

(51) International Patent Classification:
H04L 29/06 (2006.01)(21) International Application Number:
PCT/US2015/040917(22) International Filing Date:
17 July 2015 (17.07.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/026,278 18 July 2014 (18.07.2014) US(71) Applicant: **INTERDIGITAL TECHNOLOGY CORPORATION** [US/US]; 200 Bellevue Parkway, Suite 300, Wilmington, Delaware 19809 (US).(72) Inventors: **ROY, Michael**; 22 de Chantilly, Candiac, Québec J5R 6C5 (CA). **JOLICOEUR, Martin**; 130 Madrid, Dollard-des-Ormeaux, Québec H9B 3M1 (CA). **DILALLO, Kevin**; 5995 rue Tousignan, Laval, Québec H7K 0B8 (CA).(74) Agent: **WALSH, Christina R.**; Volpe and Koenig, P.C., 30 S. 17th Street, United Plaza, Philadelphia, Pennsylvania 19103 (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).

Published:

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

(54) Title: ADVANCED FLOW ROUTING: A METHOD TO AUGMENT LEGACY ROUTING CAPABILITIES

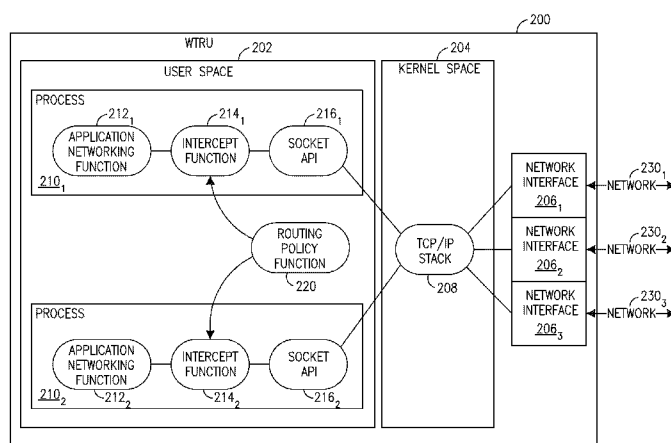


FIG. 2

(57) **Abstract:** Methods and apparatuses are configured for advanced flow routing. A communication device may be configured with a plurality of network interfaces each configured to access a respective network. An application may make a socket call for an IP flow to a socket application programming interface (API). An intercept function may intercept the socket call and may select a network interface for the IP flow in accordance with policy-based routing rules and based on at least one of: IP flow characteristics and routing policy information, where the IP flow characteristics may include at least one of: an application name and a destination domain name. If the IP flow matches at least one of the policy-based routing rules, the intercept function may invoke the socket call and the Transmission Control Protocol/Internet Protocol (TCP/IP) stack may route traffic for the IP flow to the network interface selected by the intercept function.

ADVANCED FLOW ROUTING: A METHOD TO AUGMENT LEGACY ROUTING CAPABILITIES

CROSS REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Application No. 62/026,278 filed July 18, 2014, which is incorporated by reference as if fully set forth.

BACKGROUND

[0002] Modern personal communication computing devices, such as cell phones, tablets, and laptops, allow users to connect to different networks simultaneously. IP traffic routing decisions therefore are typically made by the communication device to allow a user to benefit from simultaneous connections to different networks. In order to help communication devices make better decisions when connecting to different networks, policy-based routing techniques have been developed, which include techniques used to make routing decisions based on policies set by a network administrator. For example, the Inter-System Routing Policy (ISRP) defined in the 3rd Generation Partnership Project (3GPP) Access Network Discovery and Selection Function (ANDSF) specification is an example of such a policy-based routing approach.

[0003] Support of all IP routing features of a specification such as ANDSF may not be possible using traditional Transmission Control Protocol/Internet Protocol (TCP/IP) stack routing. While some of the ISRP features may be implemented using the legacy TCP/IP stack, there are other ISRP features that may not be incompatible with the TCP/IP stack. Thus, there is a need for an advanced IP flow routing techniques that enhance IP routing capabilities of communication systems that use policy-based routing techniques.

SUMMARY

[0004] Methods and apparatuses are configured for advanced flow routing. A communication device may be configured with a plurality of network interfaces each configured to access a respective network. An application may initiate an Internet Protocol (IP) network session for an IP flow by making a socket call to a socket application programming interface (API). An intercept function may intercept the socket call and may perform routing analysis to select a network interface from the plurality of network interfaces for the IP flow in accordance with policy-based routing rules and based on at least one of: IP flow characteristics and routing policy information. The IP flow characteristics may include at least one of: an application name and a destination domain name. If the IP flow matches at least one of the policy-based routing rules and the intercept function doesn't block the IP flow, the intercept function may invoke the socket call and route traffic for the IP flow to the network interface selected by the intercept function via a Transmission Control Protocol/Internet Protocol (TCP/IP) stack.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] A more detailed understanding may be had from the following description, given by way of example in conjunction with the accompanying drawings wherein:

[0006] FIG. 1A is a system diagram of an example communications system in which one or more disclosed embodiments may be implemented;

[0007] FIG. 1B is a system diagram of an example wireless transmit/receive unit (WTRU) that may be used within the communications system illustrated in FIG. 1A;

[0008] FIG. 1C is a system diagram of an example radio access network and an example core network that may be used within the communications system illustrated in FIG. 1A;

[0009] FIG. 2 shows a block diagram of an example wireless transmit/receive unit (WTRU) including Intercept Functions in the User Space;

[0010] FIG. 3 shows a flow diagram of an example flow routing procedure employing an Intercept Function in the User Space;

[0011] FIG. 4 shows a block diagram of an example WTRU including an Intercept Functions in the Kernel Space; and

[0012] FIG. 5 shows a flow diagram of an example flow routing procedure employing an Intercept Function in the Kernel Space.

DETAILED DESCRIPTION

[0013] FIG. 1A is a diagram of an example communications system 100 in which one or more disclosed embodiments may be implemented. The communications system 100 may be a multiple access system that provides content, such as voice, data, video, messaging, broadcast, etc., to multiple wireless users. The communications system 100 may enable multiple wireless users to access such content through the sharing of system resources, including wireless bandwidth. For example, the communications systems 100 may employ one or more channel access methods, such as code division multiple access (CDMA), time division multiple access (TDMA), frequency division multiple access (FDMA), orthogonal FDMA (OFDMA), single-carrier FDMA (SC-FDMA), and the like.

[0014] As shown in FIG. 1A, the communications system 100 may include wireless transmit/receive units (WTRUs) 102a, 102b, 102c, 102d, a radio access network (RAN) 104, a core network 106, a public switched telephone network (PSTN) 108, the Internet 110, and other networks 112, though it will be appreciated that the disclosed embodiments contemplate any number of WTRUs, base stations, networks, and/or network elements. Each of the WTRUs 102a, 102b, 102c, 102d may be any type of device configured to operate and/or communicate in a wireless environment. By way of example, the WTRUs 102a, 102b, 102c, 102d may be configured to transmit and/or

receive wireless signals and may include user equipment (UE), a mobile station, a fixed or mobile subscriber unit, a pager, a cellular telephone, a personal digital assistant (PDA), a smartphone, a laptop, a netbook, a personal computer, a wireless sensor, consumer electronics, and the like.

[0015] The communications systems 100 may also include a base station 114a and a base station 114b. Each of the base stations 114a, 114b may be any type of device configured to wirelessly interface with at least one of the WTRUs 102a, 102b, 102c, 102d to facilitate access to one or more communication networks, such as the core network 106, the Internet 110, and/or the other networks 112. By way of example, the base stations 114a, 114b may be a base transceiver station (BTS), a Node-B, an eNode B, a Home Node B, a Home eNode B, a site controller, an access point (AP), a wireless router, and the like. While the base stations 114a, 114b are each depicted as a single element, it will be appreciated that the base stations 114a, 114b may include any number of interconnected base stations and/or network elements.

[0016] The base station 114a may be part of the RAN 104, which may also include other base stations and/or network elements (not shown), such as a base station controller (BSC), a radio network controller (RNC), relay nodes, etc. The base station 114a and/or the base station 114b may be configured to transmit and/or receive wireless signals within a particular geographic region, which may be referred to as a cell (not shown). The cell may further be divided into cell sectors. For example, the cell associated with the base station 114a may be divided into three sectors. Thus, in one embodiment, the base station 114a may include three transceivers, i.e., one for each sector of the cell. In another embodiment, the base station 114a may employ multiple-input multiple-output (MIMO) technology and, therefore, may utilize multiple transceivers for each sector of the cell.

[0017] The base stations 114a, 114b may communicate with one or more of the WTRUs 102a, 102b, 102c, 102d over an air interface 116, which may be any suitable wireless communication link (e.g., radio frequency (RF),

microwave, infrared (IR), ultraviolet (UV), visible light, etc.). The air interface 116 may be established using any suitable radio access technology (RAT).

[0018] More specifically, as noted above, the communications system 100 may be a multiple access system and may employ one or more channel access schemes, such as CDMA, TDMA, FDMA, OFDMA, SC-FDMA, and the like. For example, the base station 114a in the RAN 104 and the WTRUs 102a, 102b, 102c may implement a radio technology such as Universal Mobile Telecommunications System (UMTS) Terrestrial Radio Access (UTRA), which may establish the air interface 116 using wideband CDMA (WCDMA). WCDMA may include communication protocols such as High-Speed Packet Access (HSPA) and/or Evolved HSPA (HSPA+). HSPA may include High-Speed Downlink Packet Access (HSDPA) and/or High-Speed Uplink Packet Access (HSUPA).

[0019] In another embodiment, the base station 114a and the WTRUs 102a, 102b, 102c may implement a radio technology such as Evolved UMTS Terrestrial Radio Access (E-UTRA), which may establish the air interface 116 using Long Term Evolution (LTE) and/or LTE-Advanced (LTE-A).

[0020] In other embodiments, the base station 114a and the WTRUs 102a, 102b, 102c may implement radio technologies such as IEEE 802.16 (i.e., Worldwide Interoperability for Microwave Access (WiMAX)), CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, Interim Standard 2000 (IS-2000), Interim Standard 95 (IS-95), Interim Standard 856 (IS-856), Global System for Mobile communications (GSM), Enhanced Data rates for GSM Evolution (EDGE), GSM EDGE (GERAN), and the like.

