introduced to each other as clients (e.g., using tickets and receipts with access control lists and "cpac" designations, if using the Fluffy protocol, designated D2 and D4 as clients).

[0021] Understand while shown with these particular examples in FIG. 1, other interactions and key management flows can occur.

[0022] As the network grows in complexity, new devices may be delegated a key management responsibility (e.g., device D3 is provisioned dynamically as a zone

controller at block 103 of FIG. 1). Other devices may be introduced as peers (e.g., block 106 of FIG. 1) that may interact directly using supplied credentials.

[0023] Note that credentials may be used to assign other roles (e.g., something other than "zone controller" or "client" as shown in FIG. 1) meaningful in the context of an IoT application/function. For example, devices may cooperate in the context of a publish-subscribe interaction where some devices may be given a publisher role while other devices may be assigned a subscriber role, and still other devices provided a broker role (e.g., using the "pac" designation if the Fluffy protocol is used). All roles may be issued under a credential that specifies a topic of interaction (e.g., using "crlm" or "crealm" realm entry if the Fluffy protocol is used) such that off-topic interactions may require a different credential, allowing the devices to function in different roles for the other topic.

[0024] In embodiments, a network topology can be flexibly partitioned to allow intra-zone interactions to receive minimal intermediation to apply security, privacy and safety constraints. Stated another way, a flexible re-intermediation of key management between various domains and zones may occur. Understand that a domain is a logical collection of devices under a common administrative control. In turn, a zone is a logical construct within a given domain to enable scaling of key management within the domain to logically combine a smaller collection of devices to work within a common boundary, referred to as a zone.

[0025] Understand that in different embodiments, various security policies may be identified on a zone or domain basis. In one example case, free interaction may occur within a zone, while some type of security policy may be enforced on communications between zones. Still further, greater security policies may be enforced for communications between different domains.

[0026] In any case, embodiments provide flexible and dynamic scaling of security in IoT networks, while enabling peer-to-peer communication between devices within one or more zones within a domain, as well as between devices present in one or more different zones of different domains.

[0027] Referring now to FIG. 2, shown is a block diagram of a portion of an IoT network in accordance with an embodiment. As shown, multiple devices are present in 2 different domains, namely domain A that includes a domain controller A, which may be a simple key distribution center (SKDC), underneath which a zone controller A is provided (which also may be a SKDC) ,which in turn interacts with a client IoT device of the same domain and zone. Similar devices are present in a second domain B. For example, with the Fluffy protocol the SKDC) is the entity which mediates the establishment of the pair-wise shared key between the client and service principal. Specifically, domain B includes a domain controller B, which may be a SKDC, underneath which a zone controller B is provided (which also may be a SKDC), which in turn interacts with a client IoT device of the same domain and zone.

[0028] As seen in FIG. 2, various interactions can occur in process 200 within and between zones and domains. For example, to enable direct peer-to-peer communication between client device A of the first zone and first domain and a client device B (SP B) of the second domain and the second zone, a request (block 201) can be initiated from client A to its zone controller and domain controller (blocks 202, 203) to in turn receive a ticket and a receipt (blocks 205, 206, 207, 208) that includes security credentials to control the interaction between these devices. More specifically, client A understands it is to communicate outside its domain via its known hierarchy, the Zone Controller A and the Domain Controller A. This may be enforced via client permissions (cpac in the Fluffy protocol) applied to Client A when Client A was on-boarded to domain A by Zone Controller A (which may be a SKDC). Thus, Client A may form a request, such as “PSK-REQ” in the Fluffy protocol, (block 201). The request may include fields such as, for example, Key Type (kty)(this denotes the type of key being requested by the sender (client or SP) of the request), Desired algorithm (etype)(this is the algorithm that is desired (or supported) by the sender of the request (client or SP); SKDC options (these are flags that are intended for the SKDC only); SKDCs realm (skdcrealm) (this the name of the realm, domain or zone of the SKDC); SKDC’s identity (skdcname)(this is the identity of the SKDC); Service principal’s realm (sprealm)(this the name of the realm, domain or zone of the service principal in a PSK-REQ message. In a GSK-REQ message it is the realm of the multicast group); Service principal’s identity (spname)(in a PSK-REQ message

this is the identity of the service principal. In a GSK-REQ message it is the identity or name of the multicast group); Client permissions (cpac)(in a PSK-REQ message this is the permissions desired by the client for itself. In a GSK-REQ message this is the permissions desired by the client for the multicast group). Analogous fields in non-Fluffy protocols may serve the same purpose. The request may be forwarded along to Domain Controller B and Zone Controller B (blocks 202, 203, 204) based on the "sprealm" field indicating SP B is the desired party.

[0029] A reply may be formed, such as "PSK-REP" including a miniticket contains the following (see Figure 3): Issuing SKDC's realm (skdcrealm)(this the name of the realm, domain or zone of the SKDC that issued this miniticket)(304); Issuing SKDC's identity (skdcname)(this is the identity of the SKDC that issued this miniticket); Service principal's realm (sprealm)(this the name of the realm, domain or zone of the service principal for whom this miniticket is destined)(306); Service Principal's identity (spname)(this is the identity of the service principal for whom this miniticket is destined)(305); Encrypted miniticket part (enc-part)(this is the encrypted part of the miniticket intended for the service principal. It is encrypted using the secret key shared pair-wise between the SKDC and the service principal)(307). The encrypted part contains the following (see Fluffy protocol for more details): Client's realm (crealm)(this the name of the realm, domain or zone of the client with whom the receiver of this miniticket shares the enclosed key)(308); Client's identity (cname)(this is the identity of the client with whom the receiver of this miniticket shares the enclosed key)(309); Client permissions (cpac)(this is the permissions and access control (PAC) structure containing permissions granted to the client associated with the enclosed key, and through which inter-domain security policies may be enforced)(310); Time of authentication (authtime)(this is the time at the SKDC when it created this miniticket in response to a request); Expiration time of key (endtime)(this is the expiration time of the key in this miniticket)(311); Service principal permissions (sppac): (this is the permissions and access control (PAC) structure granted to the service principal associated with the enclosed key, through which inter-domain security policies may be enforced)(312); Transited realms (transited)(this is the set of SKDCs and realms that was involved in the issuance of the miniticket for the service principal. This field is used for cross-realm ticket

issuance and may include, for example, Zone Controller B and Domain Controller B)(313); Key data (keydata)(this fields contains the key-data structure that carries the cryptographic key and other parameters necessary for operating the key)(314). This area may include a symmetric key, a public and private key pair, a public key, and/or combinations thereof.

[0030] Receipt 303 may also be returned at blocks 205, 206, 207, 208, 209. Receipt fields 304', 305', 306', 314', 311', 310' are analogous to fields 304, 305, 306, 314, 311, 310 of ticket 302. As provided in the Fluffy protocol, the miniticket is created by the SKDC (e.g., Zone Controller B) and is intended for the service principal (SP B) (although it is delivered via Client A, which initiates the process with the PSK-REQ message). The SKDC responds to the client by sending a PSK-Response (PSK-REP) message containing the miniticket and a receipt. The miniticket contains a copy of the encryption key to be delivered to the service principal by the client in a PSK-Establish (PSK-ESTB) message. As such, the sensitive parts (enc-part) of the miniticket are encrypted using the service principal's secret key (which it shares pair-wise with the SKDC). The receipt is created by the SKDC (e.g., Zone Controller B) and is used for the SKDC to deliver a key to a requesting party (be it client, service principal or multicast group members).

[0031] Figure 4 includes an analogous container but provides an asymmetric pair to Client A. Specifically, reply 401 is analogous to reply 301. Reply 401 includes fields 404, 405, 406, 414, 411, 410, 415 that are analogous to fields 304', 305', 306', 314', 311', 310', 315' (which field 415 including a key pair instead of a symmetric key). In such a case, SP B may inquire with Zone Controller B for a public key corresponding to the asymmetric pair (or a miniticket may be used as described in the Fluffy protocol).

[0032] Note that both client devices may include logic to enforce policy associated with the given devices as identified in the common key management structure that provides appropriate keys for controlling the communications. As such, the enforcement point for applying security policy is in the endpoint devices themselves, reducing complexity and network interaction. In some embodiments, a key provided in such structure can be used to bind a security policy and to create a secure

channel between the devices which may be used for further communications. For example, security monitoring 210 and ACL enforcement and privacy filtering 211 may be tied to keys (see ticket of block 209) via cpac 310', 310 and sppac 312.

