Exhibit B-B

Based on Headwater's apparent positions as to the scope of the patent's claims, as best they can be deciphered, the reference(s) charted below anticipate(s) or at least render(s) obvious the identified claims.

To the extent Plaintiff contends that a primary reference does not disclose and/or render obvious any elements of the asserted claims of the '117 Patent, the elements would nonetheless have been obvious to a POSITA at the time of the invention by the combination of the primary reference with the knowledge of someone skilled in the art and/or one or more other primary references listed as teaching those elements from the other "B" Exhibits and/or with one or more of the following primary and/or secondary references listed as teaching those elements in the claim charts below:

- U.S. 8,195,961 ("Ogawa") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b). Ogawa is a patent that was filed on January 9, 2009.
- U.S. 7,673,007 ("Mulligan") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(e). Mulligan is a patent that was published on June 12, 2008, and filed on August 6, 2007.
- U.S. 2009/0298478 ("Tyhurst") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(e). Tyhurst was filed on May 29, 2008.
- U.S. 7,975,147 ("Qumei") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(e). Qumei was filed on March 30, 2004.
- U.S. 2005/0207379 ("Shen") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Shen was filed on March 18, 2005 and published on September 22, 2005.
- U.S. 2005/0108571 ("Lu") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Lu was filed on May 19, 2004 and published on May 19, 2005.
- U.S. 2003/0145229 ("Cohen") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Cohen was filed on January 31, 2002 and published on July 31, 2003.
- U.S. 6,961,574 ("Stage") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Stage was filed as a PCT application on March 6, 2000 and issued on November 1, 2005.

- U.S. 2006/0018283 ("Lewis") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Lewis was filed on August 24, 2005 and published on January 26, 2006.

- U.S. 2007/0088801 ("Levkovitz") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Levkovitz was filed on May 4, 2006 and published on April 19, 2007.

- WO2002/032072 ("Huskins") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b). Huskins was filed as a PCT application on October 3, 2001, and published on April 18, 2002.

- U.S. 7,925,717 ("Chou") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Huskins was filed a on September 23, 2003.

- U.S. 2004/0186918 ("Lonnfors") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Lonnfors was filed on March 21, 2003 and published on September 23, 2004.

- U.S. 2009/0282256 ("Rakic") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(e). Rakic was filed on May 12, 2008, and published on November 12, 2009.

- U.S. 2007/0214245 ("Hämäläinen") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Hämäläinen was filed on March 7, 2006, and published on September 13, 2007.

- U.S. 7,082,615 ("Ellison") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Ellison was filed on September 22, 2000, and published on July 25, 2006.

- EP 1,853,044 B1 ("Shenfield") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a). Shenfield was published on July 11, 2007.

- EP Pat. App. Pub. No. 1,909,463A1 ("Hwang") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(e). Hwang was filed on October 16, 2006, and published on April 9, 2008.

- U.S. 8,041,816 ("Ozaki") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Ozaki was filed on February 8, 2000, and published on August 24, 2006.

- U.S. 2009/0307715 ("Santamaria") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(e). Santamaria was filed on August 25, 2008.

- The Transport Layer Security ("TLS") Protocol, V. 1.1 ("TLS v1.1"), available at https://www.ietf.org/rfc/rfc4346.txt, was published in April 2006 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- The TLS Protocol Version v 1.0 (Jan. 1999) ("TLS v1.0"), available at https://datatracker.ietf.org/doc/html/rfc2246, was published in January 1999 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- The Secure Sockets Layer ("SSL") Protocol, V. 3.0, available at https://web.archive.org/web/19970614041044/http://home.netscape.com/eng/ssl3/ssl-toc.html and https://web.archive.org/web/19970617034012/http://home.netscape.com/eng/ssl3/3-SPEC.HTM#1, was published in June 1997 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- Open Mobile Alliance; Multimedia Messaging Service Architecture Overview (MMSARCH) specification ("OMA"), available at https://www.openmobilealliance.org/release/MMS/V1_1-20040715-A/OMA-WAP-MMS-ARCH-V1_1-20040715-A.pdf, was published in July 2004 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- Open Mobile Alliance; OMA-ERELD-MMS-v1_2-20030923-C, Enabler Release Definition for MMS Version 1.2," available at https://www.openmobilealliance.org/release/MMS/V1_2-20030923-C/OMA-ERELD-MMS-V1_2-20030923-C.pdf was published in September 2003 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- Generic Content Download Over The Air, Version 1.0, OMA-Download-OTA-V1_0-20040625-A, Open Mobile Alliance (June 25, 2004), available at https://www.openmobilealliance.org/release/Download/V1_0-20040625-A/OMA-Download-OTA-V1_0-20040625-A.pdf, was published on June 25, 2004 and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- Security Engineering, 2nd Ed. ("Anderson") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Anderson published in 2008.

- Mostafa, "Transporting data between wireless applications using a messaging system—MMS" (Wireless Comms. and Mobile Computing, July 7, 2006) ("Mostafa") was published on July 7, 2006and thus qualifies as prior art under pre-AIA 35 U.S.C. § 102(a) and (b).

- U.S. 2006/0080527 ("Novack") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Novack was filed on August 27, 2004, and published on April 13, 2006.

- U.S. 7,088,990 ("Isomursu") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Novack was filed on October 30, 1998, and published on August 8, 2006.

- U.S. 2004/0243993 ("Okonnen") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Okonnen was filed on March 24, 2004, and published on December 2, 2006.

- EP1298518 ("Hashimoto") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b). Hashimoto was filed on September 27, 2002, and published on April 2, 2003.

- U.S. 2006/0031873 ("Fahrny") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Fahrny was filed on August 9, 2004, and published on February 9, 2006.

- U.S. 2008/0228892 ("Staack") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a). Staack was filed on May 22, 2008, and published on September, 18, 2008.

- CN101136837A ("Fan") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a). Fan was filed on September 21, 2007, and published on March 5, 2008.

- U.S. 6,986,037 ("Assmann") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Assmann was filed on May 31, 2000, and published on January 10, 2006.

- KR20010082789 ("Lee II") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b). Lee II was filed on February 21, 2000, and published on August 31, 2001.

- U.S. 2007/0190978 ("White") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). White was filed on February 1, 2005, and published on August 16, 2007.

- U.S. 2003/0078985 ("Holbrook") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Holbrook was filed on October 23, 2001, and published on April 24, 2003.

- U.S. 5,675,507 ("Bobo") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Bobo was filed on April 28, 1995, and published on October 7, 1997.

- https://web.archive.org/web/20071218065028/http://dbus.freedesktop.org:80/doc/dbus-specification.html#uuids ("D-Bus Specification V0.12") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). D-Bus Specification V0.12 was published at least as early as December 18, 2007. Earlier versions of the D-Bus specification were also published before this date.

- https://android.googlesource.com/platform/external/dbus/+/7f0990abc74a3c6aa76211a064b57f081690d340 and associated source code ("D-BUS commit 7f0990abc74a3c6aa76211a064b57f081690d340"). D-BUS commit 7f0990abc74a3c6aa76211a064b57f081690d340 was publicly available at least as early as December 8, 2008. Earlier versions of this code was available before this date.

- U.S. 2012/0208568 ("Cooley") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(e). Cooley was filed on December 3, 2008.

- https://web.archive.org/web/20071218065028/http://dbus.freedesktop.org:80/doc/dbus-specification.html#uuids ("D-Bus Specification V0.12") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). D-Bus Specification V0.12 was published at least as early as December 18, 2007. Earlier versions of the D-Bus specification were also published before this date.

- https://android.googlesource.com/platform/external/dbus/+/7f0990abc74a3c6aa76211a064b57f081690d340 and associated source code ("D-BUS commit 7f0990abc74a3c6aa76211a064b57f081690d340"). D-BUS commit 7f0990abc74a3c6aa76211a064b57f081690d340 was publicly available at least as early as December 8, 2008. Earlier versions of this code was available before this date.

- Peter James, Secure Portable Execution Environments: A Review of Available Technologies ("James") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b). James was published at least as early as January 12, 2008.

- Malcom Taylor et. al., Extended Abstract: A Hardware-Assisted Data Hiding Based Approach in Building High Performance Secure Execution Systems ("Taylor") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a). Taylor was published at least as early as June 9, 2008.

- U.S. 8,099,764 ("Herzog") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(e). Herzog is a patent that was filed on December 17, 2007.

- U.S. 2007/0203969 ("Wakasa") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b)/(e). Wakasa is a publication that was published on August 30, 2007 and is an application that was filed on August 28, 2006.
- U.S. 5,253,212 ("Ross") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b)/(e). Ross is a patent that was published on July 17, 2001 and is an application that was filed on February 17, 1998.
- U.S. 2007/0123206 ("Satake") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b)/(e). Satake is a publication that was published on May 31, 2007 and is an application that was filed on October 30, 2006.
- "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence" ("XMPP") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b). XMPP is a publication that was published in October 2004.
- "Session Initiation Protocol Extension for Instant Messaging" ("SIPE") qualifies as prior art under at least pre-AIA 35 U.S.C. §§ 102(a)/(b). SIPE is a publication that was published on August 30, 2002.
- U.S. 2006/0009198 ("Kasai") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Kasai was filed on June 9, 2005, and published on January 12, 2006.
- U.S. 7,305,697 ("Alao") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Alao was filed on May 16, 2005, and published on December 4, 2007.
- U.S. 2005/0169285 ("Wills") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Wills was filed on December 30, 2004, and published on August 4, 2005.
- U.S. 5,751,813 ("Dorenbos") qualifies as prior art under at least pre-AIA 35 U.S.C. § 102(a)/(b)/(e). Dorenbos was filed on April 29, 1996, and published on May 12, 1998.

Samsung incorporates in this chart all applicable qualifications, clarifications, and other statements made in the cover document of Samsung's Invalidity Contentions. This invalidity claim chart is based on Samsung's present understanding of the asserted claims, and Headwater's apparent construction of the claims, as set forth in Headwater's Infringement Contentions. These invalidity assertions are not an admission that Samsung concedes or acquiesces to any claim construction(s) implied or suggested by Headwater or the associated infringement claim charts. Nor is Samsung asserting any claim construction positions through these charts, including whether the preamble is a limitation. Samsung also does not concede or acquiesce that any asserted claim satisfies the requirements of 35 U.S.C. §§ 112 or 101, and submits these invalidity contentions only to the extent Headwater's assertions may be understood.

The excerpts cited herein are exemplary. For any claim limitation, Samsung may rely on excerpts cited for any other limitation and/or additional excerpts not set forth fully herein to the extent necessary to provide a more comprehensive explanation for a reference's disclosure of a limitation. Where an excerpt refers to or discusses a figure or figure items, that figure and any additional descriptions of that figure should be understood to be incorporated by reference as if set forth fully herein. Similarly, where an excerpt cites to particular text referring to a figure, the citation should be understood to include the figure and related figures as well. To the extent any cell lacks citations to the charted reference(s), this should not be taken as an admission that the reference(s) do(es) not disclose the corresponding limitation but rather indicates that Samsung does not presently have sufficient information regarding the charted reference(s) based on Samsung's present understanding of the claim limitation.

These invalidity contentions are not an admission by Samsung that the accused products or components, including any current or past version of the products or components, are covered by, or infringe the asserted claims, particularly when these claims are properly construed and applied.

'117 Claims	**Exemplary Disclosures**
[1 pre]. A network system comprising:	To the extent the preamble is a limitation, The references below disclose and/or render obvious this element. For example, see the following passages and/or figures, as well as related disclosures: Huskins at 2:11-18: "Referring to Figure 1, a network may include at least one server, head-end or network operator 10 and a plurality of clients 12 (only one of which is shown) . The server 10 may be coupled to a plurality of clients (including the client 12) through a

'117 Claims	Exemplary Disclosures
	distribution system that may be based on a wired system or a wireless or broadcast system. Examples of such networks include television distribution networks such as digital video broadcasting systems." Chou at Abstract: "Techniques are disclosed for controlling interaction between an enterprise application and a mobile client device in a communication system. Push content is generated in a wireless secure server or other type of server, responsive to information received in the wireless secure server from the enterprise application. The push content is deliverable from the wireless secure server to the mobile client device over a wireless network. The wireless secure server receives from the mobile client device, responsive to the push content, a request for additional information identifiable at least in part by the push content. The additional information is deliverable from the server to the mobile client device over the wireless network." Chou at 2:3-13: "In accordance with one aspect of the invention, push content is generated in a wireless secure server or other type of server, responsive to information received in the wireless secure server from the enterprise application. The push content is deliverable from the wireless secure server to the mobile client device over a wireless network. The wireless secure server receives from the mobile client device, responsive to the push content, a request for additional information identifiable at least in part by the push content. The additional information is deliverable from the server to the mobile client device over the wireless network." Chou at 2:27-37: "The mobile client device may register with the enterprise application through interaction with the wireless secure server. For example, the wireless secure server

'117 Claims	Exemplary Disclosures
	may comprise a password-protected register connector through which registration information associated with the mobile client device is supplied from the wireless secure server to the enterprise application. The wireless secure server may be operative to obtain user and device profile information from the mobile client device, and to store the profile information temporarily until the profile information is extracted from the wireless secure server by the enterprise application." Rakic at [0043]-[0044]: "FIG. 1 illustrates the concepts described herein. As shown, network 100 may include a client device 102, an administrator device 104, a gateway 106, and a database device 108. Depending on the implementation, network 100 may include additional, fewer, or different devices than those illustrated in FIG. 1. For example, network 100 may include multiple client devices 102. "Device 102 may include any of the following devices that have the ability to or are adapted to communicate and interact with another device, such as a radiotelephone or a mobile telephone with ultra wide band or Bluetooth communication capability; a personal communications system (PCS) terminal that may combine a cellular radiotelephone with data processing, facsimile, and/or data communications capabilities; an electronic notepad, a laptop, and/or a personal computer that communicate with wireless peripherals (e.g., a wireless keyboard, speakers, etc.); a personal digital assistant (PDA) that can include a telephone; a Global Positioning System device and/or another type of positioning device; a gaming device or console; a peripheral (e.g., wireless headphone); a digital camera; or another type of computational or communication device." Shenfield at [0006]: "The present system and method preferably provide for a dynamic content delivery architecture and system that allows generic applications and content providers to be

'117 Claims	Exemplary Disclosures
	added to the system without the necessity to modify the architecture. Specifically, the present system and method allows for a mobile device to become a dynamic application platform in which applications can be added and content provided to the mobile device, where the architecture of the dynamic content delivery system does not limit the type of application that can be installed on the device nor the type of content that the device receives." Shenfield at [0024]: "Service provider 120 communicates over wireless network 130 with a push client 140 that is located on a mobile device. Push client 140 will be described in more detail below. Push client 140 receives the content that is being delivered from content provider 110 and can communicate the content with a client application 150, which ultimately consumes the content." Shenfield at [0067]: "As with push server 410 of Figure 4, push client 510 includes various blocks that deal with messaging, various blocks that deal with metadata, and various blocks that deal with both messaging and metadata." Shenfield at [0080]: "Content expiry and replacement block 566 is the same as content replacement and expiry block 466 of Figure 4. The expiry of content and replacement of content can thus be handled at push client 510 in addition to the push server or push proxy." Ogawa at Abstract "A data encryption transmission system and associated methodology is provided including a data input site that compresses and encrypts data based on a shared encryption key and then transmits the compressed and encrypted data to an external network. A database server which is operably linked to the external network and

'117 Claims	Exemplary Disclosures
	stores, manages, transmits, and receives data. A removable storage stores an identification code, and a client site which is configured to receive the removable storage generates an encryption key based on the identification code, receives encrypted data from the external network, decrypts and expands the received data based on the shared encryption key, encrypts the data based on the encryption key and saves the encrypted data, and finally decrypts the encrypted data and outputs the data using the encryption key." Ogawa at 1:15-43 "BACKGROUND The present advancements are directed to encryption methodologies and a portable storage device for storing data thereto, and more particularly to a data preservation system including a removable storage device such as a USB memory drive which may be used, for example, to acquire materials, documents, etc. via an information network such as the Internet and encrypting, transmitting, and preserving the data within the removal storage. In commercial settings, sales personnel commonly present marketing and product information to potential customers via a personal computing device. Naturally, as the degree of documentation and data increases with inventory, sales volume and the like, the amount of marketing and product information increases proportionately. In this way, a large amount of sensitive data may need to be stored and presented to the portable platform. Increasingly, up to date information may be acquired remotely, over wireless networks via the Internet. Of course, public networks expose sensitive data to potential data theft, packet sniffing, and hacking with respect to the enterprise web interface. Furthermore, even in tightly managed, remote access, if the remote

'117 Claims	Exemplary Disclosures
	location is not also closely monitored for security issues, theft and tampering of the data from the downloaded destination can occur. Accordingly, a removable storage device, configurable for remote deployment is provided to address the deficiencies noted above." Ogawa at 1:45-64 "SUMMARY OF EXEMPLARY ASPECTS OF THE ADVANCEMENTS In one exemplary aspect of the current invention, a data encryption system contains a data input site, a database server, a removable storage, and a client site. The data input site compresses and encrypts data based on a shared encryption key, and transmits the compressed and encrypted data to an external network. The database server is connected to the external network and stores, manages, transmits, and receives the compressed and encrypted data. The removable storage stores an identification code. The client site is adapted to be operably linked with the removable storage and generates an encryption key based on the identification code. The client site then receives the compressed and encrypted data from the external network, decrypts and expands the compressed and encrypted data based on the shared encryption key, encrypts the data based on the encryption key, and saves the encrypted data. The client site decrypts the encrypted data and outputs the data using the encryption key." Ogawa at 4:23-36 "The exemplary data input site 1 of FIG. 1 is a hardware platform of a personal computing device such as a PC employing an Intel Pentium processor. The instruction set of units 11-15, as well those of database server 2 and client site 3 may be provided as a utility application, background daemon, or component of an operating system, or modified TCP/IP stack executing in conjunction with a local processor and operating system such as Microsoft VISTA®, Unix, Solaris, Linux,

'117 Claims	Exemplary Disclosures
	Apple MAC-OS and other systems known to those skilled in the art. Alternatively, those skilled in the art will recognize the applicability to mobile devices such as PDAs, phones, and portable entertainment devices which employ Symbian, Microsoft Mobile® and other mobile operating systems." Cohen at [0032]: [0032] Initially, a notification sink such as a wireless device negotiates security information with the notification source out-of-band from the notification mechanism. In this description and in the claims, negotiating "out-of-band from the notification mechanism" means that the security information is negotiated without the message transit points becoming apprised of the security information. Instead, the security information is shared only between the notification source and sink. When a notification is to be sent, the notification source generates a notification such as a push message that includes an encrypted form of the notification, as well as clear-text supplemental information that may be used to decrypt the notification. The notification sink then uses the security information previously negotiated, as well as the supplemental information provided in the notification, to decrypt the notification. The message transit points were not apprised of the security information needed to decrypt the notification. Accordingly, the notification traverses the message transit points without the content of the notification becoming clear to the message transmit points. Novack at Abstract: "Inter-process communications between a first process and a second process are secured. A first process recognizes that inter-process communications must be initiated with a second process. A first computer that supports the first process submits a request to initiate inter-process communications with a second computer that supports the second process. The second computer replies to a request to initiate

'117 Claims	Exemplary Disclosures
	inter-process communications by replying to the first computer with a request for information of a digital certificate that authenticates the first process." Novack at [0013]: "The present invention enables secure inter-process communications. The implementation of secure inter-process communications will be described in the context of a client-server network and a peer-to-peer network. Of course, the present invention is also applicable to environments other than the networks described herein, including any network environment that includes inter-process communications between different processes running on one or more computers." Novack at [0015]: "According to an aspect of the present invention, a method is provided for securing inter-process communications between a first process and a second process. The method includes receiving, from a device that supports the first process, a request to initiate inter-process communications. The method also includes replying to the device with a request for information of a digital certificate that authenticates the identity of the first process." Novack at [0020]: "According to yet another aspect of the present invention, the first process executes on a first server and the second process executes on a second server. The first server and the second server are provided within a secured perimeter." *See also* Novack at [0026]. Novack at [0021]: "According to an aspect of the present invention, a computer-readable medium is provides that stores a program that secures inter-process communications between a

'117 Claims	Exemplary Disclosures
	first process and a second process. The computer-readable medium includes a receiving code segment that receives, from a device that supports the first process, a request to initiate inter-process communications. The computer-readable medium also includes a digital certificate requesting code segment that replies to the device with a request for information of a digital certificate that authenticates the identity of the first process." *See also* Novack at [0027]. Novack at [0033]: "FIG. 1 shows an exemplary method for securing inter-process communications. The process starts at S100, and a first process begins to run on a first computer at S102. A determination is made that the services of a second process are needed at S105, and the first process requests initiation of an inter-process communications session at S110. The request at S110 marks the beginning of inter-process communications. The first computer may send the request to a second computer that runs a second process in the same network. The first computer receives an acknowledgement and a request for authentication at S115. In the example above, the second computer analyzes the initial request and generates an acknowledgement and request for authentication that is received by the first computer at S115." Novack at [0034]: "The analysis on the second computer may be performed by the requested process or another process. As an example, the second computer may analyze all incoming communications requests and demand authentication of all requesters. The second computer may demand authentication of only requestors requesting inter-process communications sessions. Of course, the second computer may demand authentication of requestors for only one or more particularly sensitive processes among many. The analysis at the second computer may be performed by the