[0021] The base station 114b in FIG. 1A may be a wireless router, Home Node B, Home eNode B, or access point, for example, and may utilize any suitable RAT for facilitating wireless connectivity in a localized area, such as a place of business, a home, a vehicle, a campus, and the like. In one embodiment, the base station 114b and the WTRUs 102c, 102d may implement a radio technology such as IEEE 802.11 to establish a wireless local area network (WLAN). In another embodiment, the base station 114b

and the WTRUs 102c, 102d may implement a radio technology such as IEEE 802.15 to establish a wireless personal area network (WPAN). In yet another embodiment, the base station 114b and the WTRUs 102c, 102d may utilize a cellular-based RAT (e.g., WCDMA, CDMA2000, GSM, LTE, LTE-A, etc.) to establish a picocell or femtocell. As shown in FIG. 1A, the base station 114b may have a direct connection to the Internet 110. Thus, the base station 114b may not be required to access the Internet 110 via the core network 106.

[0022] The RAN 104 may be in communication with the core network 106, which may be any type of network configured to provide voice, data, applications, and/or voice over internet protocol (VoIP) services to one or more of the WTRUs 102a, 102b, 102c, 102d. For example, the core network 106 may provide call control, billing services, mobile location-based services, pre-paid calling, Internet connectivity, video distribution, etc., and/or perform high-level security functions, such as user authentication. Although not shown in FIG. 1A, it will be appreciated that the RAN 104 and/or the core network 106 may be in direct or indirect communication with other RANs that employ the same RAT as the RAN 104 or a different RAT. For example, in addition to being connected to the RAN 104, which may be utilizing an E-UTRA radio technology, the core network 106 may also be in communication with another RAN (not shown) employing a GSM radio technology.

[0023] The core network 106 may also serve as a gateway for the WTRUs 102a, 102b, 102c, 102d to access the PSTN 108, the Internet 110, and/or other networks 112. The PSTN 108 may include circuit-switched telephone networks that provide plain old telephone service (POTS). The Internet 110 may include a global system of interconnected computer networks and devices that use common communication protocols, such as the transmission control protocol (TCP), user datagram protocol (UDP) and the internet protocol (IP) in the TCP/IP internet protocol suite. The networks 112 may include wired or wireless communications networks owned and/or operated by other service providers. For example, the networks 112 may

include another core network connected to one or more RANs, which may employ the same RAT as the RAN 104 or a different RAT.

[0024] Some or all of the WTRUs 102a, 102b, 102c, 102d in the communications system 100 may include multi-mode capabilities, i.e., the WTRUs 102a, 102b, 102c, 102d may include multiple transceivers for communicating with different wireless networks over different wireless links. For example, the WTRU 102c shown in FIG. 1A may be configured to communicate with the base station 114a, which may employ a cellular-based radio technology, and with the base station 114b, which may employ an IEEE 802 radio technology.

[0025] FIG. 1B is a system diagram of an example WTRU 102. As shown in FIG. 1B, the WTRU 102 may include a processor 118, a transceiver 120, a transmit/receive element 122, a speaker/microphone 124, a keypad 126, a display/touchpad 128, non-removable memory 130, removable memory 132, a power source 134, a global positioning system (GPS) chipset 136, and other peripherals 138. It will be appreciated that the WTRU 102 may include any sub-combination of the foregoing elements while remaining consistent with an embodiment.

[0026] The processor 118 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 118 may perform signal coding, data processing, power control, input/output processing, and/or any other functionality that enables the WTRU 102 to operate in a wireless environment. The processor 118 may be coupled to the transceiver 120, which may be coupled to the transmit/receive element 122. While FIG. 1B depicts the processor 118 and the transceiver 120 as separate components, it will be appreciated that the processor 118 and the transceiver 120 may be integrated together in an electronic package or chip.

[0027] The transmit/receive element 122 may be configured to transmit signals to, or receive signals from, a base station (e.g., the base station 114a) over the air interface 116. For example, in one embodiment, the transmit/receive element 122 may be an antenna configured to transmit and/or receive RF signals. In another embodiment, the transmit/receive element 122 may be an emitter/detector configured to transmit and/or receive IR, UV, or visible light signals, for example. In yet another embodiment, the transmit/receive element 122 may be configured to transmit and receive both RF and light signals. It will be appreciated that the transmit/receive element 122 may be configured to transmit and/or receive any combination of wireless signals.

[0028] In addition, although the transmit/receive element 122 is depicted in FIG. 1B as a single element, the WTRU 102 may include any number of transmit/receive elements 122. More specifically, the WTRU 102 may employ MIMO technology. Thus, in one embodiment, the WTRU 102 may include two or more transmit/receive elements 122 (e.g., multiple antennas) for transmitting and receiving wireless signals over the air interface 116.

[0029] The transceiver 120 may be configured to modulate the signals that are to be transmitted by the transmit/receive element 122 and to demodulate the signals that are received by the transmit/receive element 122. As noted above, the WTRU 102 may have multi-mode capabilities. Thus, the transceiver 120 may include multiple transceivers for enabling the WTRU 102 to communicate via multiple RATs, such as UTRA and IEEE 802.11, for example.

[0030] The processor 118 of the WTRU 102 may be coupled to, and may receive user input data from, the speaker/microphone 124, the keypad 126, and/or the display/touchpad 128 (e.g., a liquid crystal display (LCD) display unit or organic light-emitting diode (OLED) display unit). The processor 118 may also output user data to the speaker/microphone 124, the keypad 126, and/or the display/touchpad 128. In addition, the processor 118 may access

information from, and store data in, any type of suitable memory, such as the non-removable memory 130 and/or the removable memory 132. The non-removable memory 130 may include random-access memory (RAM), read-only memory (ROM), a hard disk, or any other type of memory storage device. The removable memory 132 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 118 may access information from, and store data in, memory that is not physically located on the WTRU 102, such as on a server or a home computer (not shown).

[0031] The processor 118 may receive power from the power source 134, and may be configured to distribute and/or control the power to the other components in the WTRU 102. The power source 134 may be any suitable device for powering the WTRU 102. For example, the power source 134 may include one or more dry cell batteries (e.g., nickel-cadmium (NiCd), nickel-zinc (NiZn), nickel metal hydride (NiMH), lithium-ion (Li-ion), etc.), solar cells, fuel cells, and the like.

[0032] The processor 118 may also be coupled to the GPS chipset 136, which may be configured to provide location information (e.g., longitude and latitude) regarding the current location of the WTRU 102. In addition to, or in lieu of, the information from the GPS chipset 136, the WTRU 102 may receive location information over the air interface 116 from a base station (e.g., base stations 114a, 114b) 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 102 may acquire location information by way of any suitable location-determination method while remaining consistent with an embodiment.

[0033] The processor 118 may further be coupled to other peripherals 138, 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 138 may include 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.

[0034] FIG. 1C is a system diagram of the RAN 104 and the core network 106 according to an embodiment. As noted above, the RAN 104 may employ an E-UTRA radio technology to communicate with the WTRUs 102a, 102b, 102c over the air interface 116. The RAN 104 may also be in communication with the core network 106.

[0035] The RAN 104 may include eNode-Bs 140a, 140b, 140c, though it will be appreciated that the RAN 104 may include any number of eNode-Bs while remaining consistent with an embodiment. The eNode-Bs 140a, 140b, 140c may each include one or more transceivers for communicating with the WTRUs 102a, 102b, 102c over the air interface 116. In one embodiment, the eNode-Bs 140a, 140b, 140c may implement MIMO technology. Thus, the eNode-B 140a, for example, may use multiple antennas to transmit wireless signals to, and receive wireless signals from, the WTRU 102a.

[0036] Each of the eNode-Bs 140a, 140b, 140c may be associated with a particular cell (not shown) and may be configured to handle radio resource management decisions, handover decisions, scheduling of users in the uplink and/or downlink, and the like. As shown in FIG. 1C, the eNode-Bs 140a, 140b, 140c may communicate with one another over an X2 interface.

[0037] The core network 106 shown in FIG. 1C may include a mobility management gateway (MME) 142, a serving gateway 144, and a packet data network (PDN) gateway 146. While each of the foregoing elements are depicted as part of the core network 106, it will be appreciated that any one of these elements may be owned and/or operated by an entity other than the core network operator.

[0038] The MME 142 may be connected to each of the eNode-Bs 142a, 142b, 142c in the RAN 104 via an S1 interface and may serve as a control node. For example, the MME 142 may be responsible for authenticating users

of the WTRUs 102a, 102b, 102c, bearer activation/deactivation, selecting a particular serving gateway during an initial attach of the WTRUs 102a, 102b, 102c, and the like. The MME 142 may also provide a control plane function for switching between the RAN 104 and other RANs (not shown) that employ other radio technologies, such as GSM or WCDMA.

[0039] The serving gateway 144 may be connected to each of the eNode Bs 140a, 140b, 140c in the RAN 104 via the S1 interface. The serving gateway 144 may generally route and forward user data packets to/from the WTRUs 102a, 102b, 102c. The serving gateway 144 may also perform other functions, such as anchoring user planes during inter-eNode B handovers, triggering paging when downlink data is available for the WTRUs 102a, 102b, 102c, managing and storing contexts of the WTRUs 102a, 102b, 102c, and the like.

[0040] The serving gateway 144 may also be connected to the PDN gateway 146, which may provide the WTRUs 102a, 102b, 102c with access to packet-switched networks, such as the Internet 110, to facilitate communications between the WTRUs 102a, 102b, 102c and IP-enabled devices. An access router (AR) 150 of a wireless local area network (WLAN) 155 may be in communication with the Internet 110. The AR 150 may facilitate communications between APs 160a, 160b, and 160c. The APs 160a, 160b, and 160c may be in communication with STAs 170a, 170b, and 170c.

[0041] The core network 106 may facilitate communications with other networks. For example, the core network 106 may provide the WTRUs 102a, 102b, 102c with access to circuit-switched networks, such as the PSTN 108, to facilitate communications between the WTRUs 102a, 102b, 102c and traditional land-line communications devices. For example, the core network 106 may include, or may communicate with, an IP gateway (e.g., an IP multimedia subsystem (IMS) server) that serves as an interface between the core network 106 and the PSTN 108. In addition, the core network 106 may provide the WTRUs 102a, 102b, 102c with access to the networks 112, which may include other wired or wireless networks that are owned and/or operated by other service providers.

[0042] Routing protocols have been developed for policy-based routing for communication devices that can connect to two or more different networks simultaneously. Proposed herein are advanced IP flow routing techniques that enhance the IP routing capabilities of such communication systems that use a policy-based routing approach.