[0033] Note that in some embodiments, the common key management structure can be used as a generic ticket format that can be used for data exchanges. In this way, the complexity of setting up an independent TLS session between devices to send the data can be avoided. As an example, data can be securely containerized within this common key management structure (see Figures 3 and 4) and header information within the key management structure can be used to appropriately route packets of a given data communication. Note that a secure data channel can be established between devices responsive to the request for inter-domain (and inter-zone) interaction between the two IoT devices. Inter-zone and inter-domain interactions may require additional vetting when IoT devices cross zonal and domain boundaries. As such, inter-zonal and/or inter-domain interactions may receive added scrutiny as part of key issuance (see, e.g., elements 210, 211).

[0034] Embodiments may include a scenario where a Client B initiates a connection to Client A where a session is established without first obtaining a domain (zone) ticket and receipt. Security enforcement logic may require dynamically obtaining said ticket and receipt but may omit issuance of a session key by a SKDC, instead allowing use of the ephemeral session key as the ticket prescribes. A security benefit of this approach is the SKDC need not be hardened for key storage as a trusted third party. A second benefit is that the session key may be managed more precisely having a ticket context and validity that supersedes a caching policy of an ephemeral session key. The ticket may further prescribe a method for transforming the ephemeral session key into a long-term key using a key derivation function that accepts as input the ephemeral key and attributes of the ticket.

[0035] For example, a first domain may need to establish a secure channel with a second domain with who it does not share a pre-existing trust relationship (e.g., there is no presumption of a common PKI or common Kerberos server). Hence, the domains will need to perform an ad-hoc key agreement using Diffie-Hellman or PAKE (password authenticated key exchange). In either case, there is a

presumption of shared context in which to base trust. It is either context for pre-shared passwords or a context for ensuring there is no man-in-the-middle attacker (e.g., both endpoints have implemented a TEE that can attest its environment to the other).

[0036] Embodiments may include receiving a receipt or ticket containing a public asymmetric key (e.g., FIG. 4) identifying a Client B that may be used to validate a signature created by Client B in connection with a session establishment to Client A. The receipt or ticket may contain similar attributes as described in other embodiments of a ticket or receipt so that a security processor may consistently apply a security policy according to a well-defined domain or zone context.

[0037] If the request for a credential is approved, any zonal and domain considerations can be captured as policies that are applied at the point of interaction between endpoints. For example, Client-A device and SP-B device may engage in inter-domain interactions, while applying appropriate policies. Such policies may include, as representative examples, applying data filtering, device monitoring, bandwidth reservations, and the like to ensure privacy, security and safety objectives for the respective organizations and context are appropriately applied.

[0038] Embodiments may use a common key management context structure that accommodates both symmetric and asymmetric key types. Referring now to FIG. 3, shown is a common key management context structure 300 in accordance with an embodiment. As shown in FIG. 3, a request message captures the network topology context including the zone or domain authority, specific devices having domain/zone authority and constraints of key usage based on the role, topic or informational context.

[0039] Also, as noted above regarding FIG. 3, the structure (which may be provided by a given key management entity to a client device responsive to a request (e.g., for interaction with another device in an IoT network)) may be for a symmetric key. However, other embodiments provide for issuance of both symmetric and asymmetric keys under the common context. This allows the same credential to be

flexibly applied when application, network and transport circumstances call for one or the other cryptographic method.

[0040] Also, as noted above regarding FIG. 4, key management context structure 400 may be used in connection with an asymmetric key. Note that the structure further facilitates provisioning of credentials that may be used to authenticate, encrypt and integrity protect the device or data as appropriate for the intended use, whether symmetric or asymmetric cryptography is supported natively.

[0041] Thus, as described above, embodiments provide many advantages described non-exhaustively as follows. In embodiments, a key management system is self-configuring as the IoT network increases in numbers of devices. For example, an initial key management device may dynamically assign key management roles to other devices (e.g., D1 assigning Zone Controller role to D3 at t3). This may occur as a first device, such as D1, becomes overwhelmed and needs to shed some management responsibilities. And this initial device (e.g., D1) or another device (e.g., D3) can dynamically assign key management roles for delegating key management tasks (e.g., to one or more additional domain and/or zone controllers such that, for example, D3 could have determined D4 was a zone controller for a topic instead of a mere client). In this way, the key management system may further dynamically assign publish-subscribe, group formation, and other interaction patters of IoT networks.

[0042] In addition, using a common key management context structure as described herein, both symmetric and asymmetric keys can be used in equivalent circumstances for authentication and protection of devices and/or data.

[0043] Embodiments may further use a key management system to define a network topology of inter-zonal and inter-domain interaction points where security, safety and privacy considerations may be controlled at the request to cross such boundaries (but before the connection that enables such boundary crossings is established). Thus, a request by Client A to communicate with SP B indicates (based on headers such as “sprealm”) that a cross-domain communication is desired so any

key to facilitate such communication may be based (e.g., sppac and cpac) on the need for policies to be complied with before other communications occur.

[0044] In addition, the key management system may be configured to automatically revoke or rescind roles, privileges and access based on an expiration value (e.g., element 311) and contextual change (e.g., change to crealm 308 or sprealm 306).

[0045] In some embodiments the techniques described herein may be implemented within a trusted execution environment (TEE) such as Intel® SGX, Intel® MemCore, Intel® CSE, Intel® VT-X, Intel® IOT-OS with Smack, or ARM TrustZone, etc. to perform security operations as described herein.

[0046] Referring now to FIG. 5, shown is a block diagram of an example system with which embodiments can be used. As seen, system 900 may be a smartphone or other wireless communicator or any other IoT device. A baseband processor 905 is configured to perform various signal processing with regard to communication signals to be transmitted from or received by the system. In turn, baseband processor 905 is coupled to an application processor 910, which may be a main CPU of the system to execute an OS and other system software, in addition to user applications such as many well-known social media and multimedia apps. Application processor 910 may further be configured to perform a variety of other computing operations for the device.

[0047] In turn, application processor 910 can couple to a user interface/display 920, e.g., a touch screen display. In addition, application processor 910 may couple to a memory system including a non-volatile memory, namely a flash memory 930 and a system memory, namely a DRAM 935. In some embodiments, flash memory 930 may include a secure portion 932 in which secrets and other sensitive information may be stored. As further seen, application processor 910 also couples to a capture device 945 such as one or more image capture devices that can record video and/or still images.

[0048] Still referring to FIG. 5, a universal integrated circuit card (UICC) 940 comprises a subscriber identity module, which in some embodiments includes a secure storage 942 to store secure user information. System 900 may further

include a security processor 950 that may couple to application processor 910. A plurality of sensors 925, including one or more multi-axis accelerometers may couple to application processor 910 to enable input of a variety of sensed information such as motion and other environmental information. In addition, one or more authentication devices 995 may be used to receive, e.g., user biometric input for use in authentication operations.

[0049] As further illustrated, a near field communication (NFC) contactless interface 960 is provided that communicates in a NFC near field via an NFC antenna 965. While separate antennae are shown in FIG. 5, understand that in some implementations one antenna or a different set of antennae may be provided to enable various wireless functionalities.

[0050] A power management integrated circuit (PMIC) 915 couples to application processor 910 to perform platform level power management. To this end, PMIC 915 may issue power management requests to application processor 910 to enter certain low power states as desired. Furthermore, based on platform constraints, PMIC 915 may also control the power level of other components of system 900.

[0051] To enable communications to be transmitted and received such as in one or more IoT networks, various circuitries may be coupled between baseband processor 905 and an antenna 990. Specifically, a radio frequency (RF) transceiver 970 and a wireless local area network (WLAN) transceiver 975 may be present. In general, RF transceiver 970 may be used to receive and transmit wireless data and calls according to a given wireless communication protocol such as 3G or 4G wireless communication protocol such as in accordance with a code division multiple access (CDMA), global system for mobile communication (GSM), long term evolution (LTE) or other protocol. In addition a GPS sensor 980 may be present, with location information being provided to security processor 950 for use as described herein when context information is to be used in a pairing process. Other wireless communications such as receipt or transmission of radio signals, e.g., AM/FM and other signals may also be provided. In addition, via WLAN transceiver 975, local wireless communications, such as according to a Bluetooth™ or IEEE 802.11 standard can also be realized.