'117 Claims	Exemplary Disclosures
	requested process or by a secondary security program that performs security analysis for all communications or all inter-process communications. Accordingly, when the request is initially received, the second computer determines that authentication is necessary before an inter-process communications session can proceed." Novack at [0035]: "In one example, the first process is running on an application server that provides application services over the internet to requesting clients. The application server interacts with a management server that manages multiple databases as a back-end service. The first process may be run when a user contacts an application server and downloads presentation layer elements of an application to interact with the application server. The application includes instructions to invoke additional processes for predetermined user requests. Accordingly, the application will request the services of a second process at S110." Novack at [0044]: "FIG. 2 shows an exemplary network architecture for secure inter-process communications. As shown, a **computer** 201 communicates through a **network** 220 with an **application server** 230. As an example, the **application server** 230 may be a web server that provides internet applications for users. The user downloads a web page onto the **computer** 201 when the user contacts an internet address corresponding to the desired internet application. Typically, the web page is accessed and downloaded using hyper text transport protocol (HTTP)." Novack at [0045]: "The **application server** 230 communicates through another **network** 240 with **management server** 250. As an example, the **network** 240 may be a secure local area network that enables the **application**

'117 Claims	Exemplary Disclosures
	server 230 to obtain back-end processing. Of course, the network 240 may be the same network as the network 220." Novack at [0046]: "The management server 250 manages databases 254 and 256. The databases 254 and 256 store data associated with the service provided by the application server 230. As an example, the application server 230 may provide internet services for accounts of a service provider. Thus, the application server 230 may provide services for a bank, a utility, or a content provider such as a news service. While the user of the computer 201 is intentionally provided with direct access to the application server 230, the configuration of the network shown in FIG. 2 is intended to help deny the user of the computer 201 direct access to the management server 250 and the databases 254, 256. Further, to ensure that the communications between the application server 230 and the management server 250 are secure, a digital certificate is installed on the application server 230, and the digital certificate is bound to the application server 230 and/or a process running on the application server 230." Novack at [0047]: "In an embodiment, the application server 230 and the management server 250 communicate using the secure sockets layer (SSL) protocol (i.e., HTTPS), for secure and trustworthy communications. A secure sockets layer session begins with an exchange of messages known as a handshake. In the embodiment of FIG. 2, the handshake occurs between the application server 230 and the management server 250. The handshake includes the application server 230 contacting the management server 250. In response to the application server 230 contacting the management server 250, the management server 250 sends the application server 230 a digital certificate. The application server 230 responds by providing the management server 250 with a digital certificate. Of course, an exchange of two

'117 Claims	Exemplary Disclosures
	digital certificates does not need to occur in every case; rather, a single digital certificate may be used when only one party (e.g., the application server 230) to a communications session needs to be authenticated." Novack at [0048]: "The **management server** 250 may require authentication of the **application server** 230 for each instance of inter-process communications. In this regard, even if the management server's 250 only function is to process requests for the **application server** 230, the **management server** 250 may need to authenticate the application server for each request for inter-process communications when, for example, the **application server** 230 widely interacts with the public and is therefore subject to unknown risks. The **management server** 250 may request verification of the application server's 230 address, the particular requesting function, and any other information that will help authenticate the request. Of course, the **management server** 250 may demand a digital certificate that is bound to the **application server** 230 or even a particular process that runs on the **application server** 230." Novack at [0058]: "The **device** 400 is not limited to a client or server. Rather, the **device** 400 may be a router, a proxy, an intermediary, or any other type of device that supports a process which at least occasionally requires inter-process coordination and communication. In an embodiment, the **device** 400 is a peer in a peer-to-peer network. Peer-to-peer network devices are particularly in need of the present invention, as peer devices often communicate with unknown devices that can be used to initiate unauthorized interaction. Using secure inter-process communications as described herein allows a device to ensure the authenticity of not just the (unknown) user and device from which communications are received, but also the actual process supported by a peer communications device. In this regard, a user in a peer-to-peer network may not always be aware when their device has been co-opted to initiate unauthorized

'117 Claims	Exemplary Disclosures
	interactions. Accordingly, unauthorized interactions can be addressed because communications are authenticated for the specified processes on the **device** 400. Accordingly, if an unauthorized interaction is attempted in a peer-to-peer network, the unauthorized interaction is effectively discouraged because the **device** 400 will not accept communications from processes that are not authenticated." Novack at [0064]: "The device in **FIG.** 4 may include multiple processes that communicate with each other using the secure inter-process communications of the present invention. Additionally, although the terms "client" and "server" are used to refer to devices, the devices may be defined and distinguished by a set of software instructions installed as a pre-packaged program. Further, while examples of the uses of the present invention include secure client/server networks and peer-to-peer networks, secure inter-process communications can be used to enhance security in any inter-process communications environment. Further, while the embodiment shown in **FIG.** 2 only shows a **single computer** 201 and a **single application server** 230, secure inter-process communications can be used to enhance security for **multiple application servers** 230 belonging to a single entity or multiple **users using computers** 201. Additionally, while **networks** 220 and 240 are separately designated in **FIG.** 2, the networks may be the same, such as an internet protocol (IP) network. Further, while a certification authority is discussed in the context of the present invention, a certification authority is not shown in the figures. In this regard, a certification authority may communicate over the **network** 220 and/or the **network** 240 to provide digital certificates to the **application server** 230 and the **management server** 250. Of course, the digital certificates may be provided by the certification authority as portable memory modules that are installed on the **application server** 230 or the **management server** 250. Furthermore, the entity that provides the **application**

'117 Claims	Exemplary Disclosures
	server 230 and/or the management server 250 may also act as a certification authority for its customers using the computers 102." Santamaria at [0005]: "Users of a mobile device (e.g., laptop, palmtop, portable media player, smartphone, multimedia mobile phone, mobile gaming systems such as a "Gameboy", etc.) may subscribe to one or more notification services. For example, users may subscribe to one or more "push" email services such as .Mac, Microsoft Exchange ActiveSync, push-IMAP, Yahoo! Push, etc. In the case of a push email service, for example, the email server may automatically transmit email messages and/or calendar updates to the email client on the mobile device for the user without the user requesting the item. In other words, the user does not request (poll) the email server for email messages or other items of interest. Thus, a push notification service is a persistent notification service. In addition, other types of services may use a similar push architecture (e.g., update/upgrade services, news services, weblog services, podcast services, etc.). In order to maintain a push notification service, the mobile device periodically refreshes the connection to the push notification service (e.g., by transmitting a ping message to the push server." Santamaria at [0071]: "FIG. 8 is a block diagram illustrating an exemplary data processing system which may be used in some embodiments of the invention. For example, the data processing system 800 may be a handheld computer, a personal digital assistant (PDA), a mobile telephone, a portable gaming system, a portable media player, or a handheld computer which may include a mobile telephone, a media player, and/or a gaming system. As another example, the data processing system 800 may be a network computer or an embedded processing device within another device." Isomursu at 2:1-18:

'117 Claims	Exemplary Disclosures
	"The present invention concerns a terminal for a communication network, the terminal capable of supporting a plurality of applications and having means of communicating user messages wherein it comprises means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header. Accordingly the terminal may readily have a plurality of different applications on such can be arranged into the terminal at a later stage. The later addition of applications can be done by direct contact of over the air contact to another device. One user message may contain data relating to one application indicated by the header, or a user message could contain data relating to several application, indicated by different headers, e.g. so that the header indicating a specific application is followed by the data relating to that specific application." Isomursu at 2:56-3:33: "The intention is that any user messages can be used, but in following mainly short messages will be referred to as an example. In addition to being fast to send, the advantages of a short message service can be utilised, such as not reserving the voice channel. Application related information can either be pre-stored in a terminal memory (permanent memory) or a user may store the application related information in a terminal memory (cache memory) by contacting a server by means of a terminal. Depending on the application, the user may enter user input or modify the information in the applications. In another application the information relating to an application may be sent by a service provider and the information may be such that it is not possible for the user to modify it, only to request the service provider to modify it. The information readily printed in the application can also be transmitted. An application type identifier or header is preferably added to the transmission, so that a receiving terminal identifies the short message as not an ordinary short message, but as a short message containing information relating to and intended for a specific application. The identifier can be a code in an address or a control field of

'117 Claims	Exemplary Disclosures
	the short message, or it can be a code in the message part of the short message. Because it has been realised that the short message service, already existing in the mobile station system, can be utilised for sending information on applications, the advantages are, e.g., that there is no need to establish an off-line server for sending the application related information, such as, for example, in the system presented in Publication WO 94/23394. Especially advantageous is that one and the same server, i.e. the SMS server (the Short Message Service Centre SM-SC) can be used for sending and forwarding information relating to any application, so there is no need to have separate servers for each application. The SMS server will forward any short message and the terminal will address the information to the correct application according to the header or identifier in the message. And since a short message can be sent simultaneously with a circuit-coupled connection, the sending of the application related information does not engage the terminal's communication line, e.g., in case of a simultaneously incoming call. A network like the GSM network is maintained by several operators and usually each operator has at least one SMS server of their own. In this case naturally any SMS server or several servers may be used for the invention." Isomursu at 3:54-51: "A terminal according to the present invention is wherein it comprises means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header. Another terminal according to the present invention is wherein it comprises means for sending data relating to one of said applications in a user message and means for adding a header to the user message, the header indicating the respective application that the data relates to. Correspondingly, a system according to the present invention is wherein the transmitting terminals comprise means for sending data relating to one of said applications in a user message and means for adding a header to the user message, the header indicating the respective application that the

'117 Claims	Exemplary Disclosures
	data relates to, and the receiving terminals comprise means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header." Isomursu at 5:33-51: "Additionally short messages may be sent by a personal computer PC. In this case the mobile services switching centre MSC has a connection to a server GTW (Gateway), which has a connection to e.g. Internet. Thus a computer PC having a connection to Internet (or directly to the server GTW as shown by the broken line) may fetch a WWW (World Wide Web) page from Internet, which physically can be found e.g. from the server GTW. Optionally a service provider or operator may have a separate server GTW having a connection to the mobile services switching centre MSC for sending short messages or other user messages. When using such a WWW page for sending a short message the user inputs the connection information (e.g. phone number) of the receiving terminal MS2 and the message to be sent, whereby the message is transferred via Internet and the server GTW to the mobile services switching centre MSC and further to the short message service centre SM-SC from which the message is directed to the receiving terminal MS2 via the mobile network." Isomursu at 6:28-59: "In the present invention, using short messages as an example, the transmission of each application related information will be identified by means of a specific code, i.e., an identifier, which enables the receiving terminal to process the received message directly into an application, as specified, containing the received data. The identifier is preferably implemented by using ASCII characters in an information field of the short message transmission frame, i.e., in a field INFO (**FIG. 5**), which contains the actual short message in characters. Alternatively the identifier may be in the form of some other character or string code, such as bits, since for sending a

'117 Claims	Exemplary Disclosures
	short message the data is anyway converted into bits. Because the information relating to the applications is transmitted in a short message, it can also be received by means of an ordinary mobile station, which does not support the application service, but is capable of transmitting and/or receiving short messages. By placing the application identifier in the field INFO, there is also the advantage that in an ordinary mobile station, which does not support this type of application service, but is capable of transmitting and/or receiving short messages, the application identifier is displayed to a user of the terminal, e.g. a communicator and, hence, the user notices not having received an ordinary short message, but the information relating to a specific application. In addition, a user of this type of ordinary mobile station can also transmit a message, such as mentioned above, by first writing, on the message, the identifier of the application in question in characters, and the rest of the information correctly divided. The reception of such a transmission by means of a terminal, according to the present invention, will produce a fully received application related information record." Isomursu at 7:6-51: "FIG. 6 illustrates an example of an application with an application record pre-stored in a terminal, the user input information (the application record) on which can be sent to another terminal as a short message. This application type is a so-called "Business Card". The application runs business card contents and contains the following information: name, position, company, contact information, etc. Each information can be in its own field or the application may only have one field, whereinto all the information is fed. FIG. 6 also illustrates the transmission of the information on an application as a short message. In this case, an identifier of the application type may be, e.g., 'BC' or 'Business Card' as illustrated in the figure. A terminal, according to the present invention, adds the application identifier (e.g. as letters or in other form) to the beginning of the information field of the short message to be transmitted first. The contents of the different fields ends automatically in a line feed character. On

'117 Claims	Exemplary Disclosures
	the basis of this character, a receiving terminal is capable of dividing the received information into the corresponding fields on the application. If this type of message was transmitted as a short message from an ordinary mobile station, a user would write, at the beginning of the message, an application identifier, i.e., in the case of FIG. 6, 'Business Card', after that a line feed character [cr], then the information on the application in succession or by field (depending on the application specification), i.e., first the information in the name field and a line feed character, etc. A received 'Business Card' can be stored in a memory of the terminal, where business cards can thus be stored in an electronic form and can be retrieved from the memory and looked at by means of the Business Card application. The information in a Business Card application may of course vary depending on the device. In some terminals it could mean e.g. only the name and phone number, which can be used as a Short Dial application. Accordingly the invention could be used to update contents of a Short Dial application. In this way the user could make a backup of the contents of the Short Dial application by sending the contents to a service center for storage. If the terminal (e.g. a mobile phone) is later lost/stolen or destroyed in which case the user will have to acquire a new terminal, he/she can in addition to activating the terminal, also download (retrieve) all Short Dial application contents to the terminal. Of course the same solution can be used for the contents of any application in the terminal. Several other applications will be explained here." Isomursu at 9:4-46: "By means of an application 'Service DTMF Commands', information can be received, e.g., from a network operator in order to utilise services provided by the operator. An application identifier can be 'Service DTMF Commands', and it may have fields for a sender's name, a DTMF command, and an explanation field. The command can be, e.g., a password, a user identifier, or something else related to the services provided by the operator. A user may use the received command, e.g., a password, when utilising the offered services, in which case, the user does not have

'117 Claims	Exemplary Disclosures
	to input the password through the keyboard, because the password can be obtained directly from the application (or the information in the application) in question. Other applications to which information may be received from a network operator are any applications which require some settings in the terminal before an application can be used. An example is an 'Internet Access Point' application which contains information necessary for the terminal to use a WWW browser for example. This information may be provider name, phone number, modem initialisation information, server information, IP address (Internet Protocol). Another example of an application that needs some settings is a Fax Box application. A faxbox is a service implemented e.g. in a server, which receives all faxes for a specific user. The user can then retrieve the faxes from the faxbox. For retrieving faxes the terminal needs to have the contact information to the server. This could be implemented so that when the terminal has received a new fax to the faxbox, the terminal receives a notification in a user message of a received fax, but additionally the user message would include a UID information (Unique Identifier), i.e. contact information to the faxbox server and other necessary information, such as filename of the fax and a password needed. The Fax Box application could function either manually, i.e. by user execution, or automatically so that the terminal contacts the faxbox server automatically for retrieving a received fax according to the identifier information contained in the notifying user message. The contact can be made as fax call to retrieve the fax. Optionally it could be a user message sending the identifier information back to the server and additionally the fax connection number of the terminal so that the server then can contact the terminal for sending the received fax to it via a fax call." Isomursu at 18:60-19:39: "Sending information of a service page in a short message according to the present invention saves considerably the power consumption of the terminal and, thus, prolongs its useful life, which is an important objective in battery-operated devices. In addition, savings are made in phone call expenses, when query information can be

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 26 of 442

'117 Claims	Exemplary Disclosures
	sent together with the short message. The whole system has been illustrated in more detail in FIG. 8. A terminal according to the present invention has been presented with reference numeral 1, by means of which a communications link (reference 101) to the Internet INT can be established. The invention could also be implemented by means of a device having means, according to the present invention, for processing information on an application into a short message, which is sent through an ordinary mobile station by coupling the device to the mobile station. Such a device could be, e.g., a computer. To simplify matters, only a communicator will be discussed here as an example of a terminal according to the invention. When the communications link is established to the Internet by means of a communicator 1, a service page of a provider of services can be downloaded (reference 102) from the Internet into its memory and user interface. By means of solutions known for prior art, the user, after having filled in the page, has sent the service page by means of the communicator 1 to a server SERV of the provider of services, i.e., along route 101–101′. This means that the communications link to the Internet should be open for transmission. In a system according to the present invention, the information on the service page is sent (reference 103) in a short message to a short message service centre SM-SC, which sends it further (reference 104) to the server SERV of the provider of services. The transmission of the service page through the Internet, according to prior art, lasts considerably longer than the transmission of a short message and, thus, due to the present invention, a shorter transmission time is achieved, thereby, effecting a saving in power, since, in the communicator, transmission and reception, in particular, consume a lot of power compared to other functions. In addition, in the solution according to prior art, a circuit-coupled connection is in use in transmission, in which case the communicator is engaged during transmission. On the other hand, the sending of a short message does not engage the circuit-coupled connection, and an additional advantage is that the short message service centre SM-SC will send the

'117 Claims	Exemplary Disclosures
	message to a receiver later, if the telephone number of the provider of services happens to be unreachable during the transmission of the message." Isomursu at 20:48-21:20: "In FIG. 9, there is a block diagram of the implementation of a terminal according to the present invention, which in the figure is shown as a device also having means for transmission over the air such as a mobile phone. However a similar implementation can be used for a terminal device of a service provider which usually does not have means for direct radio frequency transmission but is connected to such means (e.g. a base station) over the network as is shown in FIG. 2 (e.g. the personal computer PC or server GTW). The terminal is preferably a mobile phone or communicator, which has circuits and a user interface that enable the processing of different applications. The terminal 1 in FIG. 9 comprises, for communication using radio communication, a radio unit RU (the reference has not been marked in the figure), which comprises a transmitter branch 2, known from an ordinary mobile station, (comprising blocks implementing coding, interleaving, ciphering, modulating, and transmitting), a receiving branch 3 (comprising receiving, de-modulating, de-ciphering, de-interleaving, and implementing blocks) and, for transmission using radio communication, a duplex filter 4 that distinguishes between a received and transmitted message, as well as an antenna 5. The terminal has a main control circuit 6 that controls its operation. Furthermore, the main control circuit 6 comprises still a RU controller 7 that carries out control functions of an ordinary mobile station. In addition, the terminal main control circuit 6 comprises blocks 8–12 for carrying out the functions of a data processing unit of the terminal and for sending application related information as a short message according to the present invention. Thus, the blocks 8–12 can be said to form a data processing unit DU of the terminal. The controls of the radio unit RU and the terminal's data processing unit DU do not have to be integrated into the main control circuit but, instead, they could also be implemented apart from each

'117 Claims	Exemplary Disclosures
	other, so that the RU control circuit 7 is on the radio unit's side, and on the data processing unit's side, there is the DU processor 8, which is in connection with the RU control circuit 7 for establishing communication between the radio unit and the data processing unit." Okonnen at [0009]: "Aspect of the present invention may be found in an electronic device network. The network may comprise a plurality of servers and a plurality of an electronic devices communicatively coupled to at least one of the plurality of servers. The electronic devices may be adapted to employ at least one of a plurality of update agents resident in the electronic devices to update one of software and firmware in the electronic devices. The electronic devices are also adapted to provisioning the plurality of update agents with parameters and data used to facilitate update operations in the electronic device." Okonnen at [0011]: "In an embodiment according to the present invention, the network may further comprise at least one of an update server, and a plurality of generators. The generators may be adapted to generate updates able to be processed by at least one provisioned update agent in electronic device. The update server may be adapted to store updates accessible by the plurality of servers. The plurality of servers may be communicatively coupled to the update server." Okonnen at [0046]: "Electronic devices may be adapted to access servers to retrieve updates for updating at least one of firmware and software. An electronic device may be, for example, a mobile electronic device having software/firmware, such as, mobile cellular phone handsets, personal digital assistants (PDA's), pagers, MP3 players, digital cameras, etc. An update may comprise firmware and software updates that

'117 Claims	Exemplary Disclosures
	modify or change the version of a particular firmware or software installed in the electronic device, for example, upgrading to a newer version, repairing a bug in the firmware/software, etc. An update may also add new services to the electronic device or delete services, as desired by a service provider, device manufacturer, or an end-user." Okonnen at [0047]-[0048]: "FIG. 1 is a block diagram an electronic device network 105 according to an embodiment of the present invention. In the electronic device network illustrated in FIG. 1, an electronic device, for example mobile handset 107, may be communicatively coupled to at least one of a plurality of device servers, such as device server A 135 and device server B 137, for example. "The electronic device, for example, mobile handset 107, may be capable of employing one of several available update agents, for example, update agent A 113 and update agent B 115, resident in the electronic device to update software and/or firmware in the electronic device. The electronic device, for example, mobile handset 107, may also capable of provisioning the available update agents, such as for example, update agent A 113 and update agent B 115, with the necessary parameters and data needed to facilitate the respective update operations." *See also* Okonnen at Fig. 1. Fahrny at [0021]: "Headend: The control center of a cable television system, where broadcast signals are received and distributed. The headend generally contains antennas, preamplifiers, frequency converters, demodulators, encoders, compressors, automatic switching equipment and other related equipment that receives, amplifies, filters, encrypts, encodes, and converts incoming satellite and terrestrial streams for presentation to distribution channels."