[0043] The following terms are used herein. A Multiple Interfaces Host (MIF) may be a host having the capability to concurrently attach to multiple IP Networks. An IP Network may be a computer network using the Internet Protocol (IP). For example, an IP network may be physical (e.g. Ethernet, Wireless LAN or cellular) or virtual (e.g. tunnel, virtual private network (VPN), or virtual local area network (VLAN)). The Access Network Discovery and Selection Function (ANDSF) is an entity within a 3rd Generation Partnership Project (3GPP) evolved packet core that may provide WTRUs with rules policing connection and usage of non-3GPP (e.g. IEEE 802.11 based Wi-Fi or IEEE 802.16 based WIMAX) and 3GPP networks.

[0044] An Inter-System Routing Policy (ISRP) may be used by an ANDSF node and may include a set of one or more flow distribution rules defining how IP traffic originating from a host may be distributed over the various IP Networks available to that host. A Flow Distribution Rule may be an ANDSF rule containing IP Flow information and an associated routing rule Information. For example, 3GPP Technical Standard (TS) 24.312 defines 3 different types of Flow Distribution Rules: ForFlowBased, ForServiceBased and ForNonSeamlessOffload.

[0045] An IP Flow may be a sequence of packets from a source host to a destination host, where the destination host may be another host, a multicast group, or a broadcast domain for example. IP Flow and IP Traffic are used interchangeably herein. Routing is the process of selecting a path for IP Traffic (i.e. IP Flow) in a network. At the host level, where the IP traffic originates, routing refers to the action of selecting the IP Network to send the IP Traffic to.

[0046] IP Flow Information may be information that identifies an IP Flow and may be included in an ANDSF node. For example, the IP Flow Information may correspond to an IP Flow node defined in 3GPP TS 24.312, where the IP Flow node may be a container that defines the characteristics of IP traffic. Routing Rule Information may be information used to perform IP Network selection for routing IP flows and may be included in an ANDSF node. For example, the Routing Rule Information may correspond to the Routing Rule node in 3GPP ST 24.312.

[0047] The TCP/IP stack may be part of the kernel of a network-capable computer system and may constitute the cornerstone of IP capabilities of the host. The TCP/IP stack may allow routing of IP packets based on their IP characteristics such as ports, IP address(es), and/or protocols. In scenarios with multiple available networks, increased online content and/or mobile communication devices with advanced capabilities, the mobile devices may be capable of connecting simultaneously to multiple networks. In these scenarios, the routing capabilities of communication devices may need to provide enhanced functionality such as a better user experience and/or increased communication capabilities, for example.

[0048] A communications standard, such as ANDSF, may provide routing information in the form of a routing policy, such as ISRP mentioned above, to help communication devices make better routing decisions for IP traffic over multiple different available networks. However, some TCP/IP stacks may be limited in their capability to identify IP flows and may only partially support such routing policies.

[0049] For example, an ISRP may be composed of one or more Flow Distribution containers that each may define different IP Flows and Routing Rules. Some of the characteristics included in the IP Flow information may be supported by most TCP/IP stacks (e.g. all IP capable hosts can route an IP Flow based on the destination IP address). However, any IP Flow characteristics defined in the ISRP that are not purely IP-based may not be supported by the certain TCP/IP stacks. In this case, it may not be possible to

identify an IP Flow based on the application from which it originates. Similarly, it may not be possible to identify an IP Flow based on the Domain Name (e.g. Domain Name System (DNS) entry) to which the IP flow is destined.

[0050] Advanced routing mechanisms to support communication devices that can access multiple networks simultaneously are described herein. Methodologies described herein address the routing limitations described above while maintaining compatibility with existing applications that use the TCP/IP stack and maintaining compatibility with the TCP/IP stack itself.

[0051] 3GPP specifications present some architecture enhancements to 3GPP networks to enable non-3GPP accesses, which include the support of the ANDSF. Management Objects (MOs), as defined in 3GPP specifications, may be used by the ANDSF and communication devices in the network. An example MO is the ISRP, which defines rules for routing traffic between the different interfaces of a MIF. Methodologies described herein allow implementation of full ISRP capabilities while preserving compatibility between the applications and the kernel's TCP/IP stack.

[0052] Approaches for intercepting socket calls, either in user space or kernel space, in order to realize various advanced routing manipulations are also described herein. These approaches may also be used in conjunction with methodologies used to identify flows based on their domain names, for example. These methodologies may be used together to achieve a domain name routing functionality, but may do so with different scope. For example, the methodologies described herein may be used to realize application name routing, which identifies flows using the application name from where the flow originated.

[0053] Techniques described herein improve the routing capabilities of the TCP/IP stack, and may, for example, implement the full capabilities of ISRP rules through the introduction of an Intercept Function. The Intercept Function may permit a MIF-capable computing device to make routing decisions based on traffic characteristics using techniques that may not be

available in TCP/IP stacks while maintaining compatibility with such various TCP/IP stacks. Routing of IP traffic based on IP characteristics such as IP address, port or protocol, may be supported by the Intercept Function. The Intercept Function may be further augmented to support routing based on other characteristics including, but not limited to the following characteristics: Application Based Routing according to which traffic may be routed based on the application which originates such traffic; and/or Domain Name Routing according to which traffic may be routed based on the Domain Name that is requested. Moreover, using an Intercept Function may allow a network operator to perform advanced flow routing to enhance the user experience of its subscribers by providing routing capabilities that may not otherwise be supported by a TCP/IP stack.

[0054] According to an embodiment, a method employing a User Space Intercept Function may intercept library calls made by applications in order to force a routing decision before the TCP/IP stack routing decision is made. When an application wants to perform IP-based communication, it may use the socket application programming interface (API). The socket API may be implemented in a user-space library, for example. The method may include intercepting calls made to that API in user-space so that a pre-routing decision may be made before reaching the TCP/IP stack in the kernel. For example, when an application requires a TCP connection, it may create a TCP socket that may act as an end-point to the IP communication. The application may establish a virtual connection of this TCP socket acting as an end-point to the peer server by using the “connect” socket API function. When the “connect” API call is made, the call may be intercepted and a routing decision may be made before it is allowed to reach the TCP/IP stack.

[0055] During the interception phase described above, the Intercept Function may examine the characteristics of the IP flow along with the routing rules established for the communication device. The routing rules may be, for example, the active ISRP for a device supporting ANDSF and/or rules coming from another source. The flow characteristics and/or the routing rules

may be used to make the Intercept Function's routing decision. If at least one rule applies to the flow, then the TCP/IP stack may be forced to route the flow as per the Intercept Function decision. If no routing rule applies to the flow, then the TCP/IP stack may route the flow as it normally does based on its own rules.

[0056] Some of the benefits introduced by the Intercept Function include, but are not limited, to the following. The Intercept Function may provide an opportunity to make a routing decision before the TCP/IP stack makes a routing decision. The interception code may be executed in the process space of the application making the socket API call, which may allow access to information such as the process name. This information may then be used to perform Application Based Routing features by the routing policy, which may not otherwise be supported by the TCP/IP stack. The interception code may be executed in each application process, which may allow for discovering the domain names that each application is trying to access before the domain name is converted to an IP address. This opportunity may then be used to perform Domain Name Routing by the routing policy, which may not otherwise be supported by the TCP/IP stack.

Component	Description
Kernel Space	Memory that may be reserved for running privileged kernel code, kernel extensions, and some (e.g. most) device drivers. The kernel may be the core of the computing system Operating System (OS).
User Space	Memory that may be reserved to run some or all processes, programs and/or libraries that the OS may execute and that may interact with the kernel.
Network Interface	A hardware device that may access an IP-based network.
TCP/IP stack	Component of the kernel that may provide end-to-end connectivity implementing how data should be formatted, addressed, transmitted, routed and/or received.
Process	Distinct application that may be executed in its own process space.
Networking Function	Within an application, functionality that may be in charge of performing network exchanges between the local endpoint and peer endpoints located on the network. Each application that accesses the network may have a different Networking Function. For example: - In a browser, the networking function may represent the code in charge of getting the HyperText Markup Language (HTML) data from the network. - In an e-mail client, the networking function may represent the code in charge of getting e-mail messages to and from the e-mail server.
Intercept Function	Code that may be in charge of implementing the Intercept Function.
Socket API	Computing library that may provide an API for sockets, used for data communication, which may be included in the O/S.
Policy Based Routing Function	Function that may be in charge of providing the policy routing rules applicable to a given communication device. For example: - ISRP policy on a ANDSF capable host - Routing policy from other policy-based routing system This Function may be part of the user space or the kernel space.

TABLE 1 - Component Description of User Space Intercept Function

[0057] The Intercept Function methodology may provide any of the following capabilities. The Intercept Function may perform application based routing, and/or domain name routing. The Intercept Function may allow routing based on traditional IP Flow characteristics (such as IP address, port, protocol, etc.) in addition to other characteristics such as application name and/or domain name. The use of the Intercept Function may also decouple the Intercept Function routing decision from the TCP/IP stack routing decision. Table 1 gives definitions of the components relating to the use of the Intercept Function, as described in more detail below and in the remaining figures.

[0058] FIG. 2 shows a block diagram of an example WTRU 200 including Intercept Functions 214₁ and 214₂ in the User Space 202, in accordance with an embodiment. As defined in Table 1, the WTRU 200 may include a User Space 202 and a Kernel Space 204, along with one or more network interfaces 206₁ to 206₃ that may connect to respective networks 230₁ to 230₃ (not shown). In this example, two processes 210₁ and 210₂, which may correspond to respective applications, are included in the User Space 202, where the processes 210₁ and 210₂ may respectively include, but are not limited to, any of the following elements: an Application Networking Function 212₁/212₂, an Intercept Function 214₁/214₂, and/or a Socket API 216₁/216₂. The Intercept Functions 214₁/214₂ may communicate with the Routing Policy Function 220 in order to obtain routing policy rules and/or information. Socket APIs 216₁/216₂ may interface with the TCP/IP Stack 208 located in the Kernel Space 204 to make routing decisions and route IP flow traffic to the network interfaces 206₁ to 206₃.

[0059] Applications running on computer systems may use a socket API to perform IP Network communication, where the socket API may be part of the OS. When an application initiates an IP Network session for an IP flow, the application may use socket API calls. Examples of socket API calls include, but are not limited to any of the following: establishing a TCP connection to a remote server (e.g. connect() may be used); and/or sending a UDP packet (connectionless) (e.g. sendto() may be used). An Intercept Function may

intercept the socket API call and, in some circumstances, may make a routing decision for the IP flow in place of the TCP/IP Stack, as explained in detail below.