[0052] Referring now to FIG. 6, shown is a block diagram of a system in accordance with another embodiment of the present invention. As shown in FIG. 6, multiprocessor system 1000 is a point-to-point interconnect system such as a server system, and includes a first processor 1070 and a second processor 1080 coupled via a point-to-point interconnect 1050. As shown in FIG. 6, each of processors 1070 and 1080 may be multicore processors such as SoCs, including first and second processor cores (i.e., processor cores 1074a and 1074b and processor cores 1084a and 1084b), although potentially many more cores may be present in the processors. In addition, processors 1070 and 1080 each may include a secure engine 1075 and 1085 to perform security operations such as key management, attestations, IoT network onboarding or so forth.

[0053] Still referring to FIG. 6, first processor 1070 further includes a memory controller hub (MCH) 1072 and point-to-point (P-P) interfaces 1076 and 1078. Similarly, second processor 1080 includes a MCH 1082 and P-P interfaces 1086 and 1088. As shown in FIG. 6, MCH's 1072 and 1082 couple the processors to respective memories, namely a memory 1032 and a memory 1034, which may be portions of main memory (e.g., a DRAM) locally attached to the respective processors. First processor 1070 and second processor 1080 may be coupled to a chipset 1090 via P-P interconnects 1052 and 1054, respectively. As shown in FIG. 6, chipset 1090 includes P-P interfaces 1094 and 1098.

[0054] Furthermore, chipset 1090 includes an interface 1092 to couple chipset 1090 with a high performance graphics engine 1038, by a P-P interconnect 1039. In turn, chipset 1090 may be coupled to a first bus 1016 via an interface 1096. As shown in FIG. 6, various input/output (I/O) devices 1014 may be coupled to first bus 1016, along with a bus bridge 1018 which couples first bus 1016 to a second bus 1020. Various devices may be coupled to second bus 1020 including, for example, a keyboard/mouse 1022, communication devices 1026 and a data storage unit 1028 such as a non-volatile storage or other mass storage device. As seen, data storage unit 1028 may include code 1030, in one embodiment. As further seen, data storage unit 1028 also includes a trusted storage 1029 to store sensitive information to be protected. Further, an audio I/O 1024 may be coupled to second bus 1020.

[0055] Embodiments may be used in environments where IoT devices may include wearable devices or other small form factor IoT devices. Referring now to FIG. 7, shown is a block diagram of a wearable module 1300 in accordance with another embodiment. In one particular implementation, module 1300 may be an Intel® Curie™ module that includes multiple components adapted within a single small module that can be implemented as all or part of a wearable device. As seen, module 1300 includes a core 1310 (of course in other embodiments more than one core may be present). Such core may be a relatively low complexity in-order core, such as based on an Intel Architecture® Quark™ design. In some embodiments, core 1310 may implement a TEE as described herein. Core 1310 couples to various components including a sensor hub 1320, which may be configured to interact with a plurality of sensors 1380, such as one or more biometric, motion environmental or other sensors. A power delivery circuit 1330 is present, along with a non-volatile storage 1340. In an embodiment, this circuit may include a rechargeable battery and a recharging circuit, which may in one embodiment receive charging power wirelessly. One or more input/output (IO) interfaces 1350, such as one or more interfaces compatible with one or more of USB/SPI/I2C/GPIO protocols, may be present. In addition, a wireless transceiver 1390, which may be a Bluetooth™ low energy or other short-range wireless transceiver is present to enable wireless communications as described herein. Understand that in different implementations a wearable module can take many other forms. Wearable and/or IoT devices have, in comparison with a typical general purpose CPU or a GPU, a small form factor, low power requirements, limited instruction sets, relatively slow computation throughput, or any of the above.

[0056] Embodiments may be used in many different types of systems. For example, in one embodiment a communication device can be arranged to perform the various methods and techniques described herein. Of course, the scope of the present invention is not limited to a communication device, and instead other embodiments can be directed to other types of apparatus for processing instructions, or one or more machine readable media including instructions that in response to being executed on a computing device, cause the device to carry out one or more of the methods and techniques described herein.

[0057] Embodiments may be implemented in code and may be stored on a non-transitory storage medium having stored thereon instructions which can be used to program a system to perform the instructions. Embodiments also may be implemented in data and may be stored on a non-transitory storage medium, which if used by at least one machine, causes the at least one machine to fabricate at least one integrated circuit to perform one or more operations. The storage medium may include, but is not limited to, any type of disk including floppy disks, optical disks, solid state drives (SSDs), compact disk read-only memories (CD-ROMs), compact disk rewritables (CD-RWs), and magneto-optical disks, semiconductor devices such as read-only memories (ROMs), random access memories (RAMs) such as dynamic random access memories (DRAMs), static random access memories (SRAMs), erasable programmable read-only memories (EPROMs), flash memories, electrically erasable programmable read-only memories (EEPROMs), magnetic or optical cards, or any other type of media suitable for storing electronic instructions.

[0058] The following examples pertain to further embodiments.

[0059] Example 1 includes at least one computer readable storage medium comprising instructions that when executed enable a system to: receive, in a first device, at least one of a first symmetric key and a first asymmetric key in a common key management structure, the common key management structure to accommodate asymmetric keys and symmetric keys, and further including security policy information to enable communication between the first device of a first domain of an Internet of Things (IoT) network and a second device of a second domain of the IoT network according to an inter-domain security policy; and send a first message directly from the first device to the second device according to the security policy information of the common key management structure.

[0060] In example 2 the subject matter of the Example 1 can optionally include instructions that when executed enable the system to dynamically scale the IoT network by dynamically instantiating at least one zone controller for a first zone of the IoT network, the first zone including the first device.

[0061] In example 3 the subject matter of the Examples 1-2 can optionally include instructions that when executed enable the system to dynamically scale the IoT network by dynamically instantiating at least one domain controller for the first domain, which includes the first device.

[0062] In example 4 the subject matter of the Examples 1-3 can optionally include wherein the at least one domain controller comprises a key management system.

[0063] In example 5 the subject matter of the Examples 1-4 can optionally include further comprising instructions that when executed enable the system to dynamically instantiate the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain.

[0064] In example 6 the subject matter of the Examples 1-5 can optionally include further comprising instructions that when executed enable the first device to dynamically instantiate a third device as at least one of a zone controller of the first domain and a client of the first domain.

[0065] In example 7 the subject matter of the Examples 1-6 can optionally include further comprising instructions that when executed enable the system to, in response to the first device receiving at an additional instance of the common key management structure, dynamically instantiate the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain.

[0066] In example 8 the subject matter of the Examples 1-7 can optionally include instructions that when executed enable the system to: for a first publication-subscription topic, dynamically instantiate the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain; and for a second publication-subscription topic, dynamically instantiate the first device as at least one of a domain controller for at least one of the first domain and an additional domain, a zone controller for at least one of the first zone and an additional zone of the additional domain, and a client for at least one of the first domain and the additional domain.

[0067] In example 9 the subject matter of the Examples 1-8 can optionally include instructions that when executed enable the system to: receive a second message in the first device directly from the second device; and enforce, in the first device, the inter-domain security policy with regard to the second message.

[0068] In example 10 the subject matter of the Examples 1-9 can optionally include instructions that when executed enable the system to: receive a third message in the first device directly from a third device; and enforce, in the first device, an intra-domain security policy with regard to the third message; wherein (a) the intra-domain and inter-domain policies are not equal to each other; and (b) the first and third devices, but not the second device, are included in a first domain.

[0069] In example 11 the subject matter of the Examples 1-10 can optionally include wherein the inter-domain policy corresponds to at least one of data filtering, malware screening, and bandwidth reservations.

[0070] In example 12 the subject matter of the Examples 1-11 can optionally include instructions that when executed enable the system to receive the common key management structure from a first zone controller, the first zone controller dynamically assigned a key management role from an initial key management system of the IoT network.

[0071] In example 13 the subject matter of the Examples 1-12 can optionally include wherein the first zone controller and the first device are both included in the first domain, which does not include the second device.

[0072] In example 14 the subject matter of the Examples 1-13 can optionally include instructions that when executed enable the system to: encrypt, in the first device, the first message using the first symmetric key; and send, directly from the first device to the second device, an additional instance of the first symmetric key; wherein (a) the additional instance of the first symmetric key is encrypted based on keys shared between the second device and a third device but not the first device; and (b) the second and third devices are each included in the second domain, which does not include the first device.

[0073] In example 15 the subject matter of the Examples 1-14 can optionally include wherein (a) the first domain is a logical collection of devices, including the first device, under a common administrative control administered by at least one of the collection of devices, and (b) a zone is a logical subset of the devices under a common administrative control administered by at least one of the subset of the devices.

[0074] In example 16 the subject matter of the Examples 1-15 can optionally include instructions that when executed enable the system to receive the common key management structure from a key management system responsive to a request from the first device to interact with the second device.