'117 Claims	Exemplary Disclosures
	Fahrny at [0041]: "Referring to FIG. 1 a, a diagram illustrating a media decoder (i.e., controller, processor, apparatus, circuit, device, etc.) 100 of the present invention is shown. The decoder 100 may be implemented in connection with a digital media stream distribution system (described in more detail in connection with FIGS. 2(a-b)). The controller 100 is generally implemented as a security processor (or processing system) that provides at least one security feature (e.g., encryption, decryption, authentication, security key management, copy protection, digital rights management, etc.) to at least one digital media input/output stream. The decoder 100 generally has an input 102 that receives at least one signal (e.g., VIDIN and PID), an input/output 104 that receives/presents a signal (e.g., TFHE) as well as additional data, signals, messages, and the like, an input 106 that receives a working key modifier and application function signal (e.g., WKM), and an output 108 that presents a signal (e.g., VIDOUT)." Fahrny at [0052]: "Referring to FIG. 1 c, a diagram illustrating an alternative media decoder (i.e., controller, processor, apparatus, circuit, device, etc.) 100" of the present invention is shown. The decoder 100" may be advantageously implemented in connection with video on demand (VOD) key management. The media stream VIDIN may be an encrypted VOD media stream. The media stream VIDOUT may be clear VOD media stream. The input 102 may receive the media stream VIDIN. In one example, the input 102 may receive the identifier PID. In another example, the input 102 may receive a VOD session identifier (e.g., VODID). The decoder 100" generally does not receive the working key modifier WKM." Fahrny at [0058]-[0059]: "Referring to FIG. 2 a, a diagram illustrating a media stream processing and distribution system 200 implemented in connection with the present invention is

'117 Claims	Exemplary Disclosures
	shown. The distribution system 200 generally comprises a headend 202, a network 204, at least one set top box (STB) 206 (generally a plurality of STBs 206 a-206 n), and at least one respective receiving device (i.e., receiver, transceiver, display device, etc.) 208 (generally a plurality of devices 208 a-208 n). The distribution system 200 is generally implemented as a media service provider/subscriber system wherein the provider (or vendor) generally operates the headend 202 and the network 204, and also provides a subscriber (i.e., client, customer, service purchaser, user, etc.) with the STB 206. "The STB 206 is generally located at the subscriber location (not shown, e.g., home, tavern, hotel room, business, etc.) and the receiving device 208 is generally provided by the client. The device 208 is generally implemented as a television, high definition television (HDTV), monitor, host viewing device, MP3 player, audio receiver, radio, personal computer, media player, digital video recorder, game playing device, etc. The device 208 may be implemented as a transceiver having interactive capability in connection with the STB 206, the headend 202, or both the STB 206 and the headend 202." Fahrny at [0062]: "The network 204 is generally implemented as a media stream distribution network (e.g., cable, satellite, and the like) that is configured to selectively distribute (i.e., transmit and receive) media service provider streams (e.g., standard broadcast television and radio, digital television, HDTV, audio, MP3, text messaging, games, etc.) for example, as the stream VIDIN, the downloads TFHE, and the identifiers PID and VODID, to the STBs 206 and to the receivers 208, for example, as the stream VIDOUT. The stream VIDIN, the downloads TFHE, and the identifiers PID and VODID are generally distributed based upon (or in response to) subscriber information. For example, the level of service the client has purchased (e.g., basic service, premium movie channels, etc.), the type of service the client has requested (e.g., standard TV, HDTV, interactive messaging, video on demand, pay-per-view,

'117 Claims	Exemplary Disclosures
	impulse-pay-per-view, etc.), and the like may determine the media streams that are sent to (and received from) a particular subscriber." *See also* Fahrny at Figs. 1a, 2a. White at [0048]: "While the present invention may operate within a single (standalone) computer (e.g., system 100 of FIG. 1), the present invention is preferably embodied in a multi-user computer system, such as a client/server system. FIG. 2 illustrates the general structure of a client/server database system 200 suitable for implementing the present invention. As shown, the system 200 comprises one or more client(s) 210 connected to a server 230 via a network 220. Specifically, the client(s) 210 comprise one or more standalone terminals 211 connected to a database server system 240 using a conventional network. In an exemplary embodiment, the terminals 211 may themselves comprise a plurality of standalone workstations, dumb terminals, or the like, or comprise personal computers (PCs) such as the above-described system 100. Typically, such units would operate under a client operating system, such as a Microsoft® Windows client operating system (e.g., Microsoft® Windows 95/98, Windows 2000, or Windows XP)." Holbrook at [0083]: "FIG. 1 shows a block diagram of a messaging system apparatus, according to a preferred embodiment of the invention. In this embodiment, messaging system 10 comprises a central server 12, a plurality of messaging nodes 14, and a plurality of portable messaging units 16." *See also* Holbrook at Fig. 1. Cooley at Abstract:

'117 Claims	Exemplary Disclosures
	"Systems and methods are provided that allow an application residing on a user device to intercept messages directed to that device and to extract control information and/or data from the intercepted messages for use by the application. In one embodiment, the intercepted message can cause the application to change its configuration or to present information to the user that came from a sending website. In another embodiment, a message from a device can trigger a website to send back a desired piece of data. The returned data may contain a flag, which directs the data to a formatting server. The formatting server then converts the data into a format that allows the data to be intercepted by the desired application on the user device. This then allows the desired data to be presented to the user in the context of an application resident on the user's device." Cooley at ¶[0001]: "This disclosure relates to the control and customization of applications resident on a wireless device from locations remote therefrom and more specifically to systems and methods for allowing applications resident on a mobile device to be controlled using the text messaging capability of the wireless network." Cooley at ¶[0005]: "Systems and methods are provided that allow an application residing on a user device to intercept messages directed to that device and to extract control information and/or data from the intercepted messages for use by the application. In one embodiment, the intercepted message can cause the application to change its configuration or to present information to the user that came from a sending website. In another embodiment, a message from a device can trigger a website (such as an airline website) to send back a desired piece of data. The returned data can, for example, contain a flag, which directs the data to a formatting server. The formatting server then converts the data into a format that allows the data to be intercepted by the desired application on the user device. This then allows the

'117 Claims	Exemplary Disclosures
	desired data to be presented to the user in the context of an application resident on the user's device." Cooley at ¶[0013]: "FIG. 1 shows one embodiment of a system, such as system 10, in which the concepts of the invention can be practiced. In FIG. 1 there is shown a link from a user device, such as cellular telephone 15 to website 11. This link extends from the user device through a carrier network, such as network 14 and via service center 12 to the desired website. The message returned from website 11 contains an HTTP request against service center 12. The service center then queues a Short Message Service (SMS) message back to device 15 through an SMS broker, such as broker 13. Broker 13 then routes an SMS message through any carrier network, such as network 14, which need not be the same network that was used for the originating message from the device." Cooley at ¶[0015]: "When the incoming text message arrives at the user's device it is intercepted, under control of the device processor and an application, such as application 31 (as will be discussed), running on the device. This is possible because application 31 is registered with the device operating system when it was loaded on the device. Application 31 then routes the incoming message to the proper specific application in accordance with the data contained in the text message. Thus, in the situation with the flight reservation to Las Vegas (as discussed above), information pertaining to that flight is routed to a specific application, such as to air travel application 22-1 (shown in FIG. 2B), which application is dedicated to air travel data pertaining to the device user." Cooley at ¶[0017]:

'117 Claims	Exemplary Disclosures
	"When the user touches tile 23-1 all the flights that the user has booked on, for example, AMERICAN™ airlines (AA) are displayed. Touching tile 24-1 then yields the screen shown in FIG. 2E which then displays the information that has been received from an external source. In this case this information would have come typically from the AA server in the manner discussed above. There could be any amount of information and tiles (or layers) to this information, all controlled by application 30 residing on the user's device with the updated information coming, from time to time, via messages from various external locations. Note that in FIG. 2F the gate and status of the flight have been updated. This update could have come as a result of a request from the user or could have come as a text message from the airline as part of an update service for customers. In any event, the message would have gone first to service center 12 and then via SMS broker 13 to the user's device where application 30 would have taken the data and placed it in the proper tile." Cooley at ¶[0018]: "FIG. 3 shows one embodiment of a system for controlling messages on a user device. The incoming SMS message would normally be delivered to level 32 of network stack 30 for subsequent display to device user via, for example, display 16 (FIG. 1). The display would be in text or visual form and would then go into the SMS box in text, graphics or video form as is well-known. However, application 31 intercepts the message because that application is registered with the platform to receive certain SMS messages. This protocol varies from device to device based on operating system. SMS or HTTP transports can be used via the side channel process. SMS has the advantage of being a true "push" notification. Application 31 then routes the incoming message, based on the context and content of the data contained in the message to an individual tile, such as to

'117 Claims	Exemplary Disclosures
	tile 22-1 (FIG. 2B) or even more specifically to tile 24-1 (FIG. 2D), as discussed above." Cooley at ¶[0020]: "Another method for doing this would be to run a Simple Mail Transfer Protocol (SMTP) gateway at service center 12. This would allow application 31 to receive incoming messages in email form. Such email messages could then be parsed to see if they contain certain information, such as the tile ID, etc. This approach would benefit from a fixed format email message. The parsing includes filtering the email messages to see if they contain certain information. The filtered information then would be sent to the device by an SMS message. Second, the formatted SMS message could be sent to the service center for delivery to the particular user device." Cooley at ¶[0021]: "FIG. 4 shows one embodiment of a method, such as method 40, for controlling applications resident on a mobile device from a remote location. Process 401 determines if a message is incoming to a particular device. If there is a message available, then process 402 determines if the message contains any code or data that identifies the message as one that is to be delivered to a specific application resident on the device and/or delivered customized in a particular manner. If not, then the message is stored in its normal location, for example in SMS layer 32 (FIG. 3) and delivered without modification." Cooley at ¶[0022]: "If the message does contain a "diversion" code or other data that indicates that the message is to be delivered to another location or delivered in a customized manner, then process 404 delivers the incoming message (or other data) accordingly as discussed above, such that a particular user's tile is customized for a particular

'117 Claims	Exemplary Disclosures
	application. Customizing could, for example, include one or more of the following: appearance, context, display parameters, including audio, visual, application state change. In some situations, a control command is sent to a user's tile, for example, using an SMS transport protocol such that the command controls the display parameters and/or the display message instead of actual text being sent to the tile." Cooley at ¶[0024]: "FIG. 5 illustrates an example computer system 500 adapted according to one embodiment of the present invention. That is, computer system 500 comprises an example system on which embodiments of the present invention may be implemented (such as servers 11, 12, broker 13, and handheld device 15 of the example implementation of FIG. 1). Central processing unit (CPU) 501 is coupled to system bus 502. CPU 501 may be any general purpose or specialized purpose CPU. However, the present invention is not restricted by the architecture of CPU 501 as long as CPU 501 supports the inventive operations as described herein. CPU 501 may execute the various logical instructions according to embodiments of the present invention. For example, one or more CPUs, such as CPU 501, may execute machine-level instructions according to the exemplary operational flows described above in conjunction with FIG. 4." Cooley at ¶[0028]: While FIG. 5 shows a purpose computer, it should be noted that the exact configuration of a portion of a system according to various embodiments may be slightly different. For example, handheld devices according to one or more embodiments may be any kind of processor-based device, such as a cell phone, a Personal Digital Assistant, a specialized device, and/or the like. Additionally, servers (e.g., servers 11, 12 of FIG. 1) and the SMS broker according to one or more embodiments may be any kind of processor-based device capable of sending messages (e.g., email messages, SMS messages, MMS messages), such as a personal computer, a server-type computer, and the like. Moreover, embodiments of

'117 Claims	Exemplary Disclosures
	the present invention may be implemented on application specific integrated circuits (ASICs) or very large scale integrated (VLSI) circuits. In fact, persons of ordinary skill in the art may utilize any number of suitable structures capable of executing logical operations according to the embodiments of the present invention." Bobo at Abstract: "A Message Storage and Deliver System (MSDS) is connected to a plurality of DID phone lines and receives facsimile messages, voice messages, and data messages. The MSDS assigns a separate telephone number for each user of the system and can simultaneously receive more than one message for a single user. The messages are stored in memory and are also converted into appropriate hyper-text mark-up language (HTML) files. The MSDS is connected to the Internet and notifies the users with an E-mail message each time a message is received. The MSDS can also page the user so that the user receives almost instantaneous notice of the message. The users can then connect to the MSDS through the Internet and have the messages downloaded to their computers or the users can simply preview the messages stored in the MSDS. The users of the MSDS therefore have the advantage of being able to receive their messages at any time and at any location at a reasonable cost. The MSDS offers a number of options on how the messages may be sent to the user, such as several preview options available with facsimile messages. The user can also telephone the MSDS to listen to messages or to alter the service provided by the MSDS." Wakasa at ¶[0039]: "Referring to FIG. 1, the entire configuration of an information providing system 10 according to an embodiment is explained below. As shown in FIG. 1, the information providing system 10, for example, includes portable terminal devices 100 *a*, 100 *b* and 100 *c*, a server device 200, a network 300, readers/ writers 400 *a* and 400 *b* and a PC (personal computer) 500. In the following

'117 Claims	Exemplary Disclosures
	description, the portable terminal devices 100 *a*, 100 *b* and 100 *c* are each denoted by generic reference numeral 100 if it is not necessary to distinguish the portable terminal devices 100 *a*, 100 *b* and 100 *c* from each other. By the same token, the readers/ writers 400 *a* and 400 *b* are each denoted by generic reference numeral 400 if it is not necessary to distinguish the readers/ writers 400 *a* and 400 *b* from each other." Wakasa at ¶[0040]: "The server device 200 generates a transmitted message including parameter information and the identification information of an application to be executed by the portable terminal device 100 on the basis of the parameter information. Then, the server device 200 transmits the transmitted message to the portable terminal device 100 via the network 300. The parameter information is typically information used for requesting the application of the portable terminal device 100 to display information desired by the user or carry out predetermined processing. For example, the parameter information is information requesting the application of the portable terminal device 100 to display "Hello" on the display screen. The portable terminal device 100 executes the application corresponding to the identification information of the application included in transmitted message by using the parameter information as input data. For example, by using the parameter information requesting the application of the portable terminal device 100 to display "Hello" on the display screen as input data, the portable terminal device 100 executes the application corresponding to the identification information, displaying "Hello" on the display screen of the portable terminal device 100. As an alternative, the server device 200 holds an application to be provided to the portable terminal device 100 and, in accordance with a request made by the portable terminal device 100, the server device 200 transmits the application to the portable terminal device 100 via the network 300." Satake at ¶[0136]:

'117 Claims	Exemplary Disclosures
	"The system shown in FIG. 1 is formed by a stock price information server 1, an SMS server 6 that transmits an SMS message from the stock price information server 1 to a user, a user terminal 7 serving as a mobile communication terminal belonging to the user, and a mobile exchange system 8 that relays communication between the stock price information server 1, the SMS server 6, and the user terminal 7. The mobile exchange system 8 and the user terminal 7 are connected by two transmission paths, namely, a traffic channel that is used for data communication and a paging channel that is used for SMS (i.e., PUSH messages) communication." *See also* Satake at Fig. 1. Satake at ¶[0145]: "If neither a UNTL header nor a WHLE header are recorded, then the application type list shown in FIG. 10 is checked (S15). In the application type list, an effective period is recorded for each type of application. Types of application include, for example,. stock price services, weather forecast services, news services, alarm services, games and the like. If the effective period is recorded, a value obtained by adding this effective period to the stored time of the messages to the server that is recorded in the header of the SMS messages is calculated as the time of the term of validity (S16). If the effective period is not recorded, then either termless, or an impractical term of validity such as, for example, Dec. 31, 2099 is set (S17)." *See also* Satake at ¶[0147]. *See generally* Dorenbos (discussing encryption). *See also* Kasai at ¶¶[0023]-[0051], Alao at Fig. 2, 7:29-14:10, Wills at ¶¶[0024]-[0059]

'117 Claims	Exemplary Disclosures
[1a] a plurality of device messaging agents, each executable on a respective one of a plurality of mobile end-user devices configured to exchange Internet data via a data connection to a wireless network; and	The references below disclose and/or render obvious this limitation. For example, see the following passages and/or figures, as well as related disclosures: Huskins at 2:19-28: "In one embodiment of the present invention, the server 10 may communicate with the clients 12 over a transport 14. The transport 14 may be in accordance with an analog or digital broadcasting system. As one example, the transport 14 may be compliant with the Digital Video Broadcast (DVB); Network-independent Protocol, ETS 300802, dated November 1997 and available from the European Telecommunications Standards Institute (ETS), Valbonne, France. The transport 14 may be a satellite, cable or airwave broadcasting system as examples." Huskins at 3:4-11: "In addition, the client 12 may include one or more addressable agents 44, 45, 46 and 48 that may be independently addressed by remote units such as the server 10. Moreover, by providing addressable agents 44, 45, 46 and 48 within a given client 12, messages that are specialized or which need specialized handling may be addressed to particular agents resident on the client 12 for appropriate handling." Huskins at 4:20-33: "In accordance with one embodiment of the present invention, the server 10 may implement a unidirectional messaging system. In a unidirectional messaging system, the server 10 may transmit messages to a plurality of clients that are unable to respond in any way. One example of such a network is a direct-to-home (DTH) broadcast network that may be compliant with the DVB protocol. The network may use a connection oriented communication protocol or a real time connectionless communication protocol as two examples. There are many applications of unidirectional messaging from server to client such as instant messaging, command and control and notification and signaling, as examples. In other cases, the network

'117 Claims	Exemplary Disclosures
	may be a bidirectional network, for example with an Internet Protocol (IP) multicast backbone." Huskins at 3:12-19: "The server 10 may include a network management command and control server 16. The server 16 is responsible for managing the collection of data from clients 12. The server 16 transmits messages which include a distinct service identifier (e.g., service_id=0x02) . The server 10 may also include an instant messaging or short message service (SMS) server 18 that also transmits messages having a distinct service identifier (e.g., service_id=0x04)." Huskins at 7:19-26: "However, if the particular client 12 is the intended recipient, the server 38 checks the message's service identifier and passes the message to the correct agent 44, 46 or 48, as indicated in block 64. The message is then delivered to the appropriate agent 44 as indicated in block 66. In the agent 44, the information is parsed and passed to an appropriate process for handling as indicated in block 68." Huskins at 5:26-33: "The message from the DVB MPE 26 is provided to a DVB MPE receiver 42. The receiver 42 communicates with an IP multicast module 40 and a unidirectional messaging server 38. The server 38 breaks down the message to determine whether a service identifier was included in the data stream. If so, the message is forwarded to an appropriate agent designated to receive messages with particular service identifiers . A disk management agent 44 may be coupled to a disk drive controller 47 in turn coupled to a storage device 43 that may, for example, be a hard disk drive. The disk management agent 44 may be addressed as a message recipient by the client disk management server 20. The disk management agent 44 may cause processes to be undertaken through the controller 47 that in turn control the use of the storage

'117 Claims	Exemplary Disclosures
	device 43. This control may include determining what information is stored on the storage device 43 and how that information is stored on the storage device 43." Huskins at 6:11-21: "In one embodiment of the present invention, the network management command and control server 16 may provide a specific message identifier that causes its message to be received by a network management session management agent 48 tuned to a particular service identifier. Similarly, messages from the instant messaging server 18 may include a service identifier that cause those messages to be forwarded to the agent 48 in the client 12. Likewise, messages from the data management session control server 21 may have appropriate identifiers that cause them to be shunted to the data management session control agent 45 on the client 12." *See also* Huskins at Fig. 1. Chou at 6:3-14: "One important aspect of WAP is the specification of a push architecture that allows content to be sent or "pushed" by server-based applications to WAP-compliant mobile client devices. Push functionality is particularly relevant to real-time applications that send notifications to their users, such as messaging, stock quotes and traffic update alerts. Without push functionality, these types of applications would require the devices to poll an application server for new information or status. In a wireless environment, such polling activities would constitute inefficient and wasteful use of wireless network resources. Moreover, entering URIs on mobile devices is time consuming, difficult and clumsy." Chou at 10:6-21:

'117 Claims	Exemplary Disclosures
	"Enhanced security may be provided in the illustrative embodiment through the use of security stamps, such as time stamps and device stamps. For example, a time stamp may be used to register the particular time of a given user request, while a device stamp may be used to register uniquely which device is allowed to retrieve the associated content. The content with these security stamps may be stored in association with a dynamically-generated URI on the WSS. In order to stop potential eavesdroppers to key entry at the mobile client device, the exact URI may be encoded in the SI of the WAP push content, which is invisible to the user and any third party. In addition, the service notification and encoded procedures for accessing the content on the WSS are delivered to a uniquely-identified mobile client device through WAP push content based on the registered device stamp." Rakic at [0059]: "FIG. 4 is a functional block diagram of client device 102. As shown, client device 102 may include WAP logic 402 and a secure push message client 404. Although not illustrated in FIG. 4, client device 102 may include additional functional components, such as the components that are shown in FIG. 3, an operating system (e.g., Symbian OS, Palm OS, Windows Mobile OS, Blackberry OS, etc.), an application (e.g., an instant messenger client, an email client, etc.), logic for wireless or wire-line communication protocols other than the WAP, etc." Shenfield at [0024]: "Service provider 120 communicates over wireless network 130 with a push client 140 that is located on a mobile device. Push client 140 will be described in more detail below. Push client 140 receives the content that is being delivered from content provider 110 and can communicate the content with a client application 150, which ultimately consumes the content." Shenfield at [0027]:

'117 Claims	Exemplary Disclosures
	"Specifically, a first alternative architecture 210 includes multiple content providers 110 communicating with a push proxy 122. Push proxy 122, as in the architecture of Figure 1, communicates over wireless network 130 with a push client 140. Further, multiple client applications 150 exist in architecture 210. This is therefore an N-1-1-N system having multiple content providers 110 and multiple client applications 150." *See generally* Hwang, [0001], [0035], [0097], [0100]-[0104], [0110]-[0115]; *see also* Hwang Fig. 8. Stage at 5:19-55: Typically a user controls the browser 70 through the input devices 20 and 24. A desired resource is identified using a URL. The browser controls the transceiver 3 to request access to the resource. The terminal 2 thus communicates with the protocol gateway 62 which in turn communicates with the web server 52. The WAP content returned to the transceiver 3 in response to the request is supplied to the browser 70. The final destination of the received content will however depend upon the content type of the received content. The content type is an indicator which is dependent upon the type of application for which it is intended. For example a browser application may be able to use, display or interpret a variety of content types but be unable to use a variety of other different content types. Likewise the email application 78 and the ringtone application 76 will be designed to receive content of a specific types. The routing table 72 registers applications which are usable by the terminal. On registering an application, the routing table associates with the address of the application, the content type or types that the application can process. A typical routing table 72 is illustrated in FIG. 5. The browser application 70 identifies the content type of the received content and accesses the routing table 72. The routing table returns the address of the registered application associated with the received content type and the browser transfers the

'117 Claims	Exemplary Disclosures
	received content to that address. If there is no registered application associated with the received content type, a null address is returned to the browser which sends an error message to the display 14. In this way the terminal 2 has the browser functionality and in addition the functionality of additional applications. Whereas the applications 76 and 78 are integrated within the terminal 2, the application associated with the accessory device, in this case a camera application is not integrated within the terminal 2 but is remote. The accessory device may be replaced to change and/or enhance the functionality of the terminal, or it may be removed. Stage at Fig. 4:

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 47 of 442

'117 Claims	Exemplary Disclosures

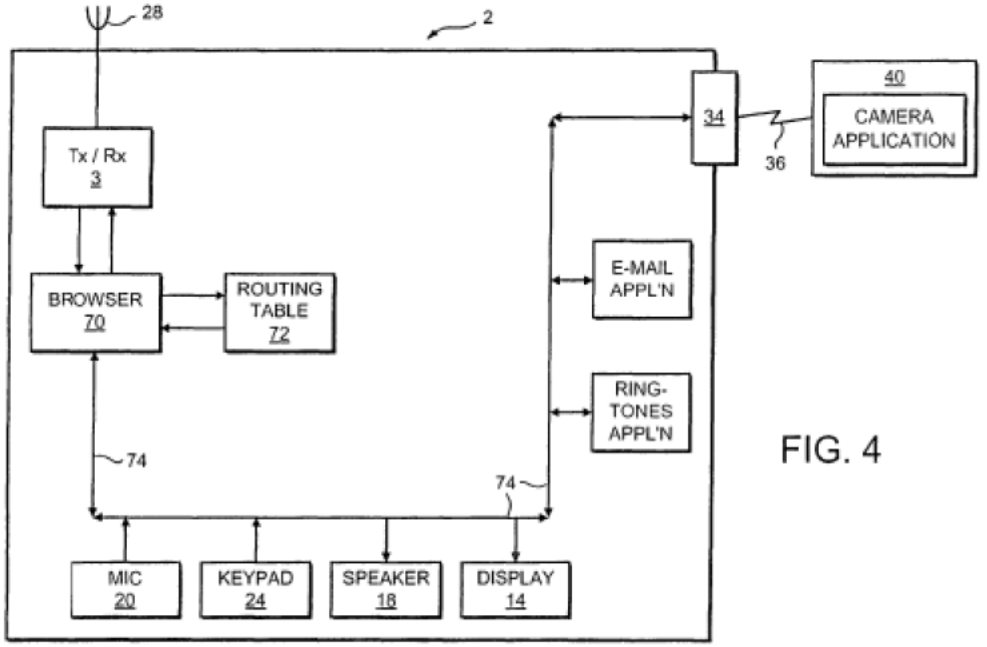


FIG. 4

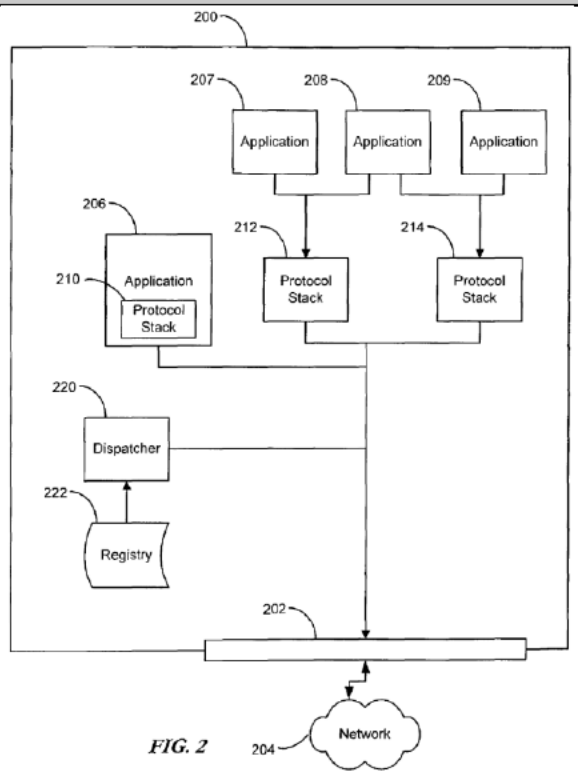
Lonnfors at [0030]:

> The present invention is directed towards using a dispatching procedure to identify and deliver incoming data to the correct client application of a device. The dispatcher may examine data arriving on one or more network ports and within one or more protocols and determine the appropriate application for the data. In another arrangement, the dispatcher may act as a lookup service for other server applications to determine the correct disposition of incoming data.

Lonnfors at [0044]:

'117 Claims	Exemplary Disclosures
	When a dispatcher 220 is used to analyze incoming data, the dispatcher 220 may look at protocol headers and other parts of the message that relate to entries in the registry 222. For protocols such as SIP, HTTP, and SMTP, the protocols may use MIME headers and other content descriptive headers. This data can be embedded in the content of the message, such as by using XML tags to encapsulate the data. Where the application receives incoming messages from multiple protocol stacks, the dispatcher may utilize the application descriptor data in the messages that include the protocol bearer (one or many), ports, content-type and the other indicators of bearer logic. These indicators of bearer logic can contain additional information such as application ID (or some similar ID) and parameters included in some headers or URI of the selected protocol bearer. Lonnfors at [0053]: By using the "application-type" identifier, the PushRegistry framework could handle multiple applications that may use the same incoming port and protocol. When receiving an incoming message, the AMS could read attribute descriptors of the incoming message (whether within the body of the message, within PushRegistry specific data portions, or elsewhere) and send the message to the correct MIDlet based on those descriptors. Lonnfors at Fig. 2:

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 49 of 442

'117 Claims	Exemplary Disclosures
	 *FIG. 2* Lonnfors at Fig. 4:

'117 Claims	Exemplary Disclosures

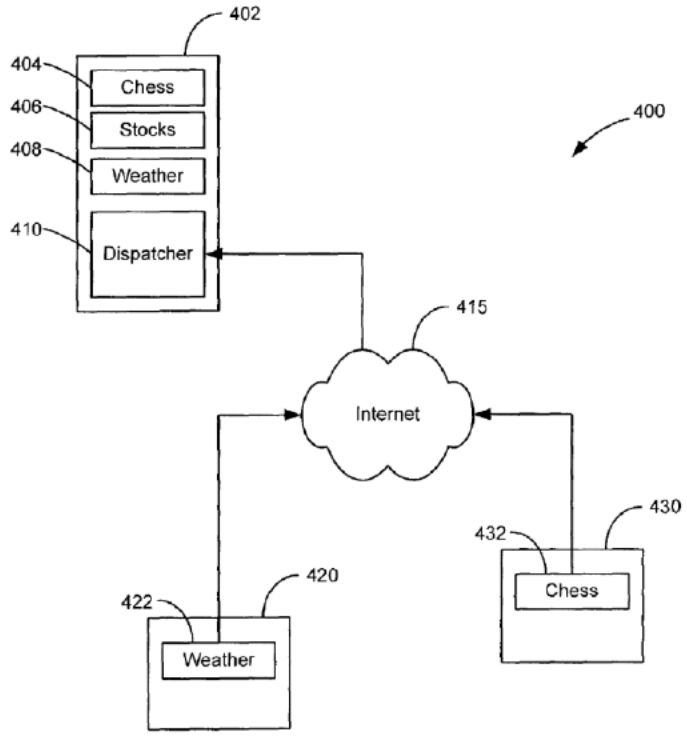


FIG. 4

Novack at [0035]:

> "In one example, the first process is running on an application server that provides application services over the internet to requesting clients. The application server interacts with a management server that manages multiple databases as a back-end service. The first process may be run when a user contacts an application server and

'117 Claims	Exemplary Disclosures
	downloads presentation layer elements of an application to interact with the application server. The application includes instructions to invoke additional processes for predetermined user requests. Accordingly, the application will request the services of a second process at S110." Novack at [0044]: "FIG. 2 shows an exemplary network architecture for secure inter-process communications. As shown, a **computer** 201 communicates through a **network** 220 with an **application server** 230. As an example, the **application server** 230 may be a web server that provides internet applications for users. The user downloads a web page onto the **computer** 201 when the user contacts an internet address corresponding to the desired internet application. Typically, the web page is accessed and downloaded using hyper text transport protocol (HTTP)." Novack at [0046]: "The **management server** 250 manages **databases** 254 and 256. The **databases** 254 and 256 store data associated with the service provided by the **application server** 230. As an example, the **application server** 230 may provide internet services for accounts of a service provider. Thus, the **application server** 230 may provide services for a bank, a utility, or a content provider such as a news service. While the user of the **computer** 201 is intentionally provided with direct access to the **application server** 230, the configuration of the network shown in FIG. 2 is intended to help deny the user of the **computer** 201 direct access to the **management server** 250 and the **databases** 254, 256. Further, to ensure that the communications between the **application server** 230 and the **management server** 250 are secure, a digital certificate is installed on the **application server** 230, and the digital certificate is bound to the **application server** 230 and/or a process running on the **application server** 230."

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 52 of 442

'117 Claims	Exemplary Disclosures
	Novack at [0064]: "The device in FIG. 4 may include multiple processes that communicate with each other using the secure inter-process communications of the present invention. Additionally, although the terms "client" and "server" are used to refer to devices, the devices may be defined and distinguished by a set of software instructions installed as a pre-packaged program. Further, while examples of the uses of the present invention include secure client/server networks and peer-to-peer networks, secure inter-process communications can be used to enhance security in any inter-process communications environment. Further, while the embodiment shown in FIG. 2 only shows a single computer 201 and a single application server 230, secure inter-process communications can be used to enhance security for multiple application servers 230 belonging to a single entity or multiple users using computers 201. Additionally, while networks 220 and 240 are separately designated in FIG. 2, the networks may be the same, such as an internet protocol (IP) network. Further, while a certification authority is discussed in the context of the present invention, a certification authority is not shown in the figures. In this regard, a certification authority may communicate over the network 220 and/or the network 240 to provide digital certificates to the application server 230 and the management server 250. Of course, the digital certificates may be provided by the certification authority as portable memory modules that are installed on the application server 230 or the management server 250. Furthermore, the entity that provides the application server 230 and/or the management server 250 may also act as a certification authority for its customers using the computers 102." Santamaria at [0005]-[0006]: "Users of a mobile device (e.g., laptop, palmtop, portable media player, smartphone, multimedia mobile phone, mobile gaming systems such as a "Gameboy", etc.) may subscribe to one or more notification services. For example, users may subscribe to

'117 Claims	Exemplary Disclosures
	one or more "push" email services such as .Mac, Microsoft Exchange ActiveSync, push-IMAP, Yahoo! Push, etc. In the case of a push email service, for example, the email server may automatically transmit email messages and/or calendar updates to the email client on the mobile device for the user without the user requesting the item. In other words, the user does not request (poll) the email server for email messages or other items of interest. Thus, a push notification service is a persistent notification service. In addition, other types of services may use a similar push architecture (e.g., update/upgrade services, news services, weblog services, podcast services, etc.). In order to maintain a push notification service, the mobile device periodically refreshes the connection to the push notification service (e.g., by transmitting a ping message to the push server). "In addition, users may subscribe to one or more "pull" services, such as "pull" email services (e.g., IMAP, POP3). In a pull email service, a user periodically checks (polls) the email server to determine if there is new email messages. If there is new email messages, they are then downloaded to the client. Many email clients support an automatic configuration of a poll interval. For example, a user of the mobile device may configure a poll interval of 10 minutes for a POP3 email account (thus the email client automatically polls the email server every 10 minutes to check for new email messages). In addition, notification services may be configured to provide network wide (e.g., Internet wide) event notification messages to multiple subscribers, where notification services may be automatically discovered by one or more subscribers and/or publishers, which is described in U.S. patent application Ser. No. 12/042,307, entitled "Automatic Notification System and Process", filed on Mar. 4, 2008, which is hereby incorporated by reference in its entirety." Isomursu at 2:1-18: "The present invention concerns a terminal for a communication network, the terminal capable of supporting a plurality of applications and having means of

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 54 of 442

'117 Claims	Exemplary Disclosures
	communicating user messages wherein it comprises means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header. Accordingly the terminal may readily have a plurality of different applications on such can be arranged into the terminal at a later stage. The later addition of applications can be done by direct contact of over the air contact to another device. One user message may contain data relating to one application indicated by the header, or a user message could contain data relating to several application, indicated by different headers, e.g. so that the header indicating a specific application is followed by the data relating to that specific application." Isomursu at 3:54-51: "A terminal according to the present invention is wherein it comprises means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header. Another terminal according to the present invention is wherein it comprises means for sending data relating to one of said applications in a user message and means for adding a header to the user message, the header indicating the respective application that the data relates to. Correspondingly, a system according to the present invention is wherein the transmitting terminals comprise means for sending data relating to one of said applications in a user message and means for adding a header to the user message, the header indicating the respective application that the data relates to, and the receiving terminals comprise means for receiving user messages having data and a header relating to one of said applications and means for addressing the data to a respective application according to said header." Isomursu at 5:33-51: "Additionally short messages may be sent by a personal computer PC. In this case the mobile services switching centre MSC has a connection to a server GTW

'117 Claims	Exemplary Disclosures
	(Gateway), which has a connection to e.g. Internet. Thus a computer PC having a connection to Internet (or directly to the server GTW as shown by the broken line) may fetch a WWW (World Wide Web) page from Internet, which physically can be found e.g. from the server GTW. Optionally a service provider or operator may have a separate server GTW having a connection to the mobile services switching centre MSC for sending short messages or other user messages. When using such a WWW page for sending a short message the user inputs the connection information (e.g. phone number) of the receiving terminal MS2 and the message to be sent, whereby the message is transferred via Internet and the server GTW to the mobile services switching centre MSC and further to the short message service centre SM-SC from which the message is directed to the receiving terminal MS2 via the mobile network." Isomursu at 6:28-59: "In the present invention, using short messages as an example, the transmission of each application related information will be identified by means of a specific code, i.e., an identifier, which enables the receiving terminal to process the received message directly into an application, as specified, containing the received data. The identifier is preferably implemented by using ASCII characters in an information field of the short message transmission frame, i.e., in a field INFO (FIG. 5), which contains the actual short message in characters. Alternatively the identifier may be in the form of some other character or string code, such as bits, since for sending a short message the data is anyway converted into bits. Because the information relating to the applications is transmitted in a short message, it can also be received by means of an ordinary mobile station, which does not support the application service, but is capable of transmitting and/or receiving short messages. By placing the application identifier in the field INFO, there is also the advantage that in an ordinary mobile station, which does not support this type of application service, but is capable of transmitting and/or receiving short messages, the application identifier

'117 Claims	Exemplary Disclosures
	is displayed to a user of the terminal, e.g. a communicator and, hence, the user notices not having received an ordinary short message, but the information relating to a specific application. In addition, a user of this type of ordinary mobile station can also transmit a message, such as mentioned above, by first writing, on the message, the identifier of the application in question in characters, and the rest of the information correctly divided. The reception of such a transmission by means of a terminal, according to the present invention, will produce a fully received application related information record." Isomursu at 7:6-51: "FIG. 6 illustrates an example of an application with an application record pre-stored in a terminal, the user input information (the application record) on which can be sent to another terminal as a short message. This application type is a so-called "Business Card". The application runs business card contents and contains the following information: name, position, company, contact information, etc. Each information can be in its own field or the application may only have one field, whereinto all the information is fed. FIG. 6 also illustrates the transmission of the information on an application as a short message. In this case, an identifier of the application type may be, e.g., 'BC' or 'Business Card' as illustrated in the figure. A terminal, according to the present invention, adds the application identifier (e.g. as letters or in other form) to the beginning of the information field of the short message to be transmitted first. The contents of the different fields ends automatically in a line feed character. On the basis of this character, a receiving terminal is capable of dividing the received information into the corresponding fields on the application. If this type of message was transmitted as a short message from an ordinary mobile station, a user would write, at the beginning of the message, an application identifier, i.e., in the case of FIG. 6, 'Business Card', after that a line feed character [cr], then the information on the application in succession or by field (depending on the application specification), i.e., first the information in the name field and a line feed character,

'117 Claims	Exemplary Disclosures
	etc. A received 'Business Card' can be stored in a memory of the terminal, where business cards can thus be stored in an electronic form and can be retrieved from the memory and looked at by means of the Business Card application. The information in a Business Card application may of course vary depending on the device. In some terminals it could mean e.g. only the name and phone number, which can be used as a Short Dial application. Accordingly the invention could be used to update contents of a Short Dial application. In this way the user could make a backup of the contents of the Short Dial application by sending the contents to a service center for storage. If the terminal (e.g. a mobile phone) is later lost/stolen or destroyed in which case the user will have to acquire a new terminal, he/she can in addition to activating the terminal, also download (retrieve) all Short Dial application contents to the terminal. Of course the same solution can be used for the contents of any application in the terminal. Several other applications will be explained here." Isomursu at 9:4-46: "By means of an application 'Service DTMF Commands', information can be received, e.g., from a network operator in order to utilise services provided by the operator. An application identifier can be 'Service DTMF Commands', and it may have fields for a sender's name, a DTMF command, and an explanation field. The command can be, e.g., a password, a user identifier, or something else related to the services provided by the operator. A user may use the received command, e.g., a password, when utilising the offered services, in which case, the user does not have to input the password through the keyboard, because the password can be obtained directly from the application (or the information in the application) in question. Other applications to which information may be received from a network operator are any applications which require some settings in the terminal before an application can be used. An example is an 'Internet Access Point' application which contains information necessary for the terminal to use a WWW browser for example. This information may be provider name, phone number, modem initialisation information,

'117 Claims	Exemplary Disclosures
	server information, IP address (Internet Protocol). Another example of an application that needs some settings is a Fax Box application. A faxbox is a service implemented e.g. in a server, which receives all faxes for a specific user. The user can then retrieve the faxes from the faxbox. For retrieving faxes the terminal needs to have the contact information to the server. This could be implemented so that when the terminal has received a new fax to the faxbox, the terminal receives a notification in a user message of a received fax, but additionally the user message would include a UID information (Unique Identifier), i.e. contact information to the faxbox server and other necessary information, such as filename of the fax and a password needed. The Fax Box application could function either manually, i.e. by user execution, or automatically so that the terminal contacts the faxbox server automatically for retrieving a received fax according to the identifier information contained in the notifying user message. The contact can be made as fax call to retrieve the fax. Optionally it could be a user message sending the identifier information back to the server and additionally the fax connection number of the terminal so that the server then can contact the terminal for sending the received fax to it via a fax call." Isomursu at 18:60-19:39: "Sending information of a service page in a short message according to the present invention saves considerably the power consumption of the terminal and, thus, prolongs its useful life, which is an important objective in battery-operated devices. In addition, savings are made in phone call expenses, when query information can be sent together with the short message. The whole system has been illustrated in more detail in FIG. 8. A terminal according to the present invention has been presented with reference numeral 1, by means of which a communications link (reference 101) to the Internet INT can be established. The invention could also be implemented by means of a device having means, according to the present invention, for processing information on an application into a short message, which is sent through an ordinary mobile station by coupling the device to the mobile station. Such a device could be,