[0060] FIG. 3 shows a flow diagram of an example flow routing procedure 300 employing an Intercept Function in the User Space, in accordance with an embodiment. The example flow routing procedure 300 may be performed, for example, by WTRU 200 of FIG. 2, such that the User Space, Application Networking Function, Intercept Function, Socket API, and TCP/IP Stack referred to in FIG. 3 may be as shown in the WTRU 200. With reference to FIG. 3, at 302, an Application Networking Function (e.g. 212₁ of FIG. 2) may initiate an IP network session for an IP flow by making a socket call to a Socket API (e.g. 216₁ of FIG. 2), where the socket call may be diverted to an Intercept Function (e.g. 214₁ of FIG. 2).

[0061] Depending on the nature of the platform, the programming environment and availability of the OS source code, multiple packaging options may be used to perform socket API calls diversion. According to an example approach, the Socket API may be a dynamic library provided by the computing device's OS. A library pre-loading strategy may be used to divert the socket API calls. In this case, applications calling a Socket API function may end-up calling the equivalent function defined in the Intercept Function.

[0062] According to another example approach, the Socket API library may be modified and recompiled to add the Intercept Function code. In this case, the OS may natively provide the Intercept Function functionality. Applications that use the Socket API dynamically may benefit from the Intercept Function functionality. Applications that may be linked statically with the library may also benefit from the Intercept Function functionality. According to another example approach, the applications themselves may be modified and recompiled to directly call the Intercept Function API.

[0063] With reference to FIG. 3, at 304, the Intercept Function may receive and analyze the socket call according to policy-based routing rules and

based on at least one of the following characteristics: IP flow characteristics and/or routing policy information.

[0064] The IP flow characteristics may include, for example, the destination IP address and port and/or the protocol. The IP flow characteristics may be available in the parameters passed to the Socket API function (e.g. Socket API 216₁ in FIG. 2). Performing the interception in the Intercept Function also may offer the opportunity to access characteristics that may not be available to the TCP/IP stack such as, for example, the application name and/or the destination domain name, which may be needed by policy-based routing such as the ISRP but may not be available to the TCP/IP stack.

[0065] The destination domain name may not be readily available in the intercepted socket API calls. However, the same interception mechanism may be used to intercept other library calls that can retrieve the destination domain name. The destination domain name to IP address mapping may then be stored for later access (e.g. during socket API intercept processing as described above).

[0066] The routing policy information may include a list of rules that apply to the IP Flows that the policy writer wants to control. The details for accessing the routing policy information may be implementation dependent. According to an example implementation, a system may be designed in which a static policy may reside in a file or may be static and hardcoded in the Intercept Function. The routing policy information may be dynamically retrieved from a server that resides in the infrastructure of the Network Operator, for example. In the example of FIG. 2, the Intercept Function 214₁ (and Intercept Function 214₂) may obtain routing policy information from a Routing Policy Function 220.

[0067] With reference to FIG. 3, at 304, the Intercept Function may perform an analysis for the IP flow using the routing policy information and/or the flow characteristics. The analysis may be performed according to rules specified by a policy based routing specification, such as 3GPP TS 24.312 for

the example case where ANDSF is used. At 306, The Intercept Function may determine if the IP flow matches at least one of the policy-based routing rules. The analysis may result in various outcomes including, but not limited to: the IP Flow matching one of the rules specified in the routing policy; or the IP Flow not matching any rule specified in the routing policy. Both example outcomes are described below.

[0068] In the case that the IP flow matches at least one of the policy-based routing rules, at 308, the Intercept Function may select an IP network interface from a plurality of network interfaces for the IP flow and may bind the IP flow to the selected IP network interface using the Socket API. The Intercept Function may bind the socket to a specific IP network interface by using the socket API to set socket options. Binding the socket may ensure that all the traffic flowing through that socket will be sent over the selected network interface. The mechanisms used to bind the IP Flow over a certain network interface, or block an IP Flow, after the Intercept Function decision is taken, may be implementation dependent.

[0069] At 310 (assuming the Interception Function didn't block the IP flow), the Intercept Function may invoke the original call flow by invoking the system socket API call (made at 302 by the Application Networking Function) and the TCP/IP stack may route traffic for the IP flow to the selected IP network interface. In this case, the TCP/IP stack flow routing may happen in accordance with the decision made by the Intercept Function. The Intercept Function may decide to forbid (i.e. block) the IP Flow in accordance with the routing policy, in which case the socket API call may not need to be invoked and an error code may be returned to the application via the Application Networking Function to indicate a failure. For example, the intercept function may decide to block an IP flow in accordance with a routing rule in the policy.

[0070] In the case that the IP flow does not match at least one of the policy-based routing rules, at 312, no further processing may happen at the Intercept Function and the Intercept Function may invoke the socket API call.

In this case, at 314, the flow routing decision for the IP flow is made by the TCP/IP stack, which may select an IP network interface from the plurality of network interfaces for the IP flow.

[0071] The example flow routing procedure 300 may be performed for one or more applications/processes and/or subsets of specifically targeted processes on a communications device (e.g. Process 210₁ and/or Process 210₂ in FIG. 2), such that such any part of the flow routing procedure 300 may occur in parallel for multiple applications/processes on the same device. The example flow routing techniques described above include advanced routing based on application and/or domain name. However, other parameters may be similarly used to identify flows for policy-based routing employing an Intercept Function. The Intercept Function methodology may be independent of specific flow identification methodologies.

[0072] FIG. 4 shows a block diagram of an example WTRU 400 including an Intercept Function 414 in the Kernel Space 404, in accordance with an embodiment. The WTRU 400 may include a User Space 402 and a Kernel Space 404, along with one or more network interfaces 406₁ to 406₃ that may connect to respective outside networks 430₁ to 430₃ (not shown). In this example, two processes 410₁ and 410₂, which may correspond to respective applications, are included in the User Space 402, where the processes 410₁ and 410₂ may respectively include, but are not limited to, any of the following elements: an Application Networking Function 412₁/412₂, and/or a Socket API 416₁/416₂. The Kernel Space 404 may include Intercept Function 414 in communication with the Routing Policy Function 420, from which the Intercept Function 414 may obtain routing policy rules and/or information. Socket APIs 416₁/416₂ may connect to the Intercept Function 414, which may in turn communicate with the TCP/IP Stack 408 to make routing decisions and route IP flow traffic to the network interfaces 406₁ to 406₃.

[0073] When an application wants to perform IP-based communication, it may use the Socket API. The Socket API functionality may use a User Space Library that may act as a thin layer of execution before control is transferred

to Kernel Space through the use of a System Call. According to an embodiment, socket API calls may be intercepted in the Kernel Space so that a routing decision may be made. For example, when an application needs a TCP connection, it may create a TCP socket that may act as an end-point to the IP communication and may establish a virtual connection of this end-point to the peer server by using the “connect” Socket API function. The User Space “connect” function may perform the processing to trigger the corresponding Kernel Space execution of the “connect.” When the Kernel space “connect” is triggered, the call may be intercepted by the Intercept Function and a routing decision may be made before it is allowed to reach the TCP/IP stack.

[0074] During this phase, the Intercept Function may examine the characteristics of the IP flow along with the routing rules established for the communication device. The routing rules may be, for example, the active ISRP policy for a device supporting ANDSF, or they may be coming from another source. Both the flow characteristics and the routing rules may be used to make the Intercept Function routing decision. If at least one rule applies to the flow, then the TCP/IP stack may be forced to route the flow as per the Intercept Function decision. If no rule applies to the flow, then the legacy TCP/IP stack may route the flow as it normally does based on its own rules.

[0075] The use of the Intercept Function may provide an opportunity to make a routing decision before the legacy TCP/IP stack makes a routing decision. The execution of the interception code in the kernel may allow access to information such as the process name. This information may then be used to perform Application Based Routing as determined by the routing policy, for example, which may not be possible if routing is only determined by the TCP/IP stack. The execution of the interception code in the kernel may allow for discovering the domain names that each application is trying to access before the domain name is converted to an IP address. The accessed domain name(s) may be used to perform Domain Name Routing as determined by the routing policy for example, which may not be possible if routing is only determined by the TCP/IP stack.

[0076] The Intercept Function approach may provide, but is not limited to, the following routing capabilities: performing application based routing; performing domain name routing; allowing routing based on IP Flow characteristics such as IP address, port, and/or in addition to characteristics such as application name or domain name; and decoupling the Intercept Function routing decision from the TCP/IP stack routing decision.

[0077] FIG. 5 shows a flow diagram of an example flow routing procedure 500 employing an Intercept Function in the Kernel Space, in accordance with an embodiment. According to the flow routing procedure 500, networking calls coming from User Space may be intercepted by positioning the Intercept Function in the Kernel Space, where the Intercept Function may make a routing decision before the TCP/IP stack makes a routing decision.

[0078] The example flow routing procedure 500 may be performed, for example, by WTRU 400 of FIG. 4, such that the User Space, Application Networking Function, Intercept Function, Socket API, and TCP/IP Stack referred to in FIG. 5 may be as shown in the WTRU 400. With reference to FIG. 5, at 502, an Application Networking Function (e.g. 412₁ of FIG. 2) may initiate an IP network session for an IP flow by making a socket API call to a Socket API (e.g. 416₁ of FIG. 4), where the Kernel interface for the Socket API Call may translate to a System Call.

[0079] The User-Space Socket API may be implemented as a thin-layer library that applications link with, and which may act as a bridge between the User-Space portion of a Socket Call and its Kernel Space counterpart. The Kernel interface for such a Socket API Call may translate to a System Call. A System Call may be how a program in the User Space requests a service from an operating system's kernel in the Kernel Space. This technique may include hardware related services, creation and execution of new processes and communication with kernel services such as scheduling, for example. Implementing System Calls may use a control transfer that may involve architecture specific features. For example, the implementation of a System Call may use a Software Interrupt mechanism. Interrupts may transfer

control to the Kernel Space code, and User Space software may set up some registers with the desired System Call number and may execute the software interrupt.

[0080] With reference to FIG. 5, at 504, in response to the socket API function being called, the Socket API function, acting as a User-Space library, may store the associated system call number in a pre-defined register and prepare the data obtained from the function call parameters for the Kernel Space. An invocation of an API function may contain parameters that may be propagated down to the kernel along with the call. For example, the `sendto()` function parameters might include the data to send, the destination IP address and/or the port.

[0081] At 506, the User Space code may trigger the Kernel Space code to initiate processing by the Intercept Function and the Intercept Function may intercept the socket API call. For example, the User Space socket API code may invoke the Kernel Space code using a software interrupt. The Kernel Space may invoke the system call interrupt service routine (ISR), which may read the system call number in the pre-defined register from User Space to determine which system call is being requested.