[0075] In example 17 the subject matter of the Example 1-16 can optionally include instructions that when executed enable the system to cause the common key management structure to expire according to expiration information of the common key management structure, and thereafter prevent interaction between the first device and the second device.

[0076] In example 18 the subject matter of the Examples 1-17 can optionally include instructions that when executed enable the system to: receive, in the first device, the first symmetric key in the common key management structure included in a first communication; and receive, in the first device, the first asymmetric key in the common key management structure included in a second communication.

[0077] In example 19 the subject matter of the Examples 1-18 can optionally include wherein the first symmetric key corresponds to a first encryption protocol and the first asymmetric key corresponds to a second encryption protocol unequal to the first encryption protocol.

[0078] In example 20 the subject matter of the Examples 1-19 can optionally include instructions that when executed enable the system to receive, in the first device, the first symmetric key and the first asymmetric key in the common key management structure.

[0079] In example 21 the subject matter of the Examples 1-20 can optionally include instructions that when executed enable the system to: receive, in the first

device, the first asymmetric key in the common key management structure; and encrypt, in the first device, the first message using the first asymmetric key.

[0080] In example 22 the subject matter of the Examples 1-21 can optionally include instructions that when executed enable the system to send the first message in a data portion of a second common key management structure.

[0081] Example 23 includes a first device comprising: at least one hardware processor to execute instructions; at least one network interface to enable the at least one hardware processor to communicate with second and third computing nodes coupled in an Internet of Things (IoT); at least one non-transitory storage medium having instructions stored thereon for causing the at least one hardware processor of the first device to: receive at least one of a first symmetric key and a first asymmetric key in a common key management structure, the common key management structure to accommodate asymmetric keys and symmetric keys, wherein the common key management structure includes security policy information to enable communication between the first device of a first domain of an Internet of Things (IoT) network and a second device of a second domain of the IoT network according to an inter-domain security policy; and send a first message directly to the second device according to the security policy information of the common key management structure.

[0082] In example 24 the subject matter of the Example 23 can optionally include The first device of claim 23, the at least one medium further comprising instructions that when executed enable the first device to dynamically instantiate at least one zone controller for a first zone of the IoT network, the first zone including the first device.

[0083] While the present invention has been described with respect to a limited number of embodiments, those skilled in the art will appreciate numerous modifications and variations therefrom. It is intended that the appended claims cover all such modifications and variations as fall within the true spirit and scope of this present invention.

What is claimed is:

1. A method comprising:

receiving, in a first device, at least one of a first symmetric key and a first asymmetric key in a common key management structure, the common key management structure to accommodate asymmetric keys and symmetric keys, and further including security policy information to enable communication between the first device of a first domain of an Internet of Things (IoT) network and a second device of a second domain of the IoT network according to an inter-domain security policy; and

sending a first message directly from the first device to the second device according to the security policy information of the common key management structure.

2. The method of claim 1, comprising dynamically scaling the IoT network by dynamically instantiating at least one zone controller for a first zone of the IoT network, the first zone including the first device.

3. The method of claim 1, comprising dynamically scaling the IoT network by dynamically instantiating at least one domain controller for the first domain, which includes the first device.

4. The method of claim 3, wherein the at least one domain controller comprises a key management system.

5. The method of claim 1, further comprising dynamically instantiating the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain.

6. The method of claim 1, further comprising the first device dynamically instantiating a third device as at least one of a zone controller of the first domain and a client of the first domain.

7. The method of claim 1, further comprising in response to the first device receiving at an additional instance of the common key management structure, dynamically instantiating the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain.

8. The method of claim 1, further comprising:

for a first publication-subscription topic, dynamically instantiating the first device as at least one of a domain controller for the first domain, a zone controller for a first zone of the first domain, and a client of the first domain; and

for a second publication-subscription topic, dynamically instantiating the first device as at least one of a domain controller for at least one of the first domain and an additional domain, a zone controller for at least one of the first zone and an additional zone of the additional domain, and a client for at least one of the first domain and the additional domain.

9. The method of claim 1, further comprising:

receiving a second message in the first device directly from the second device; and

enforcing, in the first device, the inter-domain security policy with regard to the second message.

10. The method of claim 9, further comprising:

receiving a third message in the first device directly from a third device; and

enforcing, in the first device, an intra-domain security policy with regard to the third message;

wherein (a) the intra-domain and inter-domain policies are not equal to each other; and (b) the first and third devices, but not the second device, are included in a first domain.

11. The method of claim 10, wherein the inter-domain policy corresponds to at least one of data filtering, malware screening, and bandwidth reservations.

12. The method of claim 1, further comprising receiving the common key management structure from a first zone controller, the first zone controller dynamically assigned a key management role from an initial key management system of the IoT network.

13. The method of claim 12, wherein the first zone controller and the first device are both included in the first domain, which does not include the second device.

14. The method of claim 1, further comprising :

encrypting, in the first device, the first message using the first symmetric key; and

sending, directly from the first device to the second device, an additional instance of the first symmetric key;

wherein (a) the additional instance of the first symmetric key is encrypted based on keys shared between the second device and a third device but not the first device; and (b) the second and third devices are each included in the second domain, which does not include the first device.

15. The method of claim 14, wherein (a) the first domain is a logical collection of devices, including the first device, under a common administrative control administered by at least one of the collection of devices, and (b) a zone is a logical subset of the devices under a common administrative control administered by at least one of the subset of the devices.

16. The method of claim 1, further comprising receiving the common key management structure from a key management system responsive to a request from the first device to interact with the second device.

17. The method of claim 1, further comprising causing the common key management structure to expire according to expiration information of the common

key management structure, and thereafter prevent interaction between the first device and the second device.

18. The method of claim 1, further comprising:

receiving, in the first device, the first symmetric key in the common key management structure included in a first communication; and

receiving, in the first device, the first asymmetric key in the common key management structure included in a second communication.

19. The method of claim 18, wherein the first symmetric key corresponds to a first encryption protocol and the first asymmetric key corresponds to a second encryption protocol unequal to the first encryption protocol.

20. The method of claim 1, further comprising receiving, in the first device, the first symmetric key and the first asymmetric key in the common key management structure.

21. The method of claim 1, further comprising :

receiving, in the first device, the first asymmetric key in the common key management structure; and.

encrypting, in the first device, the first message using the first asymmetric key.

22. The method of claim 1, further comprising sending the first message in a data portion of a second common key management structure.

23. At least one machine readable medium comprising a plurality of instructions that in response to being executed on a computing device, cause the computing device to carry out a method according to any one of claims 1 to 22.

24. A first device comprising:

at least one hardware processor to execute instructions;

at least one network interface to enable the at least one hardware processor to communicate with second and third computing nodes coupled in an Internet of Things (IoT);

at least one non-transitory storage medium having instructions stored thereon for causing the at least one hardware processor of the first device to:

receive at least one of a first symmetric key and a first asymmetric key in a common key management structure, the common key management structure to accommodate asymmetric keys and symmetric keys, wherein the common key management structure includes security policy information to enable communication between the first device of a first domain of an Internet of Things (IoT) network and a second device of a second domain of the IoT network according to an inter-domain security policy; and

send a first message directly to the second device according to the security policy information of the common key management structure.

25. The first device of claim 24, the at least one medium further comprising instructions that when executed enable the first device to dynamically instantiate at least one zone controller for a first zone of the IoT network, the first zone including the first device.