'117 Claims	Exemplary Disclosures
	e.g., a computer. To simplify matters, only a communicator will be discussed here as an example of a terminal according to the invention. When the communications link is established to the Internet by means of a **communicator** 1, a service page of a provider of services can be downloaded (reference 102) from the Internet into its memory and user interface. By means of solutions known for prior art, the user, after having filled in the page, has sent the service page by means of the **communicator** 1 to a server SERV of the provider of services, i.e., along **route** 101–101′. This means that the communications link to the Internet should be open for transmission. In a system according to the present invention, the information on the service page is sent (reference 103) in a short message to a short message service centre SM-SC, which sends it further (reference 104) to the server SERV of the provider of services. The transmission of the service page through the Internet, according to prior art, lasts considerably longer than the transmission of a short message and, thus, due to the present invention, a shorter transmission time is achieved, thereby, effecting a saving in power, since, in the communicator, transmission and reception, in particular, consume a lot of power compared to other functions. In addition, in the solution according to prior art, a circuit-coupled connection is in use in transmission, in which case the communicator is engaged during transmission. On the other hand, the sending of a short message does not engage the circuit-coupled connection, and an additional advantage is that the short message service centre SM-SC will send the message to a receiver later, if the telephone number of the provider of services happens to be unreachable during the transmission of the message." Isomursu at 20:48-21:20: "In FIG. 9, there is a block diagram of the implementation of a terminal according to the present invention, which in the figure is shown as a device also having means for transmission over the air such as a mobile phone. However a similar implementation can be used for a terminal device of a service provider which usually does not have means for direct radio frequency transmission but is connected to such means (e.g. a

'117 Claims	Exemplary Disclosures
	base station) over the network as is shown in FIG. 2 (e.g. the personal computer PC or server GTW). The terminal is preferably a mobile phone or communicator, which has circuits and a user interface that enable the processing of different applications. The terminal 1 in FIG. 9 comprises, for communication using radio communication, a radio unit RU (the reference has not been marked in the figure), which comprises a transmitter branch 2, known from an ordinary mobile station, (comprising blocks implementing coding, interleaving, ciphering, modulating, and transmitting), a receiving branch 3 (comprising receiving, de-modulating, de-ciphering, de-interleaving, and implementing blocks) and, for transmission using radio communication, a duplex filter 4 that distinguishes between a received and transmitted message, as well as an antenna 5. The terminal has a main control circuit 6 that controls its operation. Furthermore, the main control circuit 6 comprises still a RU controller 7 that carries out control functions of an ordinary mobile station. In addition, the terminal main control circuit 6 comprises blocks 8–12 for carrying out the functions of a data processing unit of the terminal and for sending application related information as a short message according to the present invention. Thus, the blocks 8–12 can be said to form a data processing unit DU of the terminal. The controls of the radio unit RU and the terminal's data processing unit DU do not have to be integrated into the main control circuit but, instead, they could also be implemented apart from each other, so that the RU control circuit 7 is on the radio unit's side, and on the data processing unit's side, there is the DU processor 8, which is in connection with the RU control circuit 7 for establishing communication between the radio unit and the data processing unit." Okonnen at [0009]: "Aspect of the present invention may be found in an electronic device network. The network may comprise a plurality of servers and a plurality of an electronic devices communicatively coupled to at least one of the plurality of servers. The electronic

'117 Claims	Exemplary Disclosures
	devices may be adapted to employ at least one of a plurality of update agents resident in the electronic devices to update one of software and firmware in the electronic devices. The electronic devices are also adapted to provisioning the plurality of update agents with parameters and data used to facilitate update operations in the electronic device." Okonnen at [0020]: "In an embodiment according to the present invention, the network may further comprise update agent provisioning information stored in the electronic device. The update agent provisioning information may comprise a device server URL, an index of provisioned update agents, a security key, and electronic device related information. The device server URL may provide references to servers hosting updates to be downloaded. The updates may be compatible with update agents currently available and provisioned in the electronic device." Okonnen at [0023]: "In an embodiment according to the present invention, the electronic device may further comprise an update agent table resident in non-volatile memory. The update agent table may contain references to a plurality of update agents currently available and provisioned in the electronic device. The update agent table may map update agent names, update agent address locations, and types of updates that the update agents are adapted to process, and provisioning status of the update agents for all available update agents in the electronic device." Okonnen at [0047]-[0048]: "FIG. 1 is a block diagram an electronic device network 105 according to an embodiment of the present invention. In the electronic device network illustrated in FIG. 1, an electronic device, for example mobile handset 107, may be

'117 Claims	Exemplary Disclosures
	communicatively coupled to at least one of a plurality of device servers, such as device server A 135 and device server B 137, for example. "The electronic device, for example, mobile handset 107, may be capable of employing one of several available update agents, for example, update agent A 113 and update agent B 115, resident in the electronic device to update software and/or firmware in the electronic device. The electronic device, for example, mobile handset 107, may also capable of provisioning the available update agents, such as for example, update agent A 113 and update agent B 115, with the necessary parameters and data needed to facilitate the respective update operations." Okonnen at [0061]: "Aspects of the present invention may be found in electronic devices adapted to support multiple update agents. Each update agent may have different capabilities. Some update agents may be employed as backup update agents in a situation where a primary update agent may be faulty, inappropriate, outdated, etc." Okonnen at [0069]: "FIG. 2B is a block diagram 225 illustrating an update agent table 231 stored in non-volatile memory 111 of an electronic device, for example, mobile handset 107, according to an embodiment of the present invention. The update agent table 231 in the mobile handset 107 may provide references, such as location addresses and other related information, including information regarding update agent provisioning, to the update agents currently available in the mobile handset 107." Okonnen at [0070]-[0071]: "The update agent table 231 may maintain references to a plurality of available updates and associated update agents. When a new update agent is added, such as via an update of firmware/software in the mobile handset 107, an additional update

'117 Claims	Exemplary Disclosures
	may also be inserted into the mobile handset 107. The additional update may invoke insertion of a corresponding entry in the update agent table 231 and population thereof with associated relevant information, such as the address location of the new update agent, (i.e., where the new update agent is stored in non-volatile memory 111). An update package may be downloaded and installed in the mobile handset 107 and may result in removal of an existing update agent or replacement thereof by another. "In an embodiment according to the present invention, the update agent table 231 may comprise an array of addresses for all available update agents in the mobile handset." Okonnen at [0072]-[0073]: "FIG. 2C is a block diagram 255 illustrating an update agent table 261 located in non-volatile memory 111 of an electronic device, for example, mobile handset 107, according to an embodiment of the present invention. The update agent table in the mobile handset 107 may contain references to a plurality of update agents currently available and provisioned in the mobile handset. "The update agent table 261 may maintain, for example, a mapping between update agent names, the update agent corresponding address locations in non-volatile memory, and the type of updates that the update agents are adapted to process, for all available update agents in the mobile handset 107. The update agent table 261 may also contain optional information regarding provisioning status of the update agents, such as a provisioning flag indicating whether the associated update agent has been provisioned or not." Okonnen at [0075]-[0076]: "The type information may be provided as 'file extensions' in an embodiment according to the present invention. For example, the file extension 'dup' may

'117 Claims	Exemplary Disclosures
	correspond to updates generated to update generators, such as generator A 139 and generator B 141, as illustrated in FIG. 1. The type field in each entry in the update agent table illustrated in FIG. 2C may be a list of file extensions in an embodiment according to the present invention. "Other structures and schemas are also contemplated for the update agent table 261. The update agent table 261 may map the name of an update agent to the location where the agent resides in memory. The update agent table 261 may also map the provisioning status of an update agent and the type(s) of updates the update agent is adapted to process. Management of the update agents may be facilitated by the mobile handset 107. New table entries for update agents may be added and entries no longer in use may be deleted, etc. Management of the update agents may be supported by client software, such as a device management client." *See also* Okonnen at Fig. 1. Hashimoto at [0033]-[0034]: "Also, at a time of the task execution interruption, the physical addresses at which the program and the data are arranged are often changed by the system resource management (re-location), and in order to decrypt and normally execute the program encrypted in units of blocks, there is a need for the information to be arranged in a certain address space. For this reason, a virtual address is used in the case of calculating the relative address that is expressed by a difference between the target address and the program start address (offset address). "In this way, the block interchange can be prevented effectively, and the limited physical memory can be utilized efficiently." Hashimoto at [0038]-[0039]:

'117 Claims	Exemplary Disclosures
	"Fig. 1 shows a multi-party application protection environment to which the present invention is to be applied. "A software vendor 121 produces a program 122, and distributes it to a microprocessor-A 103 provided inside a target system 102 of a system user 112. In order to protect the program to be distributed, the plaintext program 122 is encrypted." Hashimoto at [0047]: "The core 902 has a register array 915 and a task switcher 914. The task switcher 914 executes the task switching, that is the saving and the recovery of the context when the interruption occurs. The context is generated in correspondence to the execution of the task, and managed in correspondence to the task. In order to realize this, the task switcher 914 internally has a task state table 921 and a hash calculation unit 922. The task state table 921 maintains the task state in correspondence to the task ID as will be described below. Also, although not shown in the figure, the core 902 issues a notice for change of the context in the register array, a distribution key setting command, etc." Hashimoto at [0048]: "The instruction cache 903 caches the execution codes of the program. In addition to an ordinary cache tag for identifying the address, a tag for identifying a protected task is also attached thereto, and the independence of the application is guaranteed among the internal data of the microprocessor. Note that, although not shown in Fig. 2, the instruction cache 903 internally has an offset table 1207 (see Fig. 9) for describing start addresses (offsets) of the program (execution codes) and the data, as will be described below." *See also* Hashimoto at Fig. 1.

'117 Claims	Exemplary Disclosures
	Fahrny at [0008]: "According to the present invention, a controller for managing media stream decryption keys is provided. The controller comprises a media decryption engine, a table, and a content key list. The media decryption engine generally receives an encrypted media stream from a headend and presents a decrypted media stream to a receiving device in response to a decryption key. The decryption key is generally a function of a content key. The table may contain a content key and index and a plurality of corresponding content keys. Content keys that correspond to a particular encrypted media stream may be selected from the content key list using an index from the content key table, and that is referenced by an identifier received from a headend in connection with the encrypted media stream. The table may optionally (i.e., alternatively) contain initialization vector (IV) values that may be indexed and selected." Fahrny at [0023]: "Keylist: A list of decoder addresses and respective decoder keys in ordered pairs. Keylists may be used by the Uplink Control System (UCS) for generation of authorization messages that are addressed to the diagnostic circuit that is embedded in decoders that are specific to the encoder system." Fahrny at [0037]-[0038]: "For VOD services, a table of keys can be generated and delivered at session setup time. The keys for VOD service may be delivered with synchronization information related to key change as well as other information for short term working key epochs. A VOD Session ID or, alternatively, a Program ID may be used as an index to reference the keys list with the appropriate record of information for the VOD transport decryption. In alternative embodiments of reduced hierarchy key

'117 Claims	Exemplary Disclosures
	management of the present invention, one-way hashing may be implemented in the protection, selection and processing of the decryption key. "The reduced hierarchy key management of the present invention generally provides a new, more secure, and elegant system and method to deliver content keys for decrypting the program streams in conditional access systems (e.g., Broadcast and Video On Demand applications). The key management of the present invention may dramatically reduce the complexity required to deliver new content keys when a first Entitlement Message has been sent (i.e., presented, transmitted, provided, broadcast, etc.) to each set top box. The reduced hierarchy key management system and method of the present invention may be implemented as a portion of a new CAS system. The new CAS system generally provides for the manufacture and distribution of devices that are compatible with infrastructure, regardless of specific content security mechanisms that are used in that infrastructure. The new CAS system may provide far more efficient manufacturing, distribution and operations, and in fact enable new business models, including the retail availability of extremely low cost customer premises equipment (CPE) when compared to conventional approaches." Fahrny at [0043]-[0044]: "The signal PID may be implemented as a program identifier for the respective program that has been selected by a user (e.g., customer, client, viewer, listener, etc.). The signal TFHE may be implemented as at least one entitlement management message (EMM) that is received from and presented to the headend via an out-of-band (OOB) transmission. The working key modifier and application function (e.g., a factor, an operator, or a combination of a factor and an operator that was applied to the content key to enhance encryption) WKM is generally combined with a content key to generate a working key that is used to decrypt an encrypted media stream (e.g., the stream VIDIN) to generate a clear output media stream (e.g., the media stream VIDOUT).

'117 Claims	Exemplary Disclosures
	"The controller 100 generally comprises a media stream transport decryption engine 110, a table 112, a list 114, and a combiner 116. The decoder 100 is generally implemented via at least one processor (e.g., microprocessor, controller, etc.) and at least one memory (e.g., random access memory (RAM), read only memory (ROM), NVROM, flash, EPROM, etc.) where one or more processes, routines, engines, lists, tables, etc. may be stored. The engine 110, the table 112, the list 114, and the combiner 116 are generally implemented within the processor and memory of the decoder 100." Fahrny at [0046]: "The input/output 104 may provide for interfacing that corresponds to (or is related to) entitlement management message (EMM) downloads that are authenticated between the headend (e.g., headend 202, described in more detail in connection with FIGS. 2(a-b)) and the media decoder 100. The input/output 104 may further provide for interfacing that corresponds to downloads to the decoder related to at least one of entitlement structure, content keys lists, IV lists, content key index tables, and digital signatures. Fan at 13-14: FIG. 9 is a flowchart of processing a terminal according to a Push message control policy. In this process, it is assumed that a service type of a Push message is a multimedia multimedia message, and the process includes: Step 901: After receiving the Push message sent by the PI, the Push sending proxy sends the Push message to the Push receiving proxy by using a NOTIFY or MESSAGE message in the SIP message.

'117 Claims	Exemplary Disclosures
	Step 902: After receiving the Push message, the Push receiving proxy returns a 200 OK message to the Push sending proxy to notify the Push sending proxy that the Push message has been successfully received. Step 903: The Push receiving agent checks the service type carried in the Accept-Contact header field in the SIP message carrying the Push message, and determines that the Push message is a multimedia multimedia message. If the terminal does not allow to receive the message of the multimedia multimedia message service in the Push message control policy, the Push receiving agent discards the Push message without sending it to the application module, and the user does not know that a message arrives; if the Push message controls the policy, the terminal If the multimedia MMS service is not allowed to be received, the Push receiving agent temporarily stores the Push message; if the Push message control policy, the terminal allows to receive the message of the multimedia MMS service, and then sends the message to the application module for corresponding service processing. Fan at 26: The terminal sets the Push message control policy, and stores the set Push message control policy locally. After the PI sends the Push message to the Push sending proxy, the Push sending proxy sends the Push message to the terminal, and the terminal controls the policy according to the stored Push message. Process the Push message. FIG. 13 is a flowchart of processing a received Push message by the terminal, where the process includes: Step 1301: The Push sending proxy sends the Push message received from the PI to the Push receiving proxy in the terminal, and the specific method of sending the Push message is that the Push sending proxy carries the Push message in the SIP message header field and sends the Push message to the Push receiving proxy.

'117 Claims	Exemplary Disclosures
	Step 1302: The Push receiving agent compares the matching conditions between the received Push message and the Push message control policy, such as whether the content is too large, whether the keyword is not expected to be received, whether it is a message that is not currently received, and whether it is from a Trusted PI and so on. If the matching condition that the Push message satisfies is the Push content that the terminal is allowed to receive, step 1303 is continued; otherwise, if the matching condition that the Push message satisfies is the Push content that the terminal is not allowed to receive, the Push receiving proxy discards the Push message, if the Push message is satisfied. The matching condition is that the terminal is not allowed to receive the Push content. The Push receiving agent temporarily stores the Push message. After the operation of discarding or temporarily storing the Push message is performed, step 1304 is directly executed. When the Push receiving agent discards or temporarily stores the Push message, before the step 1304 is directly executed, the Push receiving agent may return a message that the Push message has been deleted to the Push sending agent. This step is an optional step. Step 1303: The Push receiving proxy returns an acknowledgement message to the Push sending proxy, and sends the Push message to the application module in the terminal to perform corresponding services. White at [0071]: "FIG. 3 is a high-level block diagram of an environment 300 in which the the QAnywhere messaging system of the present invention may be preferably implemented. As shown at FIG. 3, components of the QAnywhere messaging system include both client-side components on one or more client devices (e.g., mobile devices) connected via a network 330 to server-side components (e.g., on enterprise server systems). Although a single client is depicted at FIG. 3, a typical implementation environment would include a plurality of clients which connect from time to time to the server-side components of the present invention. As

'117 Claims	Exemplary Disclosures
	shown, the QAnywhere client components include a QAnywhere Application 310, a QAnywhere Client Library 311, a Message Store 320, QAnywhere synchronization Hooks 321, a QAnywhere Schema 322, a Transmission Rules Engine 323, a QAnywhere Agent 325, a MobiLink Synchronization Client (dbmisync) 326, and a MobiLink Listener (dbisn) 327. The QAnywhere server components include a MobiLink Server 340, aJMS Connector 343, Other Connectors 341-342, a QAnywhere Notifier 344, a Transmission Rules Engine 345, a Consolidated Database 350, and QAnywhere Synchronization Scripts 351. The client-side and server-side components and their operations are described in more detail below." *See also* White at Fig. 3. Holbrook at [0085]: "Portable messaging units 16 are usually disconnected from all other elements of messaging system 10, except when a user brings an individual portable messaging unit 40 to an individual messaging node 20, at which time temporary second communications link 72 may be established between portable messaging unit 40 and messaging node 20. Portable messaging unit 40 and messaging node 20 are arbitrary members of portable messaging units 16 and messaging nodes 14, respectively; any portable messaging unit may be connected at any messaging node. Individual users of messaging system 10 have user accounts, which may be associated with one or more distinct messaging address identifiers, such as e-mail addresses. An instance of portable messaging unit 40 may be configured for operation with one or more user accounts." Cooley at Abstract: "Systems and methods are provided that allow an application residing on a user device to intercept messages directed to that device and to extract control

'117 Claims	Exemplary Disclosures
	information and/or data from the intercepted messages for use by the application. In one embodiment, the intercepted message can cause the application to change its configuration or to present information to the user that came from a sending website. In another embodiment, a message from a device can trigger a website to send back a desired piece of data. The returned data may contain a flag, which directs the data to a formatting server. The formatting server then converts the data into a format that allows the data to be intercepted by the desired application on the user device. This then allows the desired data to be presented to the user in the context of an application resident on the user's device." Cooley at ¶[0001]: "This disclosure relates to the control and customization of applications resident on a wireless device from locations remote therefrom and more specifically to systems and methods for allowing applications resident on a mobile device to be controlled using the text messaging capability of the wireless network." Cooley at ¶[0005]: "Systems and methods are provided that allow an application residing on a user device to intercept messages directed to that device and to extract control information and/or data from the intercepted messages for use by the application. In one embodiment, the intercepted message can cause the application to change its configuration or to present information to the user that came from a sending website. In another embodiment, a message from a device can trigger a website (such as an airline website) to send back a desired piece of data. The returned data can, for example, contain a flag, which directs the data to a formatting server. The formatting server then converts the data into a format that allows the data to be intercepted by the desired application on the user device. This then allows the

'117 Claims	Exemplary Disclosures
	desired data to be presented to the user in the context of an application resident on the user's device." Cooley at ¶[0013]: "FIG. 1 shows one embodiment of a system, such as system 10, in which the concepts of the invention can be practiced. In FIG. 1 there is shown a link from a user device, such as cellular telephone 15 to website 11. This link extends from the user device through a carrier network, such as network 14 and via service center 12 to the desired website. The message returned from website 11 contains an HTTP request against service center 12. The service center then queues a Short Message Service (SMS) message back to device 15 through an SMS broker, such as broker 13. Broker 13 then routes an SMS message through any carrier network, such as network 14, which need not be the same network that was used for the originating message from the device." Cooley at ¶[0015]: "When the incoming text message arrives at the user's device it is intercepted, under control of the device processor and an application, such as application 31 (as will be discussed), running on the device. This is possible because application 31 is registered with the device operating system when it was loaded on the device. Application 31 then routes the incoming message to the proper specific application in accordance with the data contained in the text message. Thus, in the situation with the flight reservation to Las Vegas (as discussed above), information pertaining to that flight is routed to a specific application, such as to air travel application 22-1 (shown in FIG. 2B), which application is dedicated to air travel data pertaining to the device user." Cooley at ¶[0018]:

'117 Claims	Exemplary Disclosures
	"FIG. 3 shows one embodiment of a system for controlling messages on a user device. The incoming SMS message would normally be delivered to level 32 of network stack 30 for subsequent display to device user via, for example, display 16 (FIG. 1). The display would be in text or visual form and would then go into the SMS box in text, graphics or video form as is well-known. However, application 31 intercepts the message because that application is registered with the platform to receive certain SMS messages. This protocol varies from device to device based on operating system. SMS or HTTP transports can be used via the side channel process. SMS has the advantage of being a true "push" notification. Application 31 then routes the incoming message, based on the context and content of the data contained in the message to an individual tile, such as to tile 22-1 (FIG. 2B) or even more specifically to tile 24-1 (FIG. 2D), as discussed above." Cooley at ¶[0020]: "Another method for doing this would be to run a Simple Mail Transfer Protocol (SMTP) gateway at service center 12. This would allow application 31 to receive incoming messages in email form. Such email messages could then be parsed to see if they contain certain information, such as the tile ID, etc. This approach would benefit from a fixed format email message. The parsing includes filtering the email messages to see if they contain certain information. The filtered information then would be sent to the device by an SMS message. Second, the formatted SMS message could be sent to the service center for delivery to the particular user device." Cooley at ¶[0028]: While FIG. 5 shows a purpose computer, it should be noted that the exact configuration of a portion of a system according to various embodiments may be slightly different. For example, handheld devices according to one or more

'117 Claims	Exemplary Disclosures
	embodiments may be any kind of processor-based device, such as a cell phone, a Personal Digital Assistant, a specialized device, and/or the like. Additionally, servers (e.g., servers 11, 12 of FIG. 1) and the SMS broker according to one or more embodiments may be any kind of processor-based device capable of sending messages (e.g., email messages, SMS messages, MMS messages), such as a personal computer, a server-type computer, and the like. Moreover, embodiments of the present invention may be implemented on application specific integrated circuits (ASICs) or very large scale integrated (VLSI) circuits. In fact, persons of ordinary skill in the art may utilize any number of suitable structures capable of executing logical operations according to the embodiments of the present invention." Wakasa at ¶[0031]: "Referring to FIG. 1, the entire configuration of an information providing system 10 according to an embodiment is explained below. As shown in FIG. 1, the information providing system 10, for example, includes portable terminal devices 100 *a*, 100 *b* and 100 *c*, a server device 200, a network 300, readers/ writers 400 *a* and 400 *b* and a PC (personal computer) 500. In the following description, the portable terminal devices 100 *a*, 100 *b* and 100 *c* are each denoted by generic reference numeral 100 if it is not necessary to distinguish the portable terminal devices 100 *a*, 100 *b* and 100 *c* from each other. By the same token, the readers/ writers 400 *a* and 400 *b* are each denoted by generic reference numeral 400 if it is not necessary to distinguish the readers/ writers 400 *a* and 400 *b* from each other." Wakasa at ¶[0040]: "The server device 200 generates a transmitted message including parameter information and the identification information of an application to be executed by the portable terminal device 100 on the basis of the parameter information. Then,

'117 Claims	Exemplary Disclosures
	the server device 200 transmits the transmitted message to the portable terminal device 100 via the network 300. The parameter information is typically information used for requesting the application of the portable terminal device 100 to display information desired by the user or carry out predetermined processing. For example, the parameter information is information requesting the application of the portable terminal device 100 to display "Hello" on the display screen. The portable terminal device 100 executes the application corresponding to the identification information of the application included in transmitted message by using the parameter information as input data. For example, by using the parameter information requesting the application of the portable terminal device 100 to display "Hello" on the display screen as input data, the portable terminal device 100 executes the application corresponding to the identification information, displaying "Hello" on the display screen of the portable terminal device 100. As an alternative, the server device 200 holds an application to be provided to the portable terminal device 100 and, in accordance with a request made by the portable terminal device 100, the server device 200 transmits the application to the portable terminal device 100 via the network 300." Wakasa at ¶[0051]: "FIG. 2 is a block diagram showing the functional configurations of the portable terminal device 100 and the server device 200. As shown in FIG. 2, the portable terminal device 100 includes a transmitted message receiver section 102, an application determination section 104, an application-transmission request sender section 106, an application-transmission request generation section 108, an application receiver section 110, a display section 112 and an application storage section 114." Wakasa at ¶[0052]:

'117 Claims	Exemplary Disclosures
	"The transmitted message receiver section 102 is a communication interface such as a communication line, a communication circuit, a communication device or the like. The transmitted message receiver section 102 has a function for receiving a transmitted message from the server device 200. A transmitted message received by the transmitted message receiver section 102 is passed on to the application determination section 104. As described above, a transmitted message received from the server device 200 includes parameter information and the identification information of an application to be executed by the portable terminal device 100 on the basis of the parameter information." *See also* Wakasa, Fig. 2. Wakasa at ¶[0084]: "The controller 124 is a section for receiving a transmitted message transmitted by the non-contact-communication IC chip 120 via the second communication section, executing an application corresponding to identification information included in the transmitted message and displaying a result of the execution on a display screen of the portable terminal device 100. More specifically, when the controller 124 receives a transmitted message from the non-contact-communication IC chip 120, the controller 124 passes on the transmitted message received from the non-contact-communication IC chip 120 to the acceptance application 126. The acceptance application 126 then provides parameter information to the application corresponding to the identification information of the application included in the transmitted message. The identification information, such as ID 156 or URL 154, is used by the acceptance application 126 to determine as to whether or not the application corresponding to the identification information has been stored in the application storage section 114." Satake at ¶[0069]:

'117 Claims	Exemplary Disclosures
	"The mobile communication terminal according to a first aspect of the present invention comprises: a receiving section that receives a message; a storage section that stores an application that requests a server for predetermined information and receives the information; and a determining section that, when the receiving section receives a message that instructs that a predetermined application be started up, determines whether or not a term of validity is attached to the message, and , based on the term of validity, starts up the instructed application that is stored in the storage section." *See also* Satake at ¶¶[0073]-[0080]. Satake at ¶[0076]: "The mobile communication terminal according to the third aspect of the present invention comprises: a storage section that stores a plurality of applications; a receiving section that receives message that contains a startup instruction for a specific application and information on that application; a storage section that, for each of the stored applications, stores the application information and also stores information that shows whether or not the application is able to be started up by the startup instruction; and a control section that, when the receiving section receives the message instructing that the application be started up, determines whether or not startup of the application is possible by collating the application information contained in the message with the information that shows whether or not the application in the storage section is able to be started up, and controls a startup of the application based on a result of the determination."
[1b] a network message server supporting a plurality of secure Internet data connections, each secure Internet data connection between the	The references below disclose and/or render obvious this limitation. For example, see the following passages and/or figures, as well as related disclosures: Chou at 2:27-37:

'117 Claims	Exemplary Disclosures
network message server and a respective one of the mobile end-user devices via a device data connection to a wireless network,	"The mobile client device may register with the enterprise application through interaction with the wireless secure server. For example, the wireless secure server may comprise a password-protected register connector through which registration information associated with the mobile client device is supplied from the wireless secure server to the enterprise application. The wireless secure server may be operative to obtain user and device profile information from the mobile client device, and to store the profile information temporarily until the profile information is extracted from the wireless secure server by the enterprise application." Chou at 3:33-47: "FIG. 1A shows an example communication system 100 in which the present invention is implemented. The system 100 includes a network 102, an enterprise application 104, and a wireless secure server (WSS) 106. The enterprise application 104, which is assumed to run on a computer, server or other processing element associated with a given enterprise, is separated from the WSS 106 via an enterprise firewall 108, and receives multimodal input such as voice, data, etc. from sources not shown in the figure. The system 100 further includes a wireless application protocol (WAP) mobile client device 110, which communicates with the WSS 106 over network 102. More specifically, the mobile client device 110 communicates with the WSS 106 via WAP gateway (WG) 112, and via a push proxy gateway (PPG) 114 and a push initiator (PI) 116, as shown." Chou at 5:14-23: "Network 102 may represent, e.g., a global communication network such as the Internet, an intranet, a wide area network, a metropolitan area network, a local area network, a wireless cellular network, or a satellite network, as well as portions or combinations of these or other wired or wireless communication networks.

'117 Claims	Exemplary Disclosures
	"The network 102 in the illustrative embodiment may thus be viewed as a converged network which includes a mobile wireless network portion and an Internet and/or intranet network portion, as will be illustrated in FIGS. 1B and 1C." Chou at 5:48-54: "The WSS 106 in the illustrative embodiment provides secure connections to a dialogue system associated with enterprise application 104. Multimedia service content is preferably submitted by the enterprise application 104 to the WSS 106 using HTTP. As indicated previously, the WSS is connected to the network 102 and accessible by mobile client 110 outside the firewall 108." Chou at 6:15-28: "With reference to FIG. 1B, communication in accordance with a simple WAP application architecture is illustrated. In this example, WAP gateway 112 communicates with a content server 150 and with the WAP mobile client device 110 via respective portions 102A and 102B of the network 102. As mentioned previously, the network 102 in the illustrative embodiment may be viewed as a converged network which combines the capabilities of a mobile wireless network 102A and an Internet and/or intranet network 102B. The content server 150 in this example may be any computer or other server accessible to the WAP gateway 112 via the Internet, an intranet, or other communication network. The content server may therefore represent, for example, the WSS 106 or other server of the system 100." Rakic at [0065]-[0068]: "Secure push message server 504 may send secure push messages to client device 102 via a wireless network or gateway 106. Given a message or a payload from an

'117 Claims	Exemplary Disclosures
	application, a user, or an administrator, secure push message server 504 may generate a key, an electronic signature, and a secure push message. "Secure push message server 504 may generate the key based on codes (e.g., IMEI) that are obtained from client device 102, a component in client device 102 (e.g., removable memory 218), and an administrator. Depending on the implementation, additional codes or information may be used to generate the key. Once the key is generated, secure push message server 504 may generate an electronic signature based on the key and the message. "In possession of the electronic signature and the key, secure push message server 504 may generate and send the secure push message to client device 102. In one implementation, secure push message server 504 may generate the secure push message by appending the electronic signature to the message and/or other information (e.g., a timestamp). "In addition to generating/sending secure push messages to client device 102, secure push message server 504 may obtain and store information that is needed to generate the key. In one implementation, while configuring client device 102, secure push message server 504 may obtain a client device identifier from client device 102, subscriber information from removable memory 218 that is not yet installed in client device 102, and an administrator code from an administrator. Furthermore, secure push message server 504 may store the client device identifier, the information, and the administrator code in a database." Anderson at 669-670: "Another approach is to do encryption and/or authentication at the IP layer using a protocol suite known as Ipsec. IPsec defines a *security association* as the combination of keys, algorithms and parameters used to protect a particular packet stream. Protected packets are either encrypted or authenticated (or both); in the

'117 Claims	Exemplary Disclosures
	latter case, an authentication header is added that protects data integrity using HMAC-SHA1, while in the former the packet is encrypted and encapsulated in other packets. (The use of encryption without authentication is discouraged as it's insecure [151].) There's also an *Internet Key Exchange* (IKE) protocol to set up keys and negotiate parameters. IKE has been through a number of versions (some of the bugs that were fixed are discussed in [465]). "IPsec is widely used by firewall vendors who offer a *virtual private network* facility with their products; that is, by installing one of their boxes in each branch between the local LAN and the router, all the internal traffic can pass encrypted over the Internet. Individual PCs, such as workers' laptops and home PCs, can in theory join a VPN given a firewall that supports IPsec, but this is harder than it looks. Compatibility has been a major problem with different manufacturers' offerings just not working with each other; although firewall-to-firewall compatibility has improved recently, getting random PCs to work with a given VPN is still very much a hit-or-miss affair." Anderson at 670-71: "Recall that when discussing public key encryption, I remarked that a server could publish a public key *KS* and any web browser could then send a message *M* containing a credit card number to it encrypted using *KS*: $\{M\}_{KS}$. This is in essence what the TLS protocol (formerly known as SSL) does, although in practice it is more complicated. It was developed to support encryption and authentication in both directions, so that both http requests and responses can be protected against both eavesdropping and manipulation. It's the protocol that's activated when you see the padlock on your browser toolbar." "Here is a simplified description of the version as used to protect web pages that solicit credit card numbers:

'117 Claims	Exemplary Disclosures
	1. the client sends the server a *client hello* message that contains its name *C*, a transaction serial number *C#*, and a random nonce N_C; 2. the server replies with a *server hello* message that contains its name *S*, a transaction serial number *S#*, a random nonce N_S, and a certificate *CS* containing its public key *KS*. The client now checks the certificate *CS* back to a root certificate issued by a company such as Verisign and stored in the browser; 3. the client sends a *key exchange* message containing a *pre-master-secret* key, K_0, encrypted under the server public key *KS*. It also sends a *finished* message with a message authentication code (MAC)computed on all the messages to date. The key for this MAC is the *master-secret*, K_1. This key is computed by hashing the pre-master-secret key with the nonces sent by the client and server: $K_1 = h(K_{CS}, N_C, N_S)$. From this point onward, all the traffic is encrypted; we'll write this as $\{\ldots\}_{KCS}$ in the client-server direction and $\{\ldots\}_{KSC}$ from the server to the client. These keys are generated in turn by hashing the nonces with K_1. 4. The server also sends a finished message with a MAC computed on all the messages to date. It then finally starts sending the data. $C \rightarrow S : C, C\#, N_C$ $S \rightarrow C : S, S\#, N_S, CS$ $C \rightarrow S : [K0]_{KS}$ $C \rightarrow S : [finished, MAC(K_1, everythingtodate)]_{KCS}$ $S \rightarrow C : [finished, MAC(K_1, everythingtodate)\}_{KSC}, [data]_{KSC}$ "The design goals included minimizing the load on the browser, and then minimizing the load on the server. Thus the public key encryption operation is done by the client, and the decryption by the server; the standard encryption method (*ciphersuite*) uses RSA for which encryption can be arranged to be very much faster than decryption. (This was a wrong design decision as browsers

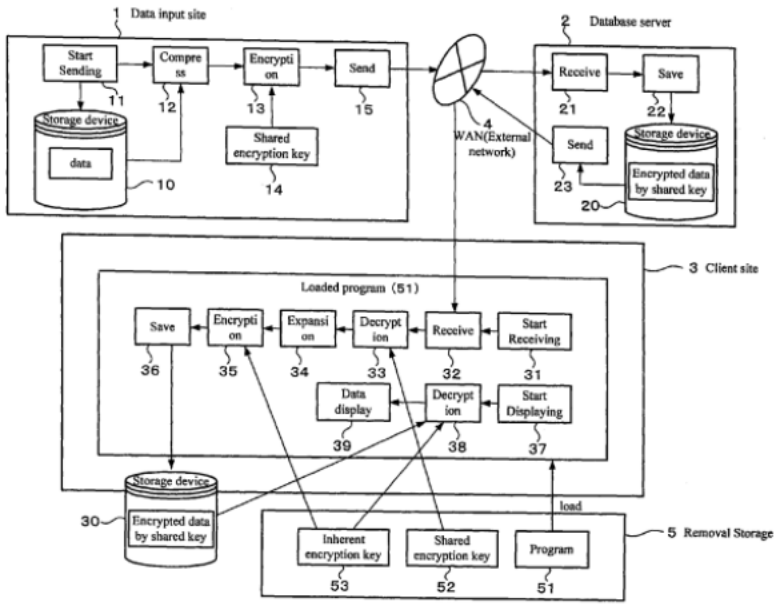
'117 Claims	Exemplary Disclosures
	generally have a lot more compute cycles to spare than servers; it has created a brisk aftermarket for crypto accelerator boards for web servers.) Also, once a client and server have established a pre-master secret, no more public key operations are needed as further master secrets can be obtained by hashing it with new nonces." *See generally* Anderson at Ch. 20 ("Telecom System Security"); § 21.4.5.6 ("TLS"). Shenfield at [0210]: "Serial port 2230 in Figure 22 , would normally be implemented in a personal digital assistant (PDA)-type mobile station for which synchronization with a user's desktop computer (not shown) may be desirable, but is an optional device component. Such a port 2230 would enable a user to set preferences through an external device or software application and would extend the capabilities of mobile station 2200 by providing for information or software downloads to mobile station 2200 other than through a wireless communication network. The alternate download path may for example be used to load an encryption key onto the device through a direct and thus reliable and trusted connection to thereby enable secure device communication. As will be appreciated by those skilled in the art, serial port 2230 can further be used to connect the mobile device to a computer to act as a modem." Ozaki at 3:40-60: "The network layer (third layer) is a layer which defines a communication method between two arbitrary nodes. In the case of TCP/IP, the third layer corresponds to an IP layer. It is possible in the data-link layer to carry out a communication between nodes on the same network medium. The role of the network layer is to communicate by using routing between two arbitrary nodes existing on the network. Here, "routing" refers to transmitting IP packets by selecting an optimum path when packets are transmitted to a host in the TCP/IP network. For example,

'117 Claims	Exemplary Disclosures
	communication is carried out in the network layer between two Ethernet segments by routing packets. Also, it is possible to route packets not through physical network media but by routing to a dial-up PPP (Point to Point Protocol) line which connects a computer to a network (Ethernet) through a telephone line, also by routing to an exclusive line using an optical fiber and the like. For this purpose, addresses (IP addresses in case of TCP/IP) which are not dependent on the physical medium are ordinarily allotted to respective nodes and the routing is carried out based thereupon. IPsec encrypts every communication in this network layer." Ozaki at 2:52-3:3: "FIG. 25 shows an example of a TCP/IP protocol stack relative to the OSI model illustrating an encryption communication by using conventional IPsec. FIG. 26 shows an example of a TCP/IP protocol stack relative to the OSI model in a case of carrying out an encryption communication by using conventional SSL. "The OSI reference model is well known and is constituted such that a bottom layer (first layer) is a physical layer, a second layer is a data-link layer, a third layer is a network layer, a fourth layer is a transport layer, a fifth layer is a session layer, a sixth layer is a presentation layer and a top layer (seventh layer) is an application layer. The hierarchy in this OSI reference model is illustrated by dividing a communication function into seven stages and a standard function module is set for every layer. In the TCP/IP of FIG. 25, only five layers of the OSI reference model are shown. "The TCP/IP protocol stack is typically expressed in software or firmware, each protocol of the stack being conceptually positioned in a respective OSI layer of the hierarchy." *See also* Ozaki at 3:4-4:27, Figs. 25-26. Mostafa at 731:

'117 Claims	Exemplary Disclosures
	"A simplified MMS architecture is depicted in Figure 1. An MMS user agent (UA), also known as an MMS client, provides means for composing, sending, retrieving, viewing, and other controlling functions to users. Storage, address resolution, routing, billing, content adaptation, routing reports, and interworking are important functions of an R/S. It also interacts with other entities (e.g., other R/Ss, VASP, other messaging systems) as shown in Figure 1." *See also* Mostafa at Figs. 1, 3; *see generally* Mostafa at 730-31. Ogawa at 3:15-34: "DETAILED DESCRIPTION A data encryption system is described herein, an exemplary embodiment employing data encryption, removable storage, and associated methodology to compresses and encrypt data based on a shared encryption key. The compressed an encrypted data is transmitted to an external network in which a database server operably linked to the external network may stores, manage, transmit, and receive the data. A removable storage stores an identification code, and a client site which is configured to receive the removable storage generates an encryption key based on the identification code, receives encrypted data from the external network, decrypts and expands the received data based on the shared encryption key, encrypts the data based on the encryption key and saves the encrypted data, and finally decrypts the encrypted data and outputs the data using the encryption key. In the drawings, the same reference numerals are used for designating the same elements throughout the several figures."

'117 Claims	Exemplary Disclosures

Figure 1



Ogawa at 3:35-5:40

"FIG. 1, is a high level block diagram of an information and telecommunication system in accordance with an exemplary embodiment. As shown in FIG. 1 components of the system include a data input site 1, database server 2, and a client site 3 connected via the external network 4. While additional components may be utilized, such as routers, bridges, network switches, enterprise infrastructure, such components are omitted as known to those in the art, the description of which is unnecessary.