[0082] The kernel ISR may extract the system call parameters and may invoke the Kernel side system call routine by doing a lookup in a table that may keep the relationship between the system call number and the system-call routine's memory location, for example. This table may be referred to as the System Call Table. In order to perform interception of the Kernel Space functions, the System Call Table may be modified at kernel run-time so that each intercepted system call routine memory location points to the location of a compatible Intercept Function. By doing so, any call to socket functions in the User Space may trigger the Intercept Function methods instead of the other kernel methods.

[0083] Depending on the nature of the platform, the programming environment, and availability of the OS source code, multiple packaging options may be used to perform diversion of socket API calls. Examples of

such packaging options include, but are not limited to, the following: the socket API calls may be diverted using a loadable module strategy that may produce the desired socket API diversion at load time; and/or the kernel may be modified and recompiled to modify the default system call table or modify the default kernel-side system call routine.

[0084] With reference to FIG. 5, at 508, once the socket API call is intercepted on the Kernel Space side, the Intercept Function may perform a routing analysis according to policy-based routing rules and/or using the routing policy information and/or the flow characteristics.

[0085] Examples of IP flow characteristics include, but are not limited to, the destination IP address and port, and/or the protocol, which may be available in the parameters passed to the Socket API function. Performing the interception in the Intercept Function may offer the opportunity to access additional characteristics that may not be available to the TCP/IP stack such as the application name and the destination domain name. The destination domain name may not readily be available in the intercepted socket API calls. However, the same interception mechanism may be used to intercept other kernel calls that may retrieve the destination domain name. The destination domain name to IP address mapping may be stored for later access, for example during socket API intercept processing.

[0086] Examples of routing policy information include, but are not limited to, a list of rules that may apply to the IP Flows that the policy writer wants to control. Accessing the routing policy information may be implementation dependent. For example, a static policy may reside in a file or may be static and may be hardcoded in the Intercept Function. For example, the routing policy information may be dynamically retrieved from a server that resides in the infrastructure of the Network Operator. In another example, the Intercept Function (e.g. 414 in FIG. 4) may obtain routing policy information from a Routing Policy Function in the Kernel Space (e.g. 420 in FIG. 4).

[0087] With reference to FIG. 5, at 508, the Intercept Function may perform routing analysis for the IP flow using the routing policy information and/or the flow characteristics. The analysis may be performed according to rules specified by the policy based routing specification, such as 3GPP TS 24.312 when ANDSF is used, for example. At 510, The Intercept Function may determine if the IP flow matches at least one of the policy-based routing rules. The analysis may result in various outcomes including, but not limited to: the IP Flow matching one of the rules specified in the routing policy; or the IP Flow not matching any rule specified in the routing policy. Both example outcomes are described below.

[0088] In the case that the IP flow matches at least one of the policy-based routing rules, at 512, the Intercept Function may select an IP network interface from a plurality of network interfaces for the IP flow and may bind the IP flow to the selected IP network interface using the Socket API to set socket options. Binding the socket may ensure that all the traffic flowing through that socket may be sent over the chosen interface. The Intercept Function phase may be executed before sending data to the TCP/IP stack. The mechanism used to bind the IP Flow over a certain interface (or block of an IP Flow) after the Intercept Function decision is taken may be implementation dependent.

[0089] At 514, assuming the Interception Function didn't block the IP flow, the Intercept Function may invoke the kernel side system call and IP flow traffic may be routed to the selected IP network interface via the TCP/IP stack (e.g. TCP/IP Stack 408 in FIG. 4). In this case, the TCP/IP stack flow routing may happen in accordance with the decision made by the Intercept Function. The Intercept Function may decide to forbid (i.e. block) the IP Flow in accordance with the routing policy, in which case the system call may not need to be invoked and an error code may be returned to the application via the Application Networking Function to indicate a failure.

[0090] In the case that the IP flow does not match at least one of the policy-based routing rules, at 516, no further processing may happen at the

Intercept Function and the Intercept Function may invoke the kernel side system call. In this case, at 518, the flow routing decision for the IP flow is made by the TCP/IP stack, which may select an IP network interface from the plurality of network interfaces for the IP flow. The flow routing procedure 500 of FIG. 5 may apply to User Space processes because it may be implemented in the Kernel Space, which is common to all processes in the communication device.

[0091] Further, the flow routing procedure 500 may be performed for one or more applications/processes and/or subsets of specifically targeted processes on a communications device (e.g. Process 410₁ and/or Process 410₂ in FIG. 4), such that such any part of the flow routing procedure 400 may occur in parallel for multiple applications/processes on the same device. The example flow routing techniques described above include advanced routing based on application and/or domain name. However, other parameters may be similarly used to identify flows for policy-based routing employing an Intercept Function. The Intercept Function methodology may be independent of specific flow identification methodologies.

[0092] Embodiments

[0093] 1. A method for advanced flow routing.

[0094] 2. The method of embodiment 1 performed by a wireless transmit/receive unit (WTRU).

[0095] 3. The method as in any of the previous embodiments comprising accessing a plurality of networks.

[0096] 4. The method as in any of the previous embodiments comprising on each of a plurality of processes running in parallel initiating an Internet Protocol (IP) network session for an IP flow associated with an application by making a socket call to a socket application programming interface (API).

[0097] 5. The method as in any of the previous embodiments comprising on each of a plurality of processes running in parallel intercepting

the socket call and performing routing analysis to select a network interface from the plurality of network interfaces for the IP flow.

[0098] 6. The method of embodiment 5 wherein the routing analysis is in accordance with policy-based routing rules.

[0099] 7. The method of embodiments 5 or 6 wherein the routing analysis is based on at least one of: IP flow characteristics and routing policy information.

[0100] 8. The method of embodiment 7 wherein the IP flow characteristics include at least one of: an application name and a destination domain name.

[0101] 9. The method as in any of the previous embodiments further comprising on each of a plurality of processes running in parallel, on a condition that the IP flow matched at least one of the policy-based routing rules, invoking the socket call.

[0102] 10. The method as in any of the previous embodiments further comprising on each of a plurality of processes running in parallel, on a condition that the IP flow matched at least one of the policy-based routing rules routing traffic for the IP flow to the network interface selected by the intercept function.

[0103] 11. The method as in any of the previous embodiments further comprising receiving the routing policy information from a routing policy function.

[0104] 12. The method as in any of the previous embodiments occurring, at least in part, in a user space.

[0105] 13. The method as in any of the previous embodiments occurring, at least in part, in a kernel space.

[0106] 14. The method as in any of the previous embodiments, wherein the initiating the IP network session and the intercepting the socket call occur in a user space, and the routing traffic occurs in a kernel space.

[0107] 15. The method as in any of the previous embodiments, wherein the initiating the IP network session occurs in a user space, and the intercepting the socket call and routing traffic occur in a kernel space.

[0108] 16. The method as in any of the previous embodiments further comprising, on a condition that the IP flow didn't match any of the policy-based routing rules, invoking the socket call.

[0109] 17. The method as in any of the previous embodiments further comprising, on a condition that the IP flow didn't match any of the policy-based routing rules, selecting a network interface from the plurality of network interfaces.

[0110] 18. The method as in any of the previous embodiments further comprising, on a condition that the IP flow didn't match any of the policy-based routing rules, routing traffic for the IP flow to the selected network interface.

[0111] 19. The method as in any of the previous embodiments, wherein the IP flow characteristics further include at least one of: destination IP address and port, or protocol.

[0112] 20. The method as in any of the previous embodiments, wherein intercepting the socket call uses a library pre-loading strategy to produce a socket API diversion.

[0113] 21. The method as in any of the previous embodiments occurring in a kernel space, wherein the socket call is translated to a system call for the kernel space.

[0114] 22. The method as in any of the previous embodiments, wherein invoking the socket call is on a condition that the IP flow was not blocked.

[0115] 23. The method as in embodiments 20-22, wherein the WTRU is configured to support 3rd Generation Partnership Project (3GPP) Access Network Discovery and Selection Function (ANDSF).

[0116] 24. The method as in any of the previous embodiments, wherein the routing policy information is in accordance with an Inter-System Routing Policy (ISRP).

[0117] 25. The method as in any of the previous embodiments performed, at least in part, by an application networking function.

[0118] 26. The method as in any of the previous embodiments performed, at least in part, by an intercept function.

[0119] 27. The method as in any of the previous embodiments performed, at least in part, by a Transmission Control Protocol/Internet Protocol (TCP/IP) stack.

[0120] 28. The method as in any of the previous embodiments performed, at least in part, by a socket application programming interface (API).

[0121] 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 media include electronic signals (transmitted over wired or wireless connections) and computer-readable storage media. 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

What is claimed is:

1. A wireless transmit/receive unit (WTRU) comprising:
 - a plurality of network interfaces each configured to access a respective network from a plurality of networks;
 - a plurality of processes running in parallel, each process comprising:
 - an application networking function configured to initiate an Internet Protocol (IP) session for an IP flow associated with an application by making a socket call to a socket application programming interface (API);
 - an intercept function configured to intercept the socket call and select a network interface from the plurality of network interfaces for the IP flow in accordance with policy-based routing rules and based on at least one of: IP flow characteristics, or routing policy information, wherein the IP flow characteristics include at least one of: an application name, or a destination domain name; and
 - on a condition that the IP flow matches at least one of the policy-based routing rules;
 - the intercept function configured to invoke the socket call; and
 - a Transmission Control Protocol/Internet Protocol (TCP/IP) stack configured to route traffic for the IP flow to the network interface selected by the intercept function.

2. The WTRU of claim 1, wherein the intercept function is configured to receive the routing policy information from a routing policy function.

3. The WTRU of claim 1, wherein the application networking function, the intercept function, and the socket API are resident in a user space of the WTRU, and the TCP/IP stack is resident in a kernel space of the WTRU.

4. The WTRU of claim 1, wherein the application networking function and the socket API are resident in a user space of the WTRU, and the intercept function and the TCP/IP stack are resident in a kernel space of the WTRU.

5. The WTRU of claim 1, further comprising:
on a condition that the IP flow did not match any of the policy-based routing rules, the socket API is configured to invoke the socket call, and select a network interface from the plurality of network interfaces; and
the TCP/IP stack is configured route traffic for the IP flow to the network interface selected by the socket API.

6. The WTRU of claim 1, wherein the IP flow characteristics further include at least one of: destination IP address, destination port, or protocol.

7. The WTRU of claim 1, wherein the intercept function intercepts the socket call using a library pre-loading strategy to produce a socket API diversion.

8. The WTRU of claim 1, wherein the intercept function is resident in a kernel space of the WTRU, and the socket call is translated to a system call for the kernel space.

9. The WTRU of claim 1, wherein the intercept function is configured to invoke the socket call on a condition that the intercept function did not block the IP flow.