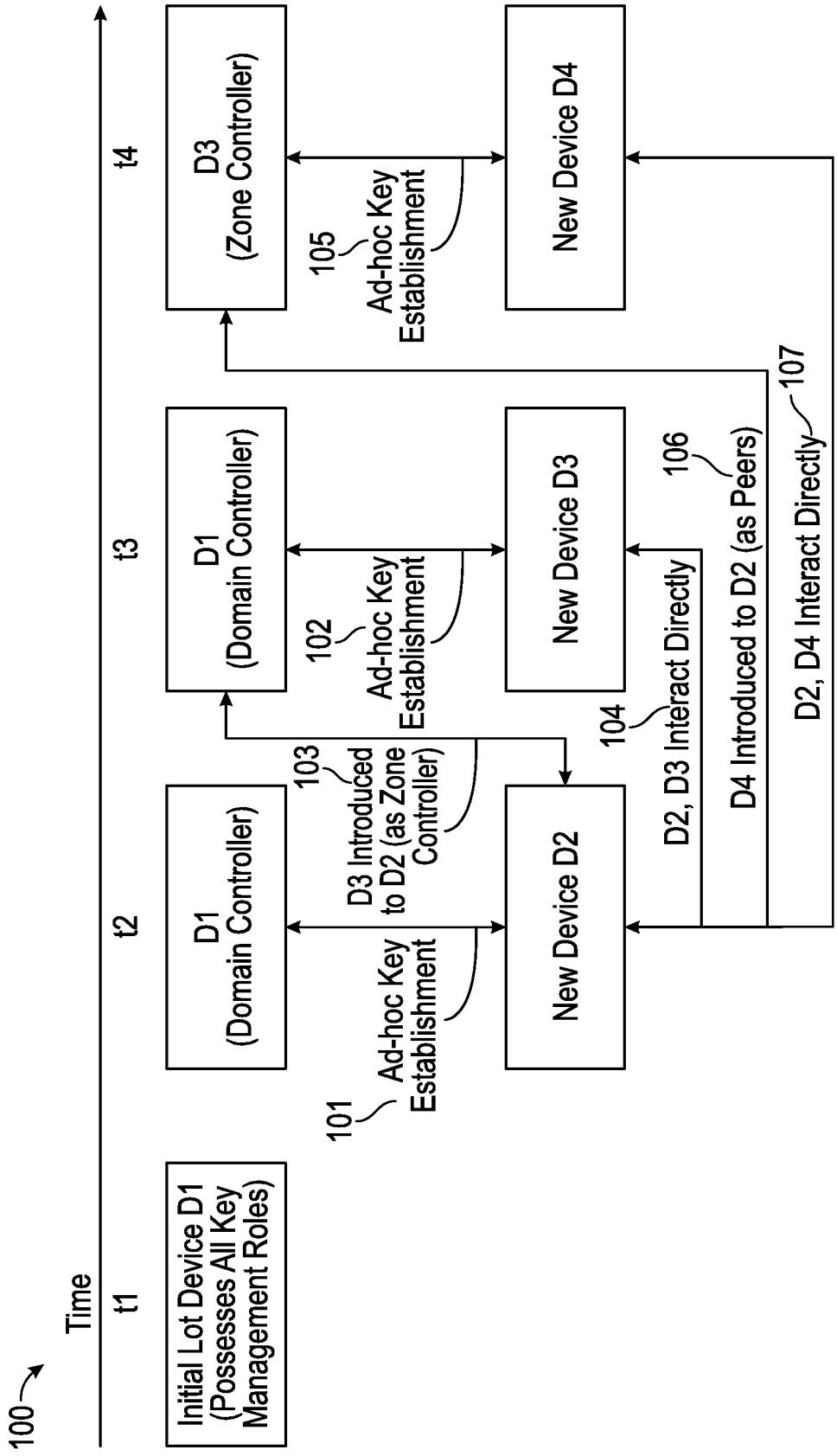

100
Time
t1
t2
t3
t4
Initial Lot Device D1 (Possesses All Key Management Roles)
D1 (Domain Controller)
D1 (Domain Controller)
D3 (Zone Controller)
New Device D2
New Device D3
New Device D4
101
Ad-hoc Key Establishment
103
D3 Introduced to D2 (as Zone Controller)
102
Ad-hoc Key Establishment
105
Ad-hoc Key Establishment
104
D2, D3 Interact Directly
106
D4 Introduced to D2 (as Peers)
107
D2, D4 Interact Directly

FIG. 1

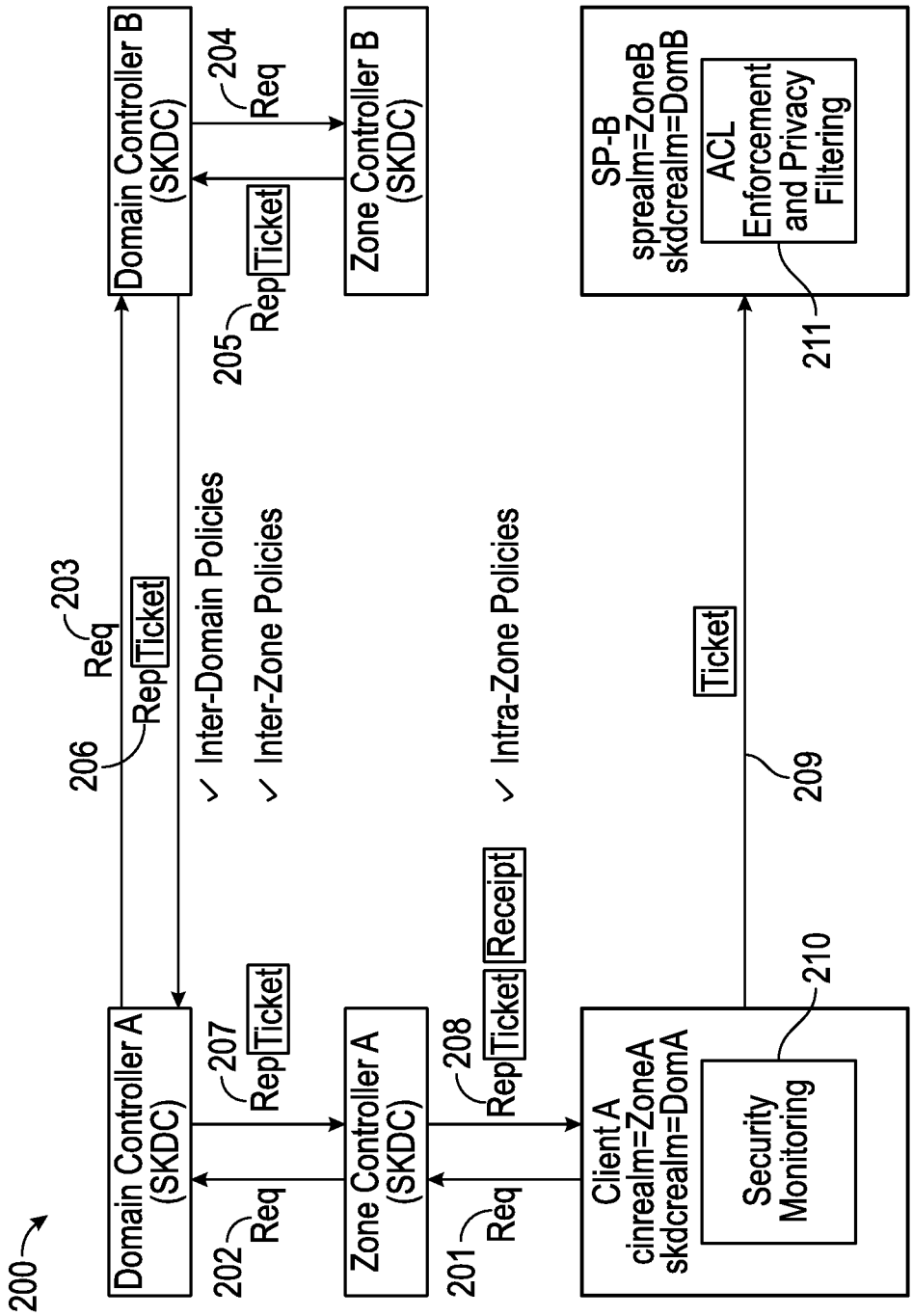

200
Domain Controller A (SKDC)
Zone Controller A (SKDC)
Client A cinrealm=ZoneA skdcrealm=DomA
Security Monitoring
210
Domain Controller B (SKDC)
Zone Controller B (SKDC)
SP-B sprealm=ZoneB skdcrealm=DomB
ACL Enforcement and Privacy Filtering
211
201 Req
202 Req
Rep Ticket
207 Rep Ticket
208 Rep Ticket Receipt
Ticket
209
203 Req
206 Rep Ticket
204 Req
205 Rep Ticket
✓ Inter-Domain Policies
✓ Inter-Zone Policies
✓ Intra-Zone Policies