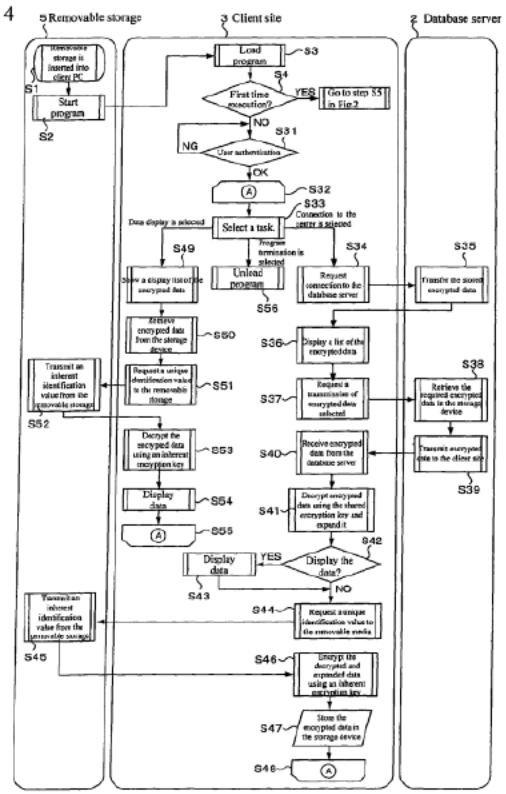
The components of FIG. 1 are shown operably linked via an external wide area telecommunication network 4. In an exemplary embodiment, the external wide area (WAN) telecommunication network 4 is the global communications network,

'117 Claims	Exemplary Disclosures
	commonly referred to as the Internet. The network components communicate via WAN 4 through hyper text transfer protocol (HTTP) in this exemplary embodiment. Of course, other connectionless or connection-oriented systems and protocols are equally embraced by the embodiment described herein. Those skilled in the art recognize that while an exemplary embodiment is defined as being used with the Internet, the present invention is equally applicable to any local area network (LAN) or other WAN configuration whether land-based or wireless in form. Similarly, the system described herein may be employed in a stand-alone configuration in which the system components are utilized in as a local application. In addition to the encryption of the embodiments described herein, further technologies may be utilized to enhance security over networks to effectively tunnel data over the network 4 as compressed and encrypted in accordance with the present advancements. For example, in network communications utilizing a conventional TCP/IP (Transmission Control Protocol/Internet Protocol) or UDP (User Datagram Protocol), encryption communication, such as IPsec (Internet Protocol Security) or SSL (Secure Socket Layer), is utilized to prevent some security risks presented during the exchange of data between network terminals. In general, encryption unit 13 would additionally provide either a common key (also called a secret key) cryptograph system, and/or a public key cryptograph system. Referring again to FIG. 1, The data input site 1 includes a storage device 10, and functional components transmission initiation unit 11, compression unit 12, encryption unit 13, shared encryption key 14, and a transmission unit 15. In the exemplary embodiment, units 11-15 are implemented as a single computer executable instruction set of a data processor of data input site 1, not shown. The specific coding of these functions may be expressed as object oriented modules, cooperating and/or independent daemons, or separate utility applications. In an alternative embodiment, these functions may be expressed via independent hardware implementations such as a FPGS, ASIC, microcontroller or PLD.

'117 Claims	Exemplary Disclosures
	Additionally, transmission unit 15 may be embodied as a NIC interface, or suitably configured network interface card or wireless transmission unit. The exemplary data input site 1 of FIG. 1 is a hardware platform of a personal computing device such as a PC employing an Intel Pentium processor. The instruction set of units 11-15, as well those of database server 2 and client site 3 may be provided as a utility application, background daemon, or component of an operating system, or modified TCP/IP stack executing in conjunction with a local processor and operating system such as Microsoft VISTA®, Unix, Solaris, Linux, Apple MAC-OS and other systems known to those skilled in the art. Alternatively, those skilled in the art will recognize the applicability to mobile devices such as PDAs, phones, and portable entertainment devices which employ Symbian, Microsoft Mobile® and other mobile operating systems. Memory required for supporting the registries, kernel and like features of the data input site 1 of FIG. 1 is omitted as well known. Likewise the description of general features of the data input site 1 such as local volatile and/or non-volatile memory, I/O capabilities, common peripheral devices, as well as corresponding functionality have been omitted for brevity, the detailed operation/description of which is well known to those skilled in the art. The specific coding and porting of the algorithms described herein is within the ability of one skilled in the art upon review of this specification and drawings. The storage device 10 of data input site 1 is designated for storing and transmitting the data. If a start instruction is issued by the transmission initiation until 11, the compression unit 12 will compress the data stored on the storage device 10 in accordance with a compression algorithm as detailed below. Compressed data are supplied to the encryption unit 13 and encrypted using a shared encryption key 14. In addition, data encrypted by the shared encryption key 14 (hereafter referred to as

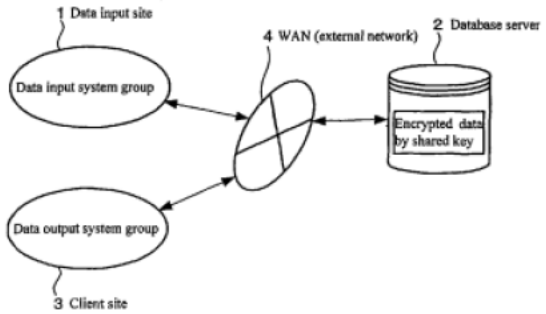
'117 Claims	Exemplary Disclosures
	shared key encrypted data) is transmitted to the network 4 via a transmission unit 15. A database server 2 houses a receiving unit 21, a storage device 22, and a sending unit 23. All the shared key encrypted data received from the network 4 is stored on a storage device 20 in an encrypted state. In the exemplary embodiment, units 21-23 are implemented as a single computer executable instruction set of a data processor of database server (2) not shown. The specific coding of these functions may be expressed as object oriented modules, cooperating and/or independent daemons, or separate utility applications. In an alternative embodiment, these functions may be expressed via independent hardware implementations such as a FPGS, ASIC, microcontroller or PLD. Additionally, transmission unit 15 may be embodied as web front end, NIC interface, or suitably configured network interface card or wireless transmission unit. As recognized by one of skill in the art, the database server 2 of FIG. 1 may embrace a hierarchical model, network model, relational model, or other database model. A web front end may be provided to present a graphical user interface (GUI) for accessing the database server 2 from a remote location. The database server 2 components may employ a windows based operating system, however alternative operating systems may include but are not limited to Unix, Solaris, Linux, as well as Apple MAC-OS. Specific implementation of this database server 2 may embrace a database management system such as Oracle, Microsoft SQL Server, MySQL, Microsoft Access, Nomad, and the like. The encrypted data stored on the storage device 20 is transmitted to the network 4 via a sending unit 23 when a request is received from the client site 3. Consequently, data transmitted from the sending unit 23 also serves as the shared key encrypted data enciphered with the shared encryption key 14.

'117 Claims	Exemplary Disclosures
	The client site, generally designated 3 includes a storage device 30, client starting unit 31, receiver unit 32, decryption unit 33, decompression (expansion) unit 34, and an encryption unit 35. Likewise, save unit 36 is shown for writing data to storage device 30 and display units 37-39 provide the necessary functionality for rendering decoded data to a display of client site 3, not shown. In the exemplary embodiment, units 31-39 are implemented as a single computer executable instruction set 51. The specific coding of these functions may be expressed as object oriented modules, cooperating and/or independent daemons, separate utility applications. In an alternative embodiment, these functions may be expressed as independent hardware implementations such as a FPGS, ASIC, microcontroller or PLD. Additionally, transmission unit 15 may be embodied as a NIC interface, or suitably configured network interface card or wireless transmission unit." Ogawa at 6:32-41 "Any processes descriptions or blocks in flow charts should be understood as representing modules, segments, portions of code which include one or more executable instructions for implementing specific logical functions or steps in the process, and alternate implementations are included within the scope of the exemplary embodiment of the present invention in which functions may be executed out of order from that shown or discussed, including substantially concurrently or in revere order, depending upon the functionality involved, as would be understood by those skilled in the art."

'117 Claims	Exemplary Disclosures
	Figure 4  Ogawa at 7:42-8:56 "The flow chart in FIG. 4 shows processing operations to move data from the database server 2 to the client site 3. FIG. 4 shows the removable storage 5 on the left, the client site 3 in the center, and the database server 2 on the right. Initially, if the removable storage 5 is operably linked to the client site 3 at step S1, the program 51 will begin step S2, and the program 51 will be loaded into the client site 3 in step S3. At the client site 3, it will be determined if this is the first time

'117 Claims	Exemplary Disclosures
	execution of the program in step S4. The above description is for the same operating procedure and steps S1-S4 as given for FIG. 2. If this is the first time execution of the program in step S4 (YES), the process continues to step S5 in FIG. 2 by displaying the operating procedure for exchanging shared encryption keys. Meanwhile, if it is not the first time (NO) in step S4, the user authentication procedure will be performed in step S31. If authentication is not confirmed (NO), step S3 will be repeated. If authentication is confirmed (OK), the process will move to the starting point for repeating the process at step S32. Next, the desired operation is selected at the client site 3 in step S33. Here, the connection to the center is selected. The client site 3 will request a connection to the database server 2 in step S34. If a connection request is received, the database server 2 transfers a list of the shared key encrypted data stored in the storage device 20 in step S35. The client site 3 displays the list of encrypted data in step S36, and requests transmission of the encrypted data in step S37. Then, the database server 2 retrieves the encrypted data from the storage device in step S38, and transmits it to the client site 3 in step S39. The client site 3 receives the shared key encrypted data from the database server 2 in step S40, and decrypts and compresses the received data using the shared encryption key in step S41. Furthermore, the client site 3 determines whether or not the data should be displayed in step S42, and displays the data in step S43 when display is selected (YES). When no display is selected (NO) in step S42, and after displaying the data in step S43, the client site 3 requests an identification code 300 for the removable storage 5 in step S44. In response to the request, the removable storage 5 transmits an identification code 300 in step S45. In the client site 3, the shared key encrypted data are decrypted and compressed in step S41, and the decrypted data are re-encrypted with the inherent encryption key

'117 Claims	Exemplary Disclosures
	in step S46, and stored on the storage device in step S47. This is the process by which the system stores and displays the shared key encrypted data received from the database server 2. After the data is stored this way, the operation will return to step S32 via step S48. If a data display is selected in step S33, the program will display a list of the inherent key encrypted data stored on the storage device in step S49 after retrieving the encrypted data from the storage device in step S50. In response, the removable storage 5 transmits an identification code 300 in step S52. The client site 3 decrypts the inherent key encrypted data retrieved from the storage device 30 using the inherent encryption key received from the removable storage 5 in step S53. This is the process by which the data stored on the storage device is displayed. After retrieving and displaying the data, the operation will return to step S32 via step S55. If program is ended in step S33, the program will be unloaded in step S56. Thus, data entered at the data input site 1 and stored in the database server 2 can be displayed on the client site 3. And, according to the exemplary embodiment, because the data stored in the database server 2 is encrypted with the shared encryption key, it is very difficult to falsify the data. Therefore, even if the data are stolen, there are very few possibilities that the data may be decoded and abused without the encryption key. Furthermore, because the downloaded data is saved after being encrypted with an inherent encryption key, unless there is a corresponding removable storage 5 which holds the inherent encryption key, there is no way to decrypt the saved data. Therefore, if a laptop with downloaded data is stolen, and if the removable storage 5 is secured separately the data of the laptop remains secure. For example, if sales personnel carry only the removal storage and procure a PC at the customer destination, the data downloaded to the PC hard disc can be encrypted

'117 Claims	Exemplary Disclosures
	with an inherent encryption key. In this situation, there is no way of decrypting the data unless there is a corresponding removable storage 5." Figure 5  Ogawa at 8:57-64 "FIG. 5 shows an overview of the architecture in accordance with the exemplary embodiment. That is, in FIG. 5, the data input system group is provided as data input site 1 and the data output system group as the client site 3. Between these two groups, data are transmitted and saved through the database server 2 and the external network 4. As shown, this exemplary data encryption system transmits and saves data between data the input system group and the data output system groups." *See also* Ogawa at 1:45-2:44, [1a]. *See generally* TLS 1.0, TLS 1.1, and SSL. Lu at [0260]:

'117 Claims	Exemplary Disclosures
	11.1. SSL/TLS Overview Lu at [0261]: Nearly all secure transactions over the Internet are carried out using either Secure Socket Layer (SSL) or its successor Transport Layer Security (TLS). These protocols provide a mechanism by which data is authenticated and encrypted as it flows through the underlying network communication links. These links are considered insecure and open to attack by third parties. Instead of achieving network security by changing the underlying transmission protocol, as done by IPSec, or by modifying higher level application protocols, as done by S-HTTP, SSL/TLS adds a separate security protocol layer located between applications and underlying transmission protocols. FIG. 16 is a schematic illustration of the SSL/TLS security layer relative to the application layer and transmission layers. The Application Layer encompasses protocols such as SMTP, HTTP, Telnet and FTP. The Application Layer makes calls to the SSL/TLS layer for implementation of secure communication. The SSL/TLS layer, in turn, is above the transmission layer, e.g., TCP/IP. Lu at [0262]: As seen from FIG. 16, SSL/TLS requires a reliable, bi-directional communication layer such as TCP. Lu at [0263]: For the reader's convenience a short tutorial in SSL/TLS is provided here. More detailed information about these protocols may be found in references, such as, (Dierks, T., Allen, C., "The TLS Protocol, Version 1.0", IETF Network Working Group. RFC 2246. See http://www.ietf.org/rfc/rfc2246.txt), which is incorporated herein by reference.

'117 Claims	Exemplary Disclosures
	Lu at [0264]: 11.2. Cryptography in SSL Lu at [0265]: The SSL protocol is built on top of existing cryptographic services that have been well established and are considered mathematically secure. A high level view and uses of these services are outlined in the Table 2 below. TABLE 2 Cryptographic Service / Use / Protects Against Authentication / Prove Identity / Forgery, digital impersonation Confidentiality / Keep Secret / Eavesdropping Message Integrity / Verify information / Alteration during transmission Lu at [0266]: The SSL protocol uses each of these cryptographic services to establish a secure connection between client and server systems. Lu at [0267] Authentication is the process of identifying yourself to another party. It proves that you really are who you say you are. There is little or no value in achieving network security if you are securing your communication with the wrong party. In SSL, server side authentication is mandated by the protocol, where as client

'117 Claims	Exemplary Disclosures
	authentication is optional. Authentication is done via the use of digital certificates. SSL supports RSA and DSS algorithms. The certificates are sent in X.509 format. Lu at [0268]: Confidentiality is achieved by encrypting the data before it is sent to the other party. Encryption is done using a symmetric algorithm like DES, Triple-DES, RC4, etc. with different key lengths. Client and server generate the symmetric keys used for encryption and decryption by first exchanging a secret. This secret is exchanged using one of two public key algorithms: RSA or Diffie-Hellman. Lu at [0269]: Finally, message integrity is maintained by appending a MAC to each message. In TLS, MAC is generated using HMAC, a standard method of computing MAC values. Both MD5 and SHA-1 algorithms are used in this process. Lu at [0270]: SSL ensures that each of these three cryptographic features is maintained during an SSL session. If any cryptographic feature is found to be compromised, the SSL session is closed. For more detail on application of cryptography see, for example, the authoritative book on this subject by Bruce Schneier (Schneier, B., Applied Cryptography, 2nd Edition, John Wiley & Sons, New York, N.Y. (1996)). Lu at [0271] 11.3. TLS 1.0 Protocol Lu at [0272] Transport Layer Security, TLS 1.0 is a standard protocol developed under the auspices of IETF. However, it involves relatively modest, and incremental security

'117 Claims	Exemplary Disclosures

improvements over SSL 3.0. There is less difference between SSL 3.0 and TLS, than there is between SSL 2.0 and SSL 3.0. In fact TLS version sent in the handshake messages is 3.1, with 3 being the major version and 1 being the minor version.

Lu at [0273]

Some of the main differences between SSL 3.0 and TLS 1.0 are outlined in Table 3.

TABLE 3

Property	SSL 3.0	TLS 1.0
Protocol version in handshake message	3.0	3.1
Types of Alert messages	12	23
MAC	non-standard	standard, HMAC
Generating key material	non-standard	standard, PRF
Certificate Verification	complex	simple
Finished message	Non-standard	standard, PRF

Lu at [0274]

The basic design of TLS has a notion of two distinct phases: handshake phase and data transfer phase. During the handshake phase, the client authenticates the server while the server can optionally authenticate the client. They both establish a set of cryptographic keys which are then used to secure the data during application phase. The handshake phase must complete successfully before application data exchange can take place.

'117 Claims	Exemplary Disclosures
	Tyhurst at [0028]: The mobile data server 120 also provides a secure exchange of data (e.g., email messages, personal information manager (PIM) data, and IM data) with the mobile communication devices 201. In at least some embodiments, communications between the mobile data server 120 and the mobile communication devices 201 are encrypted. In some embodiments, communications are encrypted using a symmetric encryption key implemented using Advanced Encryption Standard (AES) or Triple Data Encryption Standard (Triple DES) encryption. Private encryption keys are generated in a secure environment and are used for both encryption and decryption of data. In some embodiments, the private encryption key is stored on the messaging server 132 (for example, in the user's mailbox) and on the mobile communication device 201. In other embodiments, the private encryption key maintained by the mobile data server 120 is stored elsewhere. The location of the private encryption key maintained by the mobile data server 120 typically depends on the messaging server 132 which is used. The private encryption key may, in some embodiments, be regenerated by the user on mobile communication devices 201. Data sent to the mobile communication devices 201 is encrypted by the mobile data server 120 using the private encryption key retrieved from the user's mailbox or other location. The encrypted data, when received on the mobile communication devices 201, is decrypted using the private encryption key stored in memory. Similarly, data sent to the mobile data server 120 from the mobile communication devices 201 is encrypted using the private encryption key stored in the memory of the mobile communication device 201. The encrypted data, when received on the mobile data server 120, is decrypted using the private encryption key retrieved from the user's mailbox or other location. Tyhurst at [0029]:

'117 Claims	Exemplary Disclosures
	The mobile data server 120 may also comprise an Internet connectivity module (not shown) which provides TCP/IP (transmission control protocol/Internet protocol) and HTTP (hypertext transfer protocol)-based connectivity providing an Internet based service connection. The Internet connectivity module provides the mobile communication devices 102 with access to the Internet 112 and World Wide Web (WWW) and possibly other external communication networks connected directly or indirectly to the mobile data server 120. Alternatively, in other embodiments, the Internet connectivity module may be implemented by a separate server or server application which is connected to the mobile data server 120. Tyhurst at [0058]: A predetermined set of applications that control basic device operations, including data and possibly voice communication applications will normally be installed on the mobile communication device 201 during or after manufacture. Additional applications and/or upgrades to the operating system 254 or software applications 258 may also be loaded onto the mobile communication device 201 through the wireless network 101, the auxiliary I/O subsystem 250, the serial data port 252, the short-range communication subsystem 262, or other suitable subsystem 264 other wireless communication interfaces. The downloaded programs or code modules may be permanently installed, for example, written into the program memory (i.e. the flash memory 244), or written into and executed from the RAM 246 for execution by the microprocessor 240 at runtime. Such flexibility in application installation increases the functionality of the mobile communication device 201 and may provide enhanced on-device functions, communication-related functions, or both. For example, secure communication applications may enable electronic commerce functions and other such financial transactions to be performed using the mobile communication device 201. Tyhurst at [0068]:

'117 Claims	Exemplary Disclosures
	The device manager 314 is a program or program module which manages application-level wireline access to connected mobile devices 201, and/or possibly Bluetooth or other short-range wireless access to connected mobile devices 201. The device manager 314 provides a service routing application programming interface (API) which provides applications such as the Web-based device management application 302 with the ability to route traffic through the serial data connection (for example, USB) or Bluetooth® connection to the computer 117 using standard connectivity protocols. When a user connects their mobile communication device 201 to the computer 117 via a USB cable, the mobile data server 120 automatically routes traffic that was destined for the wireless network 101 to mobile communication device 201 over the network 124 and through the mobile data administration server 122 using the USB cable connecting the mobile communication device 201 to the computer 117. Similarly, any traffic destined for the wireless network 101 will be automatically sent over the USB cable to the mobile data server 120 for processing. This is sometimes referred to as least cost routing. In some embodiments, the user has the ability to automatically route traffic over a Bluetooth® connection to their computer 117 rather than relying on USB connectivity. All traffic to and from the mobile communication device 201 is routed over the network 124 (which may be, for example, a wired network), saving the user from utilizing additional bandwidth from their wireless service provider. When least cost routing is used on the mobile communication device 201, it is performed automatically by the low-level protocol layers and does not require any changes to an application for it to utilize this capability. When connected via a Bluetooth® connection, the computer software generally handles a Bluetooth® connection like a wireline connection. Cohen at [0033]: In the following description, the principles of the present invention are described using an example embodiment in which the notification is a push message.