10. The WTRU of claim 1 configured to support 3rd Generation Partnership Project (3GPP) Access Network Discovery and Selection Function (ANDSF), wherein the routing policy information is in accordance with an Inter-System Routing Policy (ISRP).

11. A method for advanced flow routing performed by a wireless transmit/receive unit (WTRU), the method comprising:

accessing a plurality of networks via a corresponding plurality of network interfaces;

on each of a plurality of processes running in parallel:

initiating an Internet Protocol (IP) session for an IP flow

associated with an application by making a socket call to a socket application programming interface (API);

intercepting the socket call and selecting a network interface from the plurality of network interfaces for the IP flow in accordance with policy-based routing rules and based on at least one of: IP flow characteristics or routing policy information, wherein the IP flow characteristics include at least one of: an application name or a destination domain name; and

on a condition that the IP flow matches at least one of the policy-based routing rules, invoking the socket call; and routing traffic for the IP flow to the selected network interface.

12. The method of claim 11, further comprising:
receiving the routing policy information from a routing policy function.

13. The method of claim 11, wherein the initiating the IP session and the intercepting the socket call occur in a user space of the WTRU, and the routing traffic occurs in a kernel space of the WTRU.

14. The method of claim 11, wherein the initiating the IP session occurs in a user space, and the intercepting the socket call and routing traffic occur in a kernel space of the WTRU.

15. The method of claim 11, further comprising:
on a condition that the IP flow did not match any of the policy-based routing rules, invoking the socket call, selecting a network interface from the plurality of network interfaces, and routing traffic for the IP flow to the selected network interface.
16. The method of claim 11, wherein the IP flow characteristics further include at least one of: destination IP address, destination port, or protocol.
17. The method of claim 11, wherein intercepting the socket call uses a library pre-loading strategy to produce a socket API diversion.
18. The method of claim 11 occurring in a kernel space of the WTRU, wherein the socket call is translated to a system call for the kernel space.
19. The method of claim 11, wherein the invoking the socket call is on a condition that the IP flow was not blocked.
20. The method of claim 11, wherein the WTRU is configured to support 3rd Generation Partnership Project (3GPP) Access Network Discovery and Selection Function (ANDSF), and wherein the routing policy information is in accordance with an Inter-System Routing Policy (ISRP).