FIG. 2

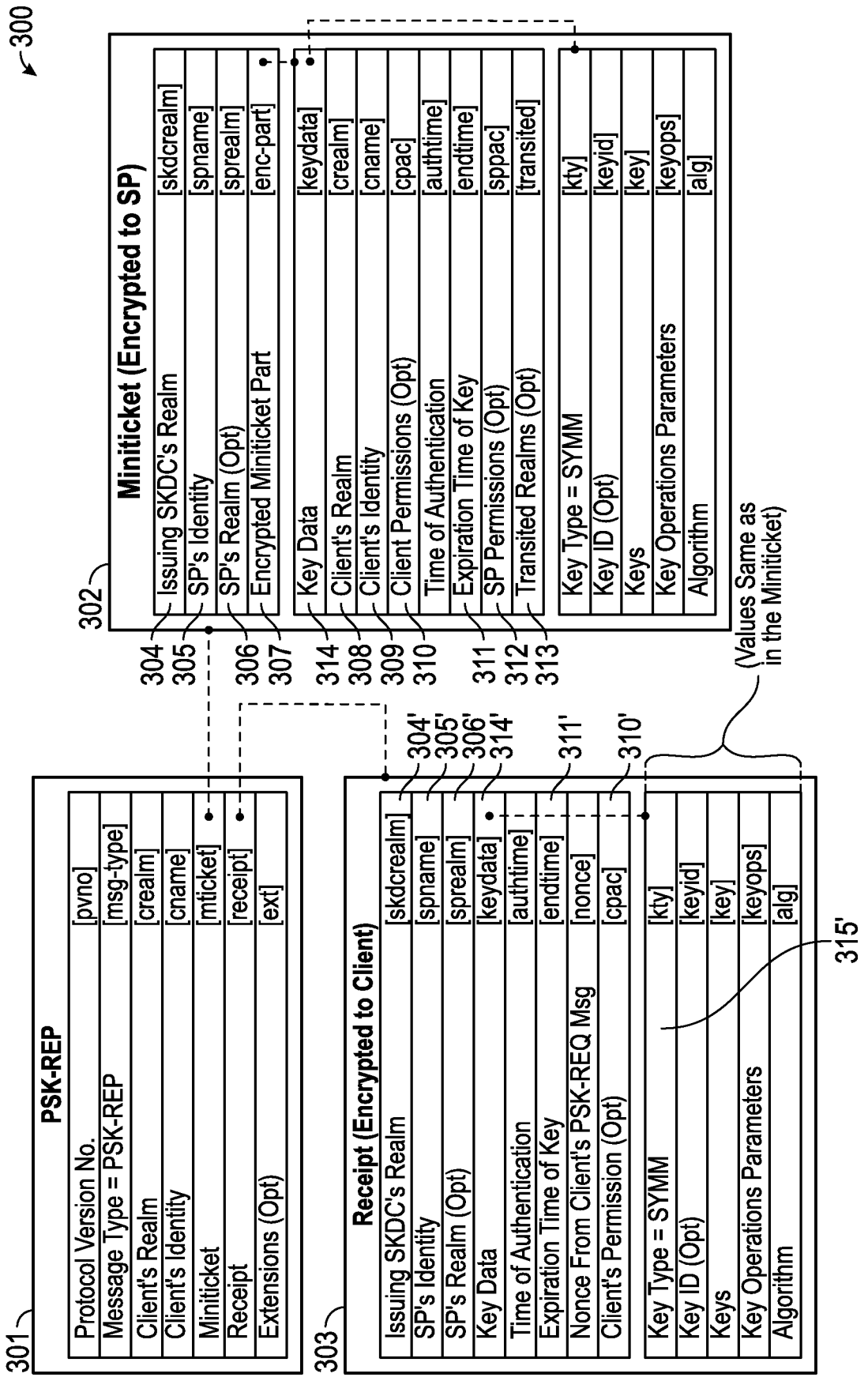

300
Miniticket (Encrypted to SP)
Issuing SKDC's Realm [skdcrealm]
SP's Identity [spname]
SP's Realm (Opt) [sprealm]
Encrypted Miniticket Part [enc-part]
Key Data [keydata]
Client's Realm [crealm]
Client's Identity [cname]
Client Permissions (Opt) [cpac]
Time of Authentication [authtime]
Expiration Time of Key [endtime]
SP Permissions (Opt) [sppac]
Transited Realms (Opt) [transited]
Key Type = SYMM [kty]
Key ID (Opt) [keyid]
Keys [key]
Key Operations Parameters [keyops]
Algorithm [alg]
302
304
305
306
307
314
308
309
310
311
312
313
PSK-REP
Protocol Version No. [pvno]
Message Type = PSK-REP [msg-type]
Client's Realm [crealm]
Client's Identity [cname]
Miniticket [mticket]
Receipt [receipt]
Extensions (Opt) [ext]
301
Receipt (Encrypted to Client)
Issuing SKDC's Realm [skdcrealm]
SP's Identity [spname]
SP's Realm (Opt) [sprealm]
Key Data [keydata]
Time of Authentication [authtime]
Expiration Time of Key [endtime]
Nonce From Client's PSK-REQ Msg [nonce]
Client's Permission (Opt) [cpac]
Key Type = SYMM [kty]
Key ID (Opt) [keyid]
Keys [key]
Key Operations Parameters [keyops]
Algorithm [alg]
303
304'
305'
306'
314'
311'
310'
315'
(Values Same as in the Miniticket)

FIG. 3

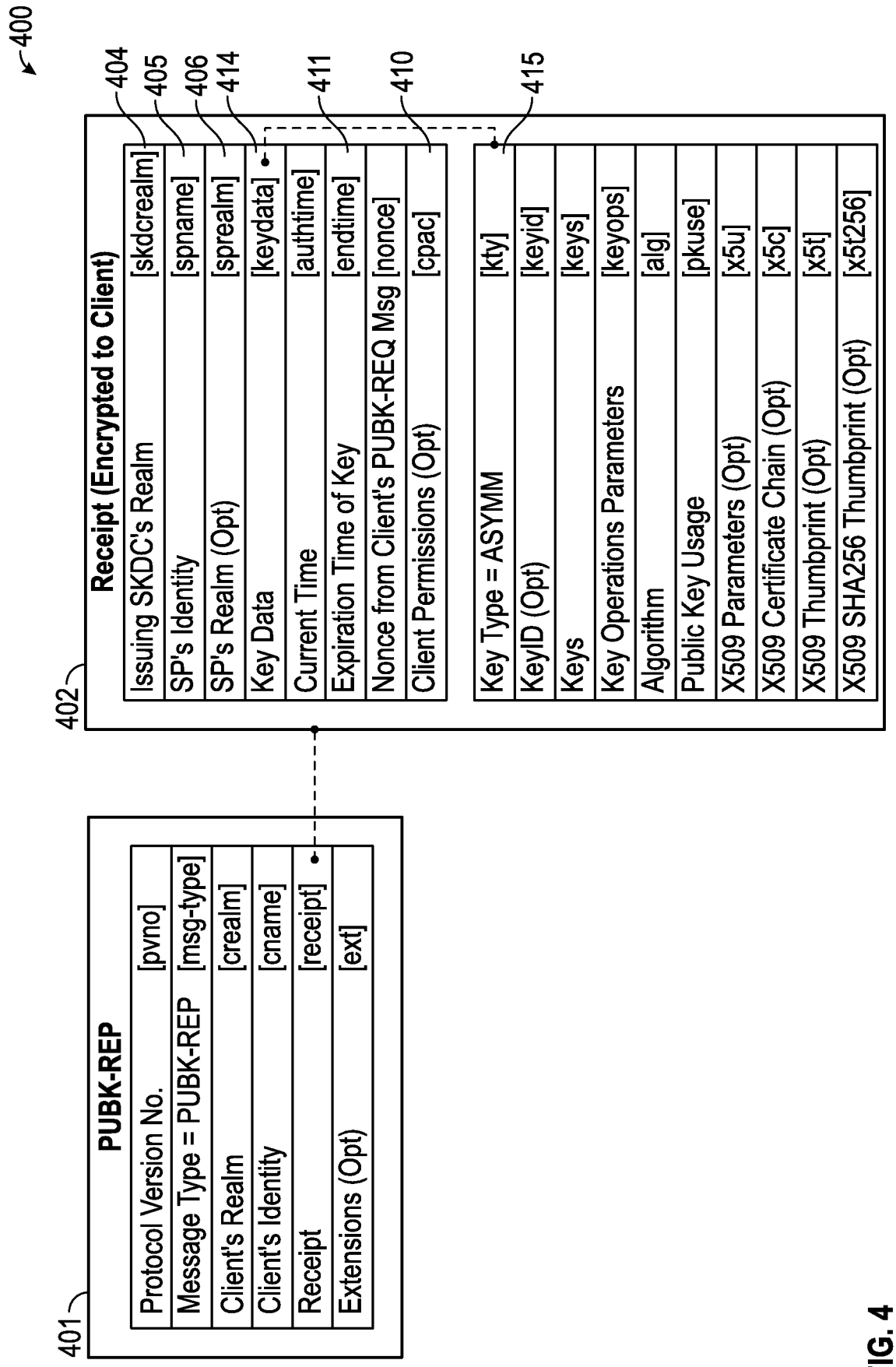

400
401
PUBK-REP
Protocol Version No. [pvno]
Message Type = PUBK-REP [msg-type]
Client's Realm [crealm]
Client's Identity [cname]
Receipt [receipt]
Extensions (Opt) [ext]
402
Receipt (Encrypted to Client)
Issuing SKDC's Realm [skdcrealm]
SP's Identity [spname]
SP's Realm (Opt) [sprealm]
Key Data [keydata]
Current Time [authtime]
Expiration Time of Key [endtime]
Nonce from Client's PUBK-REQ Msg [nonce]
Client Permissions (Opt) [cpac]
Key Type = ASYMM [kty]
KeyID (Opt) [keyid]
Keys [keys]
Key Operations Parameters [keyops]
Algorithm [alg]
Public Key Usage [pkuse]
X509 Parameters (Opt) [x5u]
X509 Certificate Chain (Opt) [x5c]
X509 Thumbprint (Opt) [x5t]
X509 SHA256 Thumbprint (Opt) [x5t256]
404
405
406
414
411
410
415