100

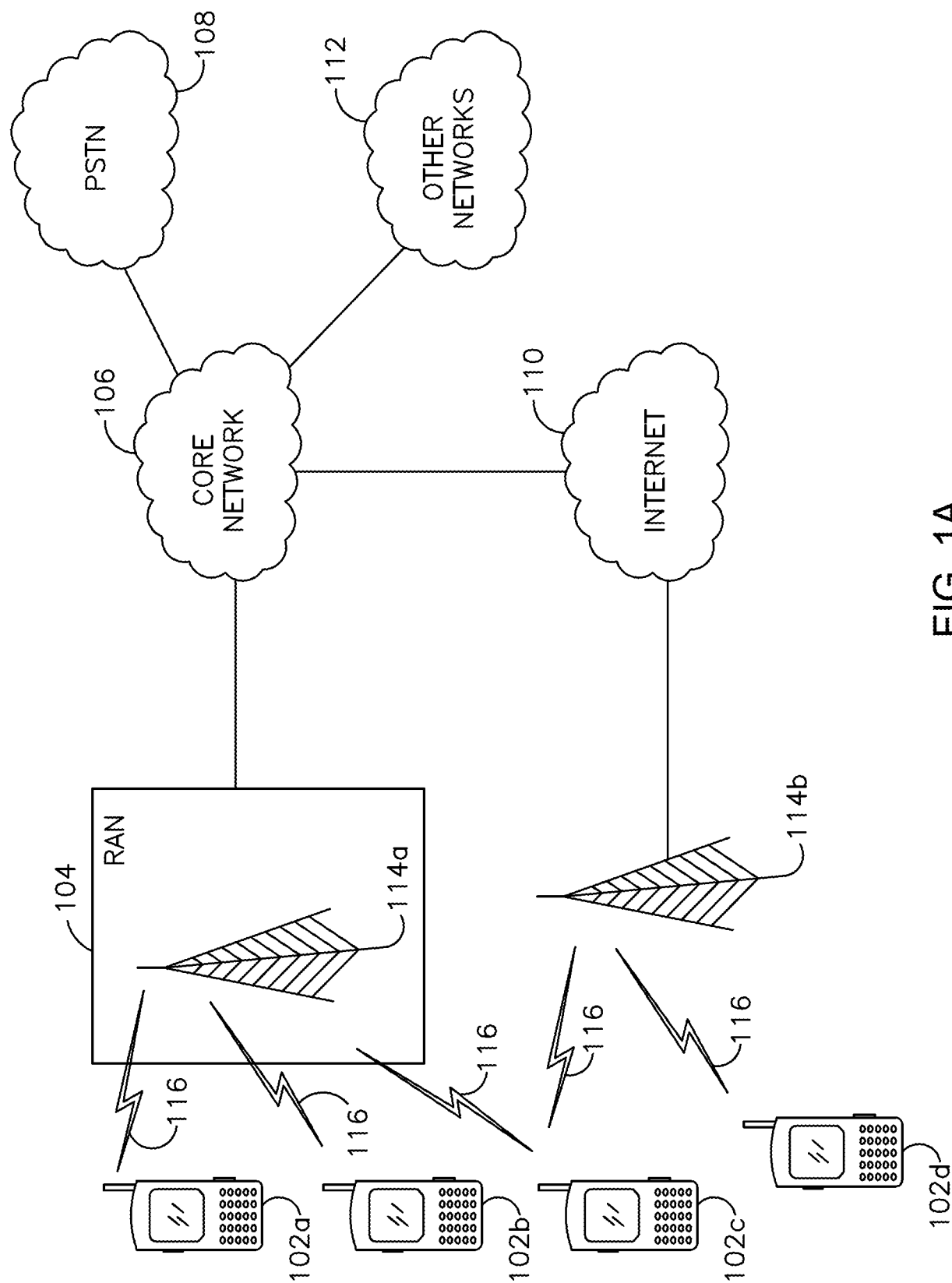


FIG. 1A

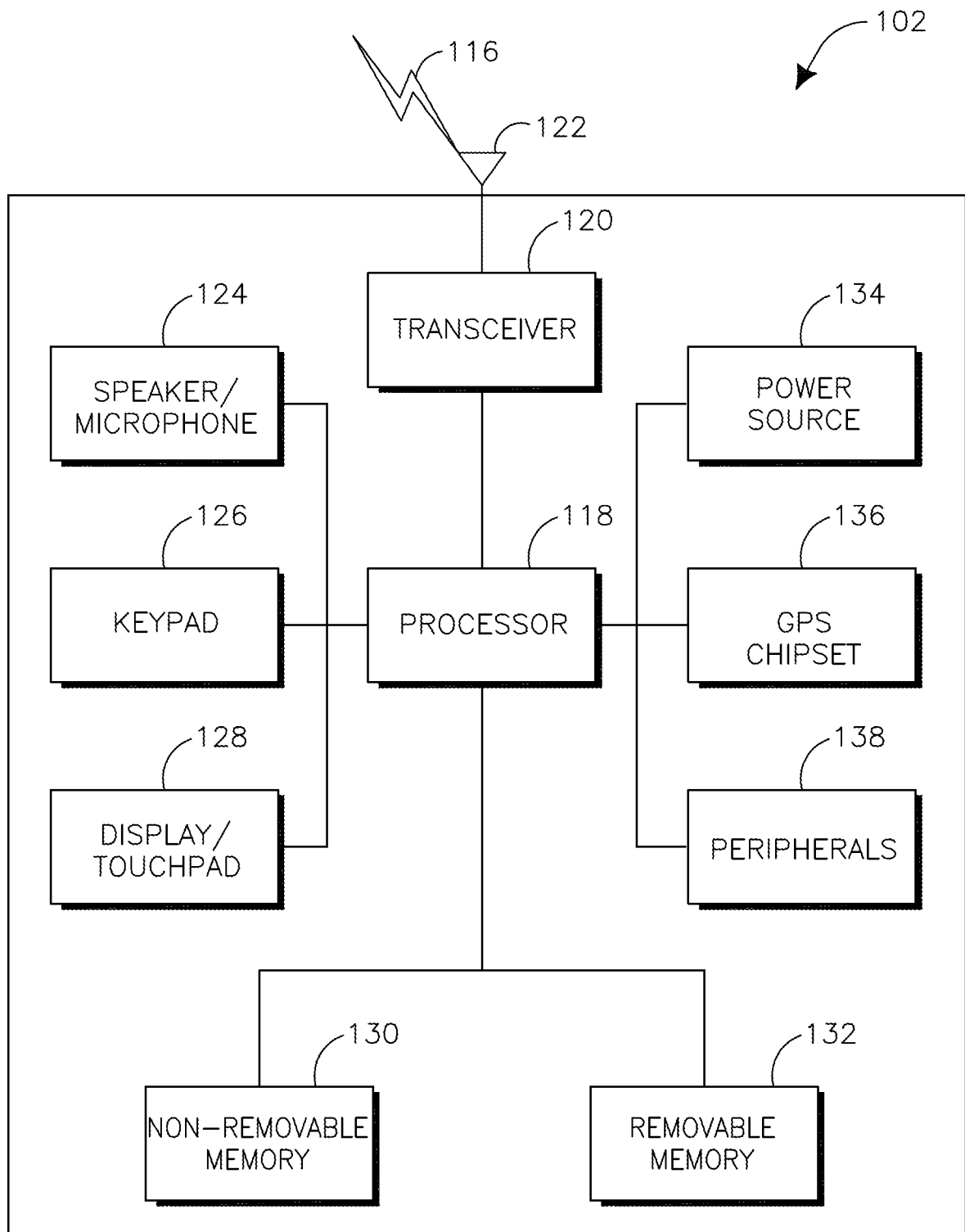


FIG. 1B

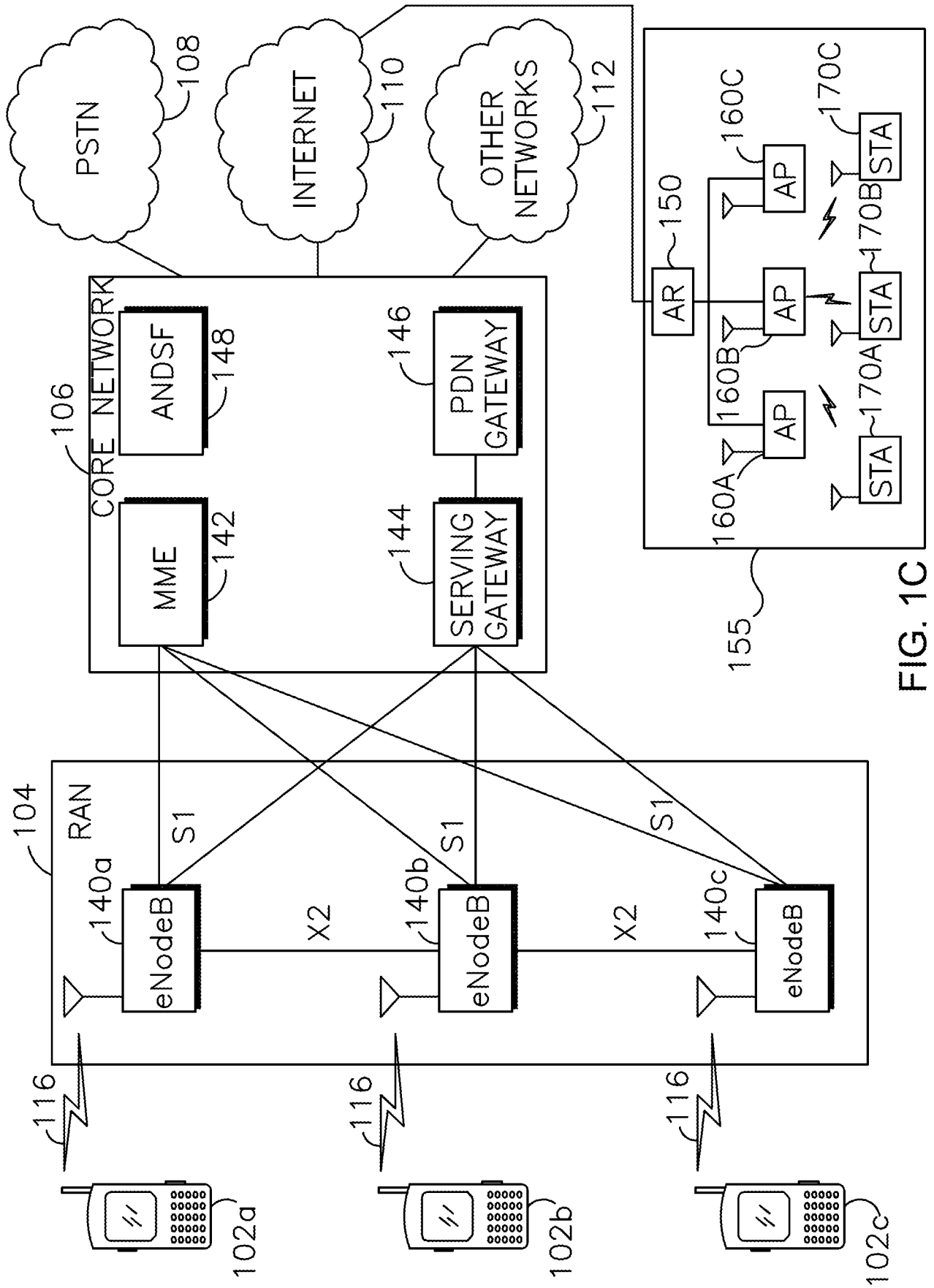


FIG. 1C

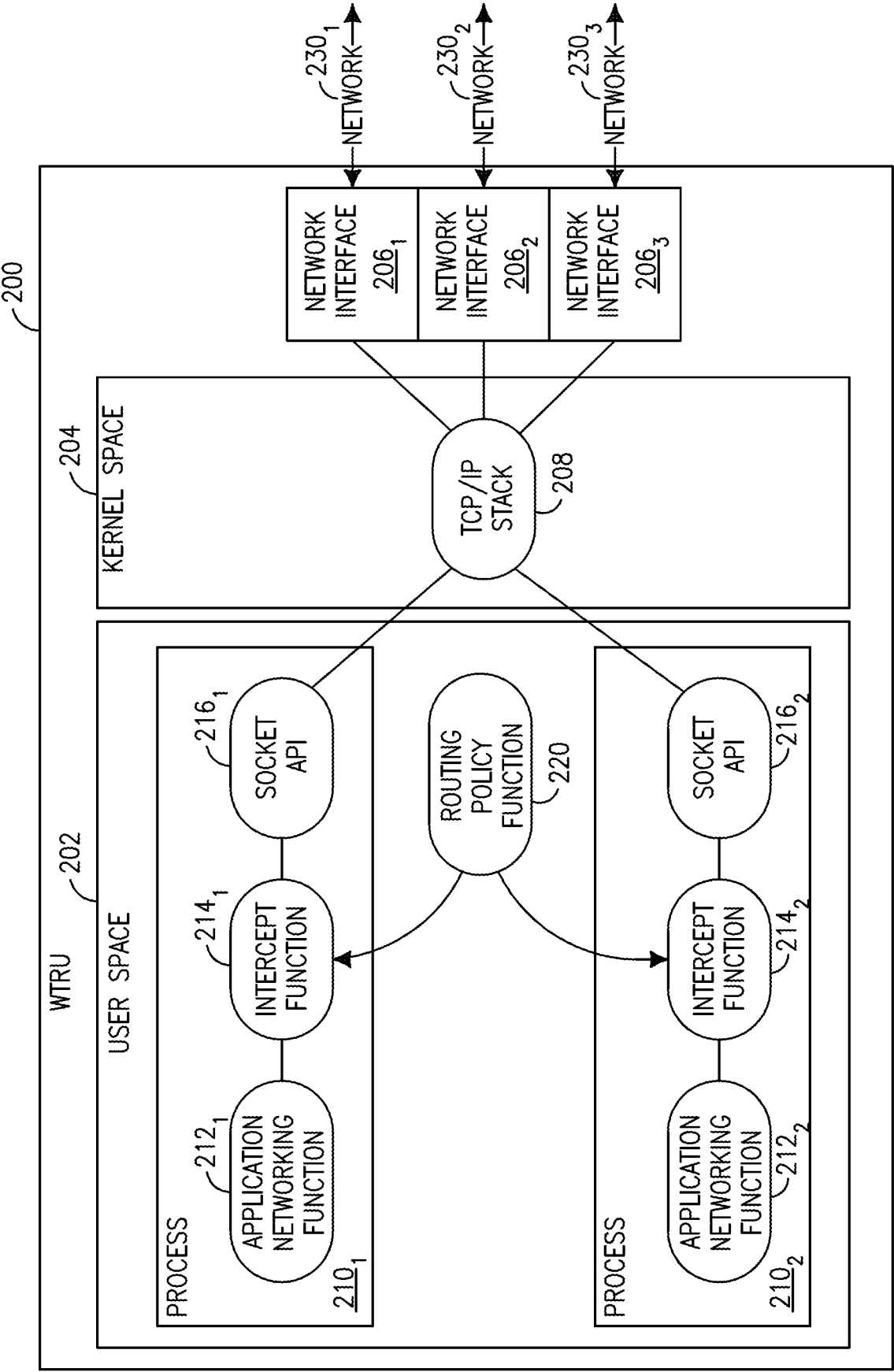
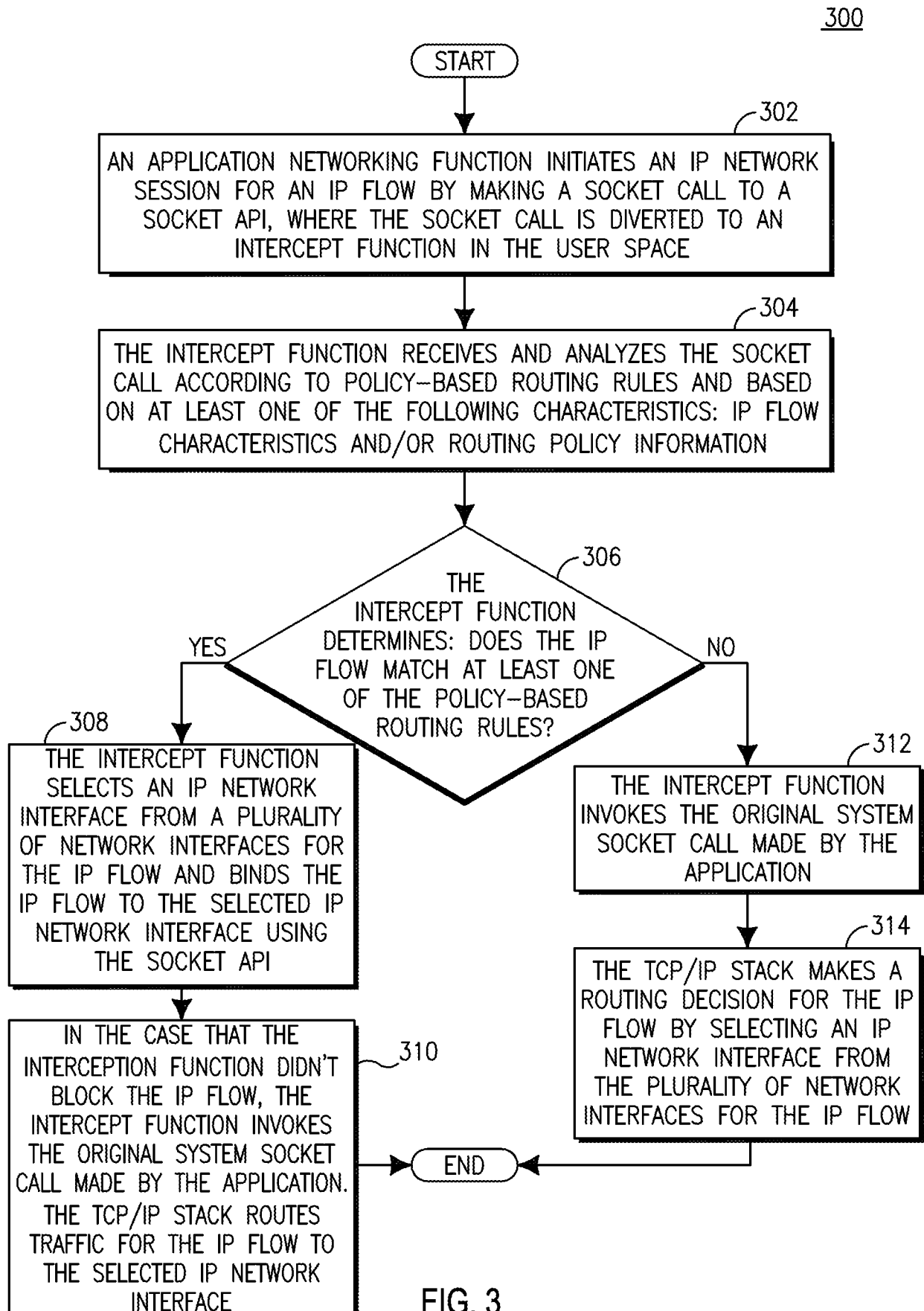