FIG. 4

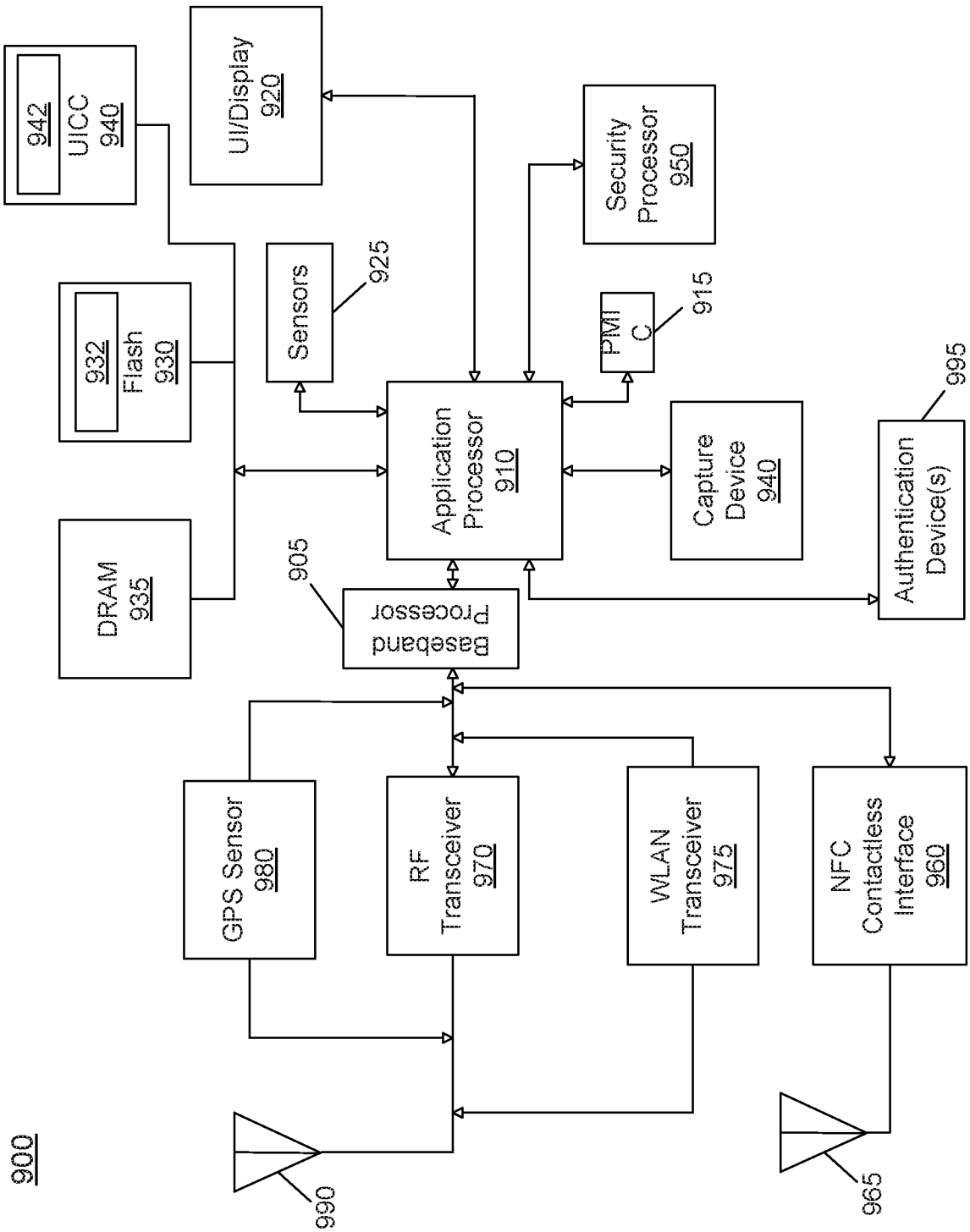

900
UICC 940
942
UI/Display 920
Security Processor 950
Sensors 925
Flash 930
932
PMIC 915
Application Processor 910
Capture Device 940
Authentication Device(s) 995
DRAM 935
905
Baseband Processor
GPS Sensor 980
RF Transceiver 970
WLAN Transceiver 975
NFC Contactless Interface 960
990
965