FIG. 2



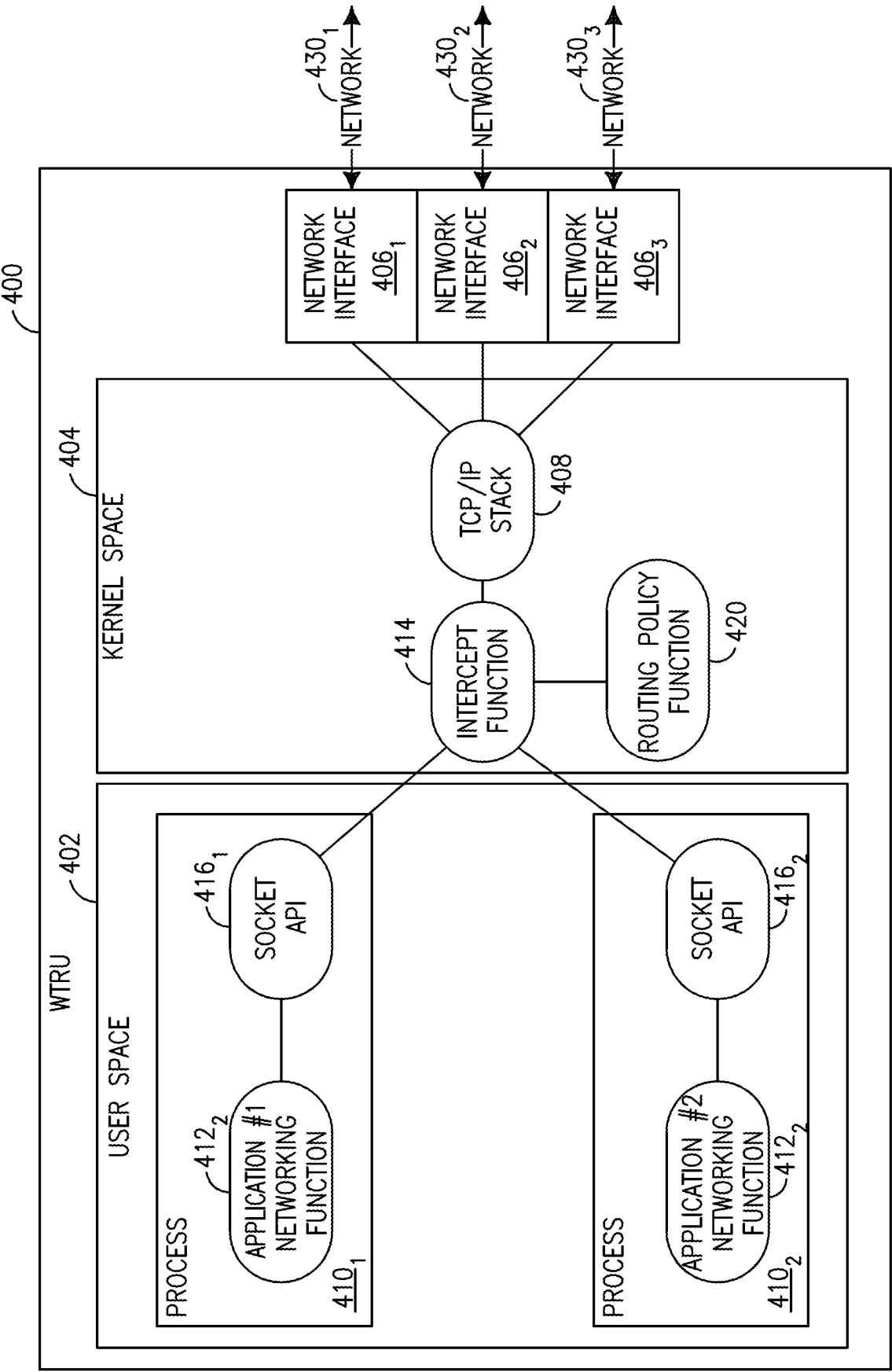
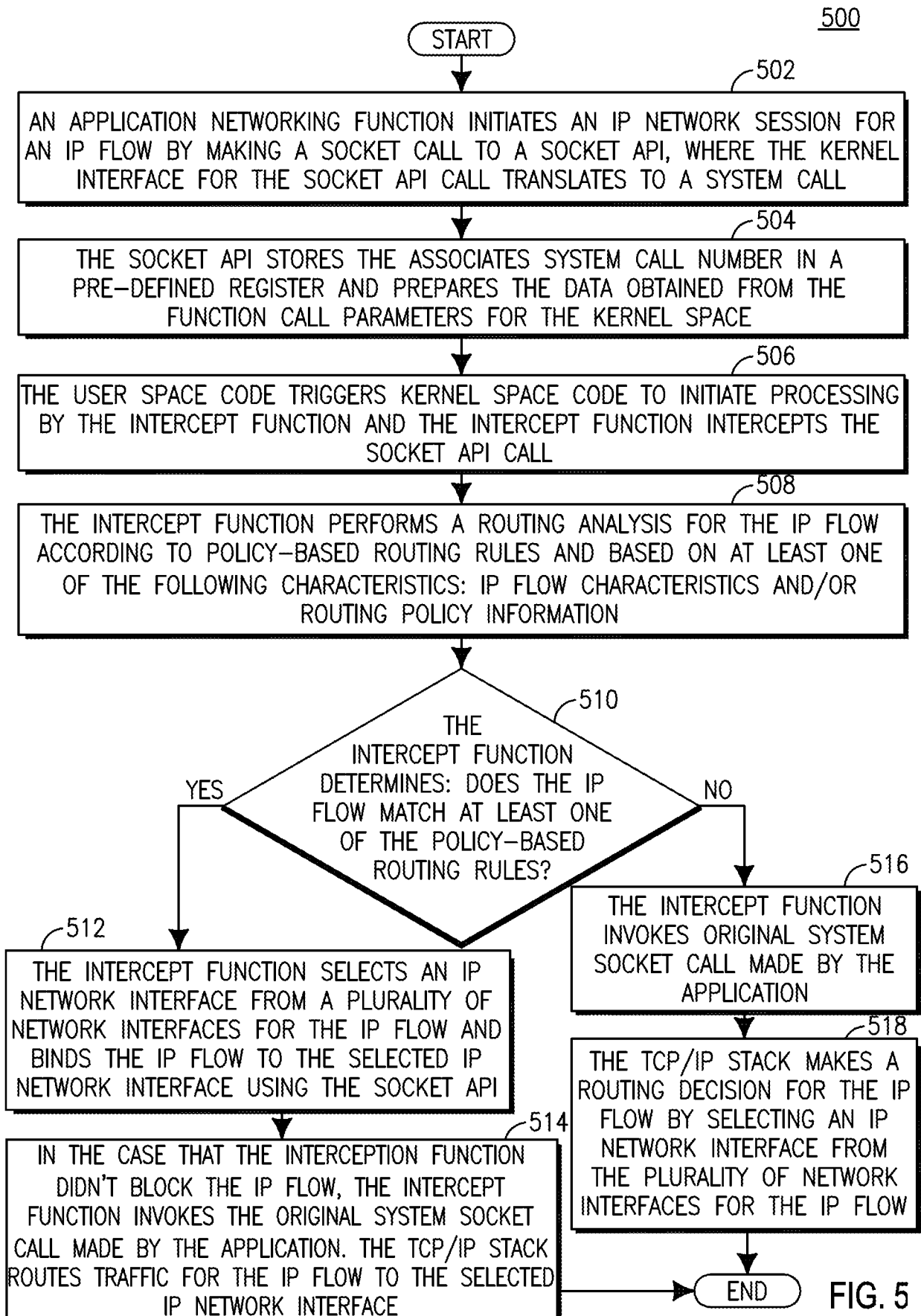


FIG. 4



INTERNATIONAL SEARCH REPORT

International application No

PCT/US2015/040917

A. CLASSIFICATION OF SUBJECT MATTER
 INV. H04L29/06
 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, IBM-TDB, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2011/188376 A1 (STUPAR PATRICK [DE] ET AL) 4 August 2011 (2011-08-04) figure 3 paragraph [0028] paragraph [0064] - paragraph [0069] -----	1-20
A	US 2012/208502 A1 (PERRAS MICHELLE [CA]) 16 August 2012 (2012-08-16) abstract figure 1a -----	1-20
A	US 2010/257276 A1 (SAVOLAINEN TEEMU ILMARI [FI]) 7 October 2010 (2010-10-07) figure 1 paragraph [0013] ----- -/-	1-20



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

21 October 2015

Date of mailing of the international search report

28/10/2015

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
 NL - 2280 HV Rijswijk
 Tel. (+31-70) 340-2040,
 Fax: (+31-70) 340-3016

Authorized officer

Suciu, Radu

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2015/040917

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2012/270538 A1 (MEYLAN ARNAUD [US] ET AL) 25 October 2012 (2012-10-25) figure 9 paragraph [0005] - paragraph [0009] -----	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2015/040917

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2011188376 A1	04-08-2011	CN 102771097 A	07-11-2012
		EP 2524480 A1	21-11-2012
		JP 2013517708 A	16-05-2013
		KR 20120123424 A	08-11-2012
		TW 201146058 A	16-12-2011
		US 2011188376 A1	04-08-2011
		WO 2011088406 A1	21-07-2011

US 2012208502 A1	16-08-2012	CA 2827098 A1	16-08-2012
		CN 103370952 A	23-10-2013
		EP 2673970 A1	18-12-2013
		JP 2014509495 A	17-04-2014
		KR 20140012089 A	29-01-2014
		SG 192712 A1	30-09-2013
		SG 10201403161U A	30-07-2014
		TW 201238383 A	16-09-2012
		TW 201528861 A	16-07-2015
		US 2012208502 A1	16-08-2012
		WO 2012109272 A1	16-08-2012

US 2010257276 A1	07-10-2010	US 2010257276 A1	07-10-2010
		WO 2009065996 A1	28-05-2009

US 2012270538 A1	25-10-2012	CN 103503520 A	08-01-2014
		EP 2700270 A1	26-02-2014
		JP 5694602 B2	01-04-2015
		JP 2014513899 A	05-06-2014
		TW 201246981 A	16-11-2012
		US 2012270538 A1	25-10-2012
		WO 2012145720 A1	26-10-2012