FIG. 5

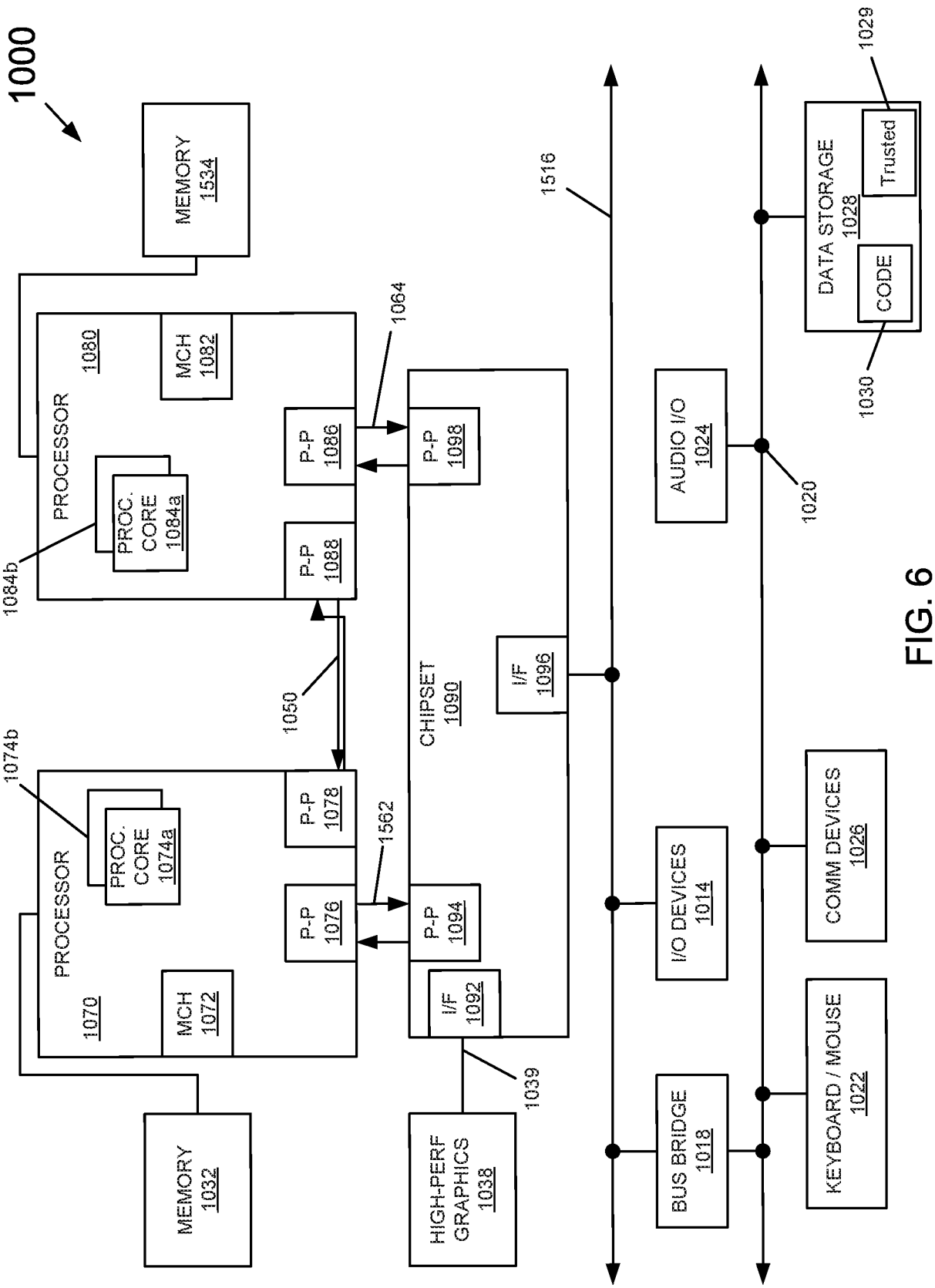

1000
MEMORY 1534
PROCESSOR 1080
MCH 1082
PROC. CORE 1084a
1084b
P-P 1086
P-P 1088
1064
1050
CHIPSET 1090
P-P 1098
I/F 1096
1516
AUDIO I/O 1024
1020
DATA STORAGE 1028
Trusted
CODE
1029
1030
1074b
PROCESSOR 1070
PROC. CORE 1074a
P-P 1078
P-P 1076
1562
P-P 1094
I/F 1092
MCH 1072
I/O DEVICES 1014
COMM DEVICES 1026
KEYBOARD / MOUSE 1022
BUS BRIDGE 1018
MEMORY 1032
1039
HIGH-PERF GRAPHICS 1038

FIG. 6

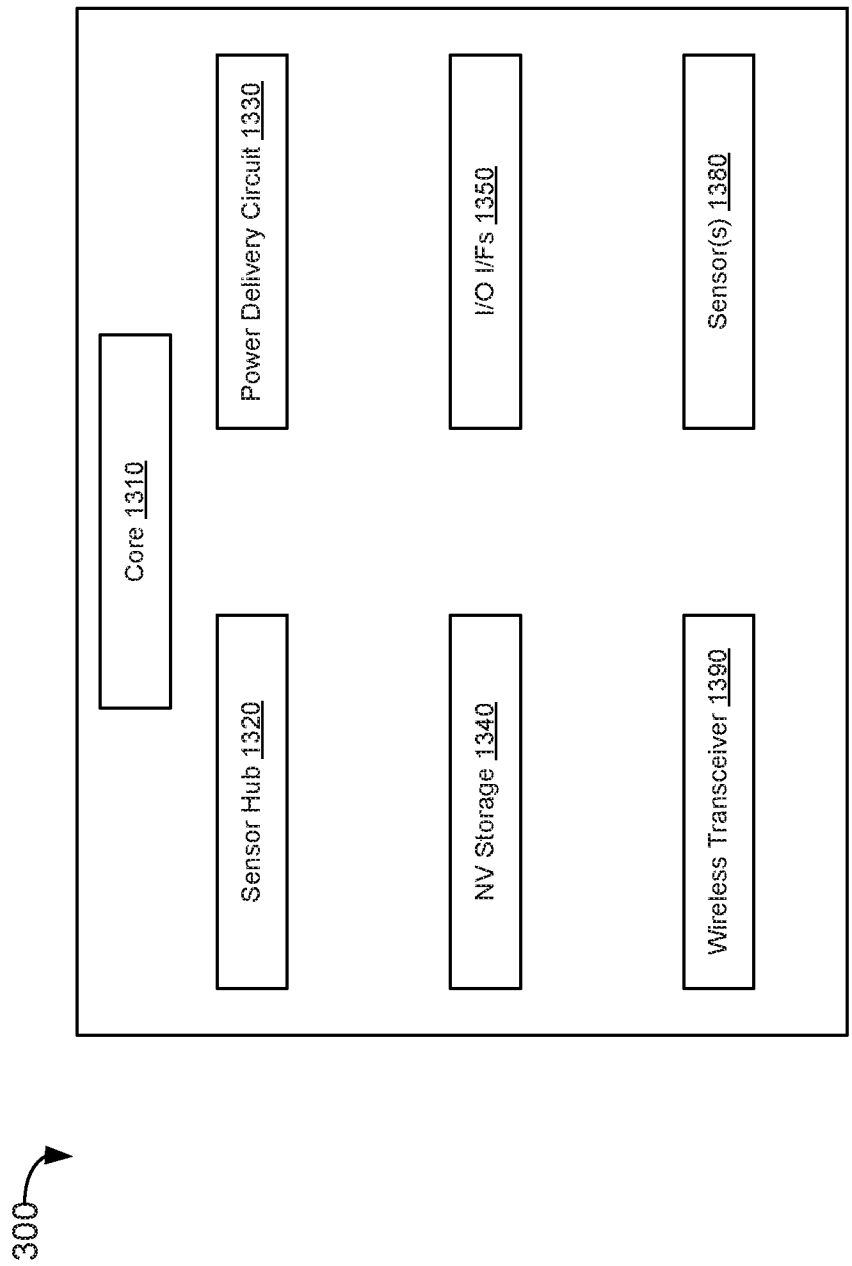

1300
Core 1310
Power Delivery Circuit 1330
I/O I/Fs 1350
Sensor(s) 1380
Sensor Hub 1320
NV Storage 1340
Wireless Transceiver 1390

FIG. 7

**INTERNATIONAL SEARCH REPORT**

International application No.
**PCT/US2016/035282**

**A. CLASSIFICATION OF SUBJECT MATTER**

**H04L 9/08(2006.01)I, H04L 29/08(2006.01)I, H04L 29/06(2006.01)I**

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 9/08; H04Q 7/24; H04L 29/06; H04L 12/56; H04L 29/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: Internet of Things (IoT), key management, encryption, inter-domain, intra-domain, symmetric, asymmetric

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | T. HARDJONO et al., 'Fluffy: Simplified Key Exchange for Constrained Environments', Network Working Group, Internet-Draft, pp. 1-26, 23 March 2015 (https://datatracker.ietf.org/doc/draft-hardjono-ace-fluffy/00/) See pages 3, 4, 6, 14, 15, 18, 20, 21; and figures 3, 4. | 1-8,12-25 |
| Y | | 9-11 |
| Y | US 2013-0170499 A1 (RANGA SRI RAMANUJAN et al.) 04 July 2013 See paragraph [0112]; and claim 51. | 9-11 |
| A | WO 2014-017959 A1 (TELEFONAKTIEBOLAGET L M ERICSSON (PUBL)) 30 January 2014 See page 13, line 1 - page 14, line 5; and figure 3. | 1-25 |
| A | US 2007-0253376 A1 (JEFFREY D. BONTA et al.) 01 November 2007 See paragraphs [0041], [0042]; and figure 4. | 1-25 |
| A | US 8266433 B1 (ROBERT PRZYKUCKI et al.) 11 September 2012 See column 17, line 26 - column 18, line 7; and figure 8. | 1-25 |

☐ 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 |
|---|---|
| 24 August 2016 (24.08.2016) | **24 August 2016 (24.08.2016)** |


Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
LEE, EUN KYU
Telephone No. +82-42-481-3580


Form PCT/ISA/210 (second sheet) (January 2015)

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2016/035282**

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| US 2013-0170499 A1 | 04/07/2013 | US 9225637 B2 | 29/12/2015 |
| WO 2014-017959 A1 | 30/01/2014 | EP 2878112 A1 | 03/06/2015 |
| | | EP 2878112 B1 | 21/10/2015 |
| | | US 2015-0195261 A1 | 09/07/2015 |
| US 2007-0253376 A1 | 01/11/2007 | AU 2007-242991 A1 | 08/11/2007 |
| | | AU 2007-242991 B2 | 03/02/2011 |
| | | BR PI0711079 A2 | 23/08/2011 |
| | | CA 2650050 A1 | 08/11/2007 |
| | | CA 2650050 C | 28/08/2012 |
| | | EP 2022282 A2 | 11/02/2009 |
| | | EP 2022282 A4 | 04/04/2012 |
| | | JP 04911480 B2 | 04/04/2012 |
| | | JP 2009-535921 A | 01/10/2009 |
| | | MX 2008013772 A | 15/04/2009 |
| | | RU 2008146960 A | 10/06/2010 |
| | | RU 2406252 C2 | 10/12/2010 |
| | | US 7817986 B2 | 19/10/2010 |
| | | WO 2007-127637 A2 | 08/11/2007 |
| | | WO 2007-127637 A3 | 11/12/2008 |
| US 8266433 B1 | 11/09/2012 | None | |