'117 Claims	Exemplary Disclosures
	However, it will be apparent to those of ordinary skill in the art, after having reviewed this description, that the principles of the present invention may apply to notification other than push messages. In this description and in the claims, a "notification" refers to any message that is electronically communicated between two end-points, regardless of the form and content of the message, and regardless of the protocol used to communicate the message. Thus, e-mails, instant messages, or any other electronic messages may be considered notifications within the scope of this definition. Cohen at [0042]: The wireless device 403 securely browses to the initial browse point using, for example, the Secure Socket Layer (also called "SSL") protocol. Using this protocol, the wireless device 403 provides a Security Parameter Index (also called "SPI") that may be used to identify a notification session. The SPI is, for example, a 32-bit value that is unique to the wireless device 403, and that may serve as a handle into the appropriate security context for the notification session. The wireless device 403 may also suggest other security information such as the master secret, the cryptographic algorithm (e.g., NULL for none, DES, 3DES, RC5 or the like), the key length, the hash algorithm (e.g., NULL, MD5-HMAC, SHA_HMAC, MD5_RCA, SHA_RCA or the like), the session expiration (in time and/or in usage), and potentially an exportable keys flag that indicates whether or not this security information is exportable. The wireless device 403 may suggest these parameters in a GET method of a HyperText Transport Protocol (HTTP) request with the parameters includes as arguments in the method. The initial browse point of the notification source 401 may then response with an acceptance or a rejection of these parameters. A rejection may also include a suggestion for alternative values. Ultimately, the required security information for the session is agreed upon. Cohen at [0045]:

'117 Claims	Exemplary Disclosures
	Although negotiating security information "out-of-band from the notification mechanism" has been described with respect to negotiating the security information within a secure SSL session, other forms of negotiating out-of-band from the notification mechanism may be performed so long as the message transit points do not become apprised of the security information. For example, if the notification sink is a wireless device that has a docking station that connects to a desktop computer, a docked wireless device may instruct the desktop computer to use its modem to establish a telephone call with the notification source using a wired connection to negotiate the security information. The security information may then be passed to the wireless device via the docking station. Cohen at claim 15: A method in accordance with claim 14, wherein the act of establishing a secure session between the notification source and the notification sink comprises the following: an act of establishing a Secure Socket Layer (SSL) session between the notification source and the notification sink. Cohen at [0048]: The notification source 401 then initiates transmission of the push message 500 to the notification sink 403 (act 304) via the at least one message transit point 404. Although the push message 500 traverses through the at least one transit points 404, the notification message portion of the push message is encrypted. Since the at least one message transit point 404 is not privy to the security information originally negotiated in act 302, the at least one message transit point 404 is not able to decrypt the notification message 502. Once the notification sink 403 receives the push message 500 (act 305), the notification sink 403 decrypts the encrypted notification message 502 using the security information previously

'117 Claims	Exemplary Disclosures
	negotiated (security information 421) along with potentially the supplemental security information 501 (act 306). Thus end-to-end security is provided since only the notification source 401 and the notification sink 403 are able to access the clear text content of the notification message. Cohen at [0072]: The encrypted payload data field 704 is a variable length field that actually contains the encrypted form of the notification, as well as potentially some encrypted header fields. For example, the field 704 may include a content type field as well as a Wireless Application Protocol (also called "WAP") application ID field. The content type field may identify the presence of XML in the multipart message, while the WAP application ID field identifies the application the device should call to process the push message. The padding field 705 ranges from zero to 255 bytes and is often used when cryptographic algorithms require the cipher text to be a certain length or, regardless of the cryptographic algorithm, padding is needed to terminate the cipher text on a 4-byte boundary. The padding length field 706 is an 8-bit value that specifies the length of the padding field 704 in bytes. The payload ID field 707 (also called the "next field" in the ESP standard) identifies the type of data included in the payload data field 703. The payload data field 704, the padding field 705, the padding length field 706, and the payload ID field 707 are all encrypted. The payload data field 704 represents an example of the notification message 502 of the push message 500. ESP object 700 may also include authentication data field 708 that includes a digital signature for authenticating the push initiator 601. Cohen at [0081]: The computer 820 may operate in a networked environment using logical connections to one or more remote computers, such as remote computers 849 a and 849 b. Remote computers 849 a and 849 b may each be another personal computer,

'117 Claims	Exemplary Disclosures
	a server, a router, a network PC, a peer device or other common network node, and typically include many or all of the elements described above relative to the computer 820, although only memory storage devices 850 a and 850 b and their associated application programs 836 a and 836 b have been illustrated in FIG. 8. The logical connections depicted in FIG. 8 include a local area network (LAN) 851 and a wide area network (WAN) 852 that are presented here by way of example and not limitation. Such networking environments are commonplace in office-wide or enterprise-wide computer networks, intranets and the Internet. Cohen at [0082]: When used in a LAN networking environment, the computer 820 is connected to the local network 851 through a network interface or adapter 853. When used in a WAN networking environment, the computer 820 may include a modem 854, a wireless link, or other means for establishing communications over the wide area network 852, such as the Internet. The modem 854, which may be internal or external, is connected to the system bus 823 via the serial port interface 846. In a networked environment, program modules depicted relative to the computer 820, or portions thereof, may be stored in the remote memory storage device. It will be appreciated that the network connections shown are exemplary and other means of establishing communications over wide area network 852 may be used. Lonnfors at [0023] An example of a target device that utilizes concepts according to embodiments of the present invention is illustrated as the mobile phone 118B. The device 118B includes, for example, a radio transceiver 134 and hardware (including the processor) coupled to an operating system (OS) 130. The present invention may include a dispatcher 132 implemented as firmware, a module, or a program running

'117 Claims	Exemplary Disclosures
	on the OS 130. The dispatcher 132 can be used in any type of OS 130, including various versions of Windows®, Linux, Unix®, PalmOS®, Symbian OS, etc. Lonnfors at [0031] A diagram of a device 200 embodying concepts of the present invention is shown in FIG. 2. The device includes a network interface 202 for communicating over a network 204. The device 200 is enabled to run multiple applications 206, 207, 208, 209 that are able to receive incoming data from the network interface 202. The applications 206, 207, 208, 209 may provide any user or system function on the device 200. For example, applications can be used for receiving and presenting data for users, such as in a multimedia communication session. The application may use incoming data for system tasks, such as updating system time or network parameters. Novack at [0015]: "According to an aspect of the present invention, a method is provided for securing inter-process communications between a first process and a second process. The method includes receiving, from a device that supports the first process, a request to initiate inter-process communications. The method also includes replying to the device with a request for information of a digital certificate that authenticates the identity of the first process." Novack at [0021]: "According to an aspect of the present invention, a computer-readable medium is provides that stores a program that secures inter-process communications between a first process and a second process. The computer-readable medium includes a receiving code segment that receives, from a device that supports the first process, a request to initiate inter-process communications. The computer-readable medium also includes a digital certificate requesting code segment that replies to the device

'117 Claims	Exemplary Disclosures
	with a request for information of a digital certificate that authenticates the identity of the first process." *See also* Novack at [0027]. Novack at [0035]: "In one example, the first process is running on an application server that provides application services over the internet to requesting clients. The application server interacts with a management server that manages multiple databases as a back-end service. The first process may be run when a user contacts an application server and downloads presentation layer elements of an application to interact with the application server. The application includes instructions to invoke additional processes for predetermined user requests. Accordingly, the application will request the services of a second process at S110." Novack at [0044]: "FIG. 2 shows an exemplary network architecture for secure inter-process communications. As shown, a **computer** 201 communicates through a **network** 220 with an **application server** 230. As an example, the **application server** 230 may be a web server that provides internet applications for users. The user downloads a web page onto the **computer** 201 when the user contacts an internet address corresponding to the desired internet application. Typically, the web page is accessed and downloaded using hyper text transport protocol (HTTP)." Novack at [0045]: "The **application server** 230 communicates through another **network** 240 with **management server** 250. As an example, the **network** 240 may be a secure local area network that enables the **application**

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 109 of 442

'117 Claims	Exemplary Disclosures
	server 230 to obtain back-end processing. Of course, the network 240 may be the same network as the network 220." Novack at [0046]: "The management server 250 manages databases 254 and 256. The databases 254 and 256 store data associated with the service provided by the application server 230. As an example, the application server 230 may provide internet services for accounts of a service provider. Thus, the application server 230 may provide services for a bank, a utility, or a content provider such as a news service. While the user of the computer 201 is intentionally provided with direct access to the application server 230, the configuration of the network shown in FIG. 2 is intended to help deny the user of the computer 201 direct access to the management server 250 and the databases 254, 256. Further, to ensure that the communications between the application server 230 and the management server 250 are secure, a digital certificate is installed on the application server 230, and the digital certificate is bound to the application server 230 and/or a process running on the application server 230." Novack at [0047]: "In an embodiment, the application server 230 and the management server 250 communicate using the secure sockets layer (SSL) protocol (i.e., HTTPS), for secure and trustworthy communications. A secure sockets layer session begins with an exchange of messages known as a handshake. In the embodiment of FIG. 2, the handshake occurs between the application server 230 and the management server 250. The handshake includes the application server 230 contacting the management server 250. In response to the application server 230 contacting the management server 250, the management server 250 sends the application server 230 a digital certificate. The application server 230 responds by providing the management server 250 with a digital certificate. Of course, an exchange of two

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 110 of 442

'117 Claims	Exemplary Disclosures
	digital certificates does not need to occur in every case; rather, a single digital certificate may be used when only one party (e.g., the application server 230) to a communications session needs to be authenticated." Novack at [0048]: "The **management server** 250 may require authentication of the **application server** 230 for each instance of inter-process communications. In this regard, even if the management server's 250 only function is to process requests for the **application server** 230, the **management server** 250 may need to authenticate the application server for each request for inter-process communications when, for example, the **application server** 230 widely interacts with the public and is therefore subject to unknown risks. The **management server** 250 may request verification of the application server's 230 address, the particular requesting function, and any other information that will help authenticate the request. Of course, the **management server** 250 may demand a digital certificate that is bound to the **application server** 230 or even a particular process that runs on the **application server** 230." Novack at [0049]: "Digital certificates are provided to the **application server** 230 and/or the **management server** 250 by a certification authority upon the request and verification of, e.g., the operator of the **application server** 230 and the **management server** 250. However, the digital certificates may also be bound to one or more individual processes on the **application server** 230 or the **management server** 250, so that inter-process communications may be authenticated as originating from an authorized process, and not just a predetermined device or user. Alternatively, the digital certificate may be used as appropriate to secure different types of communications, including inter-process communications." Novack at [0064]:

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 111 of 442

'117 Claims	Exemplary Disclosures
	"The device in FIG. 4 may include multiple processes that communicate with each other using the secure inter-process communications of the present invention. Additionally, although the terms "client" and "server" are used to refer to devices, the devices may be defined and distinguished by a set of software instructions installed as a pre-packaged program. Further, while examples of the uses of the present invention include secure client/server networks and peer-to-peer networks, secure inter-process communications can be used to enhance security in any inter-process communications environment. Further, while the embodiment shown in FIG. 2 only shows a single computer 201 and a single application server 230, secure inter-process communications can be used to enhance security for multiple application servers 230 belonging to a single entity or multiple users using computers 201. Additionally, while networks 220 and 240 are separately designated in FIG. 2, the networks may be the same, such as an internet protocol (IP) network. Further, while a certification authority is discussed in the context of the present invention, a certification authority is not shown in the figures. In this regard, a certification authority may communicate over the network 220 and/or the network 240 to provide digital certificates to the application server 230 and the management server 250. Of course, the digital certificates may be provided by the certification authority as portable memory modules that are installed on the application server 230 or the management server 250. Furthermore, the entity that provides the application server 230 and/or the management server 250 may also act as a certification authority for its customers using the computers 102." Santamaria at [0005]-[0006]: "Users of a mobile device (e.g., laptop, palmtop, portable media player, smartphone, multimedia mobile phone, mobile gaming systems such as a "Gameboy", etc.) may subscribe to one or more notification services. For example, users may subscribe to one or more "push" email services such as .Mac, Microsoft Exchange ActiveSync, push-IMAP, Yahoo! Push, etc. In the case of a push email

'117 Claims	Exemplary Disclosures
	service, for example, the email server may automatically transmit email messages and/or calendar updates to the email client on the mobile device for the user without the user requesting the item. In other words, the user does not request (poll) the email server for email messages or other items of interest. Thus, a push notification service is a persistent notification service. In addition, other types of services may use a similar push architecture (e.g., update/upgrade services, news services, weblog services, podcast services, etc.). In order to maintain a push notification service, the mobile device periodically refreshes the connection to the push notification service (e.g., by transmitting a ping message to the push server). "In addition, users may subscribe to one or more "pull" services, such as "pull" email services (e.g., IMAP, POP3). In a pull email service, a user periodically checks (polls) the email server to determine if there is new email messages. If there is new email messages, they are then downloaded to the client. Many email clients support an automatic configuration of a poll interval. For example, a user of the mobile device may configure a poll interval of 10 minutes for a POP3 email account (thus the email client automatically polls the email server every 10 minutes to check for new email messages). In addition, notification services may be configured to provide network wide (e.g., Internet wide) event notification messages to multiple subscribers, where notification services may be automatically discovered by one or more subscribers and/or publishers, which is described in U.S. patent application Ser. No. 12/042,307, entitled "Automatic Notification System and Process", filed on Mar. 4, 2008, which is hereby incorporated by reference in its entirety." Santamaria at [0025]-[0027]: "FIG. 1 illustrates an exemplary computing environment 100 according to one embodiment of the invention. The computing environment 100 includes the mobile devices 160 and 170 which are each coupled with one or more network data access elements 150. The network data access element(s) 150 may be part of various types

'117 Claims	Exemplary Disclosures
	of networks in different embodiments of the invention, including cellular networks (e.g., Global System for Mobile Communications (GSM), Enhanced Data Rates for GSM Evolution (EDGE), General Packet Radio Service (GPRS), Code Division Multiple Access (CDMA), Universal Mobile Telecommunications System (UMTS), or other cellular networks), Local Area Networks (LANs), etc. For example, if the network data access element(s) 150 are part of a GSM network, the network data access element(s) 150 may include one or more of base transceiver stations (BTSs), Base Station Controllers (BSCs), Mobile Switching Centers (MSCs), Serving GPRS Support Nodes (SGSNs), etc. As another example, if the network data access element(s) 150 are part of a LAN, the network data access element(s) 150 may include one or more network switches, routers, hubs, modems, etc. "The mobile device 160 includes the application processor 162 and the network connectivity element(s) 164 (e.g., wireless network connectivity elements (e.g., radio network connectivity elements for cellular telephony, WiFi network connectivity elements), wired network connectivity elements (e.g., wired network interface controller(s) (NICs))). The mobile device 160 communicates with the network data access element(s) 150 via the network connectivity element(s) 164, through a data context (e.g., a PDP context). The Wide Area Network (WAN) 140 (e.g., the Internet) is coupled with the network data access elements 150. The push notification services 110 and 120 and the pull notification service 130 are coupled with the WAN 140. According to one embodiment of the invention, a user of the mobile device 160 accesses and uses the push notification services 110 and 120 and the pull notification service 130 (the push notification services 110 and 120 and the pull notification service 130 provide items of interest for the user). For example, the push notification service 110 may be providing a personal push email account for the user of the mobile device 160 while the push notification service 120 may be providing a work push email account for the user of the mobile device 160. As

'117 Claims	Exemplary Disclosures
	another example, the pull notification service 130 may be providing an organizational pull email account (e.g., POP3 email account, webmail account, etc.) for the user of the mobile device 160. "In one embodiment of the invention, the application processor 162 is the central processor of the mobile device 160. For example, the application processor 162 processes the operating system(s) of the mobile device 160 (the operating system is not illustrated in FIG. 1 in order not to obscure understanding of the invention), and any applications running on the mobile device 160. Other well known features of mobile devices are not illustrated in FIG. 1 in order not to obscure understanding of the invention, including input/output devices, memory(ies), power supply, displays, etc." Isomursu at 2:56-3:33: "The intention is that any user messages can be used, but in following mainly short messages will be referred to as an example. In addition to being fast to send, the advantages of a short message service can be utilised, such as not reserving the voice channel. Application related information can either be pre-stored in a terminal memory (permanent memory) or a user may store the application related information in a terminal memory (cache memory) by contacting a server by means of a terminal. Depending on the application, the user may enter user input or modify the information in the applications. In another application the information relating to an application may be sent by a service provider and the information may be such that it is not possible for the user to modify it, only to request the service provider to modify it. The information readily printed in the application can also be transmitted. An application type identifier or header is preferably added to the transmission, so that a receiving terminal identifies the short message as not an ordinary short message, but as a short message containing information relating to and intended for a specific application. The identifier can be a code in an address or a control field of

'117 Claims	Exemplary Disclosures
	the short message, or it can be a code in the message part of the short message. Because it has been realised that the short message service, already existing in the mobile station system, can be utilised for sending information on applications, the advantages are, e.g., that there is no need to establish an off-line server for sending the application related information, such as, for example, in the system presented in Publication WO 94/23394. Especially advantageous is that one and the same server, i.e. the SMS server (the Short Message Service Centre SM-SC) can be used for sending and forwarding information relating to any application, so there is no need to have separate servers for each application. The SMS server will forward any short message and the terminal will address the information to the correct application according to the header or identifier in the message. And since a short message can be sent simultaneously with a circuit-coupled connection, the sending of the application related information does not engage the terminal's communication line, e.g., in case of a simultaneously incoming call. A network like the GSM network is maintained by several operators and usually each operator has at least one SMS server of their own. In this case naturally any SMS server or several servers may be used for the invention." Isomursu at 5:33-51: "Additionally short messages may be sent by a personal computer PC. In this case the mobile services switching centre MSC has a connection to a server GTW (Gateway), which has a connection to e.g. Internet. Thus a computer PC having a connection to Internet (or directly to the server GTW as shown by the broken line) may fetch a WWW (World Wide Web) page from Internet, which physically can be found e.g. from the server GTW. Optionally a service provider or operator may have a separate server GTW having a connection to the mobile services switching centre MSC for sending short messages or other user messages. When using such a WWW page for sending a short message the user inputs the connection information (e.g. phone number) of the receiving terminal MS2 and the message to be sent, whereby

'117 Claims	Exemplary Disclosures
	the message is transferred via Internet and the server GTW to the mobile services switching centre MSC and further to the short message service centre SM-SC from which the message is directed to the receiving terminal MS2 via the mobile network." Isomursu at 9:4-46: "By means of an application 'Service DTMF Commands', information can be received, e.g., from a network operator in order to utilise services provided by the operator. An application identifier can be 'Service DTMF Commands', and it may have fields for a sender's name, a DTMF command, and an explanation field. The command can be, e.g., a password, a user identifier, or something else related to the services provided by the operator. A user may use the received command, e.g., a password, when utilising the offered services, in which case, the user does not have to input the password through the keyboard, because the password can be obtained directly from the application (or the information in the application) in question. Other applications to which information may be received from a network operator are any applications which require some settings in the terminal before an application can be used. An example is an 'Internet Access Point' application which contains information necessary for the terminal to use a WWW browser for example. This information may be provider name, phone number, modem initialisation information, server information, IP address (Internet Protocol). Another example of an application that needs some settings is a Fax Box application. A faxbox is a service implemented e.g. in a server, which receives all faxes for a specific user. The user can then retrieve the faxes from the faxbox. For retrieving faxes the terminal needs to have the contact information to the server. This could be implemented so that when the terminal has received a new fax to the faxbox, the terminal receives a notification in a user message of a received fax, but additionally the user message would include a UID information (Unique Identifier), i.e. contact information to the faxbox server and other necessary information, such as filename of the fax and a password needed. The

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 117 of 442

'117 Claims	Exemplary Disclosures
	Fax Box application could function either manually, i.e. by user execution, or automatically so that the terminal contacts the faxbox server automatically for retrieving a received fax according to the identifier information contained in the notifying user message. The contact can be made as fax call to retrieve the fax. Optionally it could be a user message sending the identifier information back to the server and additionally the fax connection number of the terminal so that the server then can contact the terminal for sending the received fax to it via a fax call." Isomursu at 14:47-15:6: "An example of creating a menu for ringing tones is disclosed in FIG. 10 as a sequence of displays to illustrate what the user sees on the display. The command "NEW RINGING TONES" is sent in a user message to a server of a ringing tone service provider in order to request for latest ringing tones. As a response the server sends a user message containing information relating to the Menu applications for creating a menu, from which the user can select a new ringing tone. The user selects the desirable ringing tone from the menu (selects ring tone 'Popcorn'). The selection activates a user message to be sent to the server indicating the desired ring tone. After a while the terminal receives the ring tone from the server. A received ringing tone is indicated to the user using the "RINGING TONE RECEIVED" notification. The user can accept or reject the ring tone. Once the user has given acceptance, the selection menu with the options "Save" and "Playback" displays. If the user selects the Save-option the received ringing tone is saved to the phone and it appears to a ringing tone options menu. The saving is indicated to the user by displaying a "SAVED" confirmation note after which the phone goes to the idle state. If the user selects the Playback—option the received ringing tone is played to the user and after that the original selection list displays again. If the user gives a rejection of the new ring tone, the received ringing tone is discarded, the selection list is removed from the screen, and the phone goes to the idle state."

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 118 of 442

'117 Claims	Exemplary Disclosures
	Isomursu at 18:60-19:39: "Sending information of a service page in a short message according to the present invention saves considerably the power consumption of the terminal and, thus, prolongs its useful life, which is an important objective in battery-operated devices. In addition, savings are made in phone call expenses, when query information can be sent together with the short message. The whole system has been illustrated in more detail in FIG. 8. A terminal according to the present invention has been presented with reference numeral 1, by means of which a communications link (reference 101) to the Internet INT can be established. The invention could also be implemented by means of a device having means, according to the present invention, for processing information on an application into a short message, which is sent through an ordinary mobile station by coupling the device to the mobile station. Such a device could be, e.g., a computer. To simplify matters, only a communicator will be discussed here as an example of a terminal according to the invention. When the communications link is established to the Internet by means of a communicator 1, a service page of a provider of services can be downloaded (reference 102) from the Internet into its memory and user interface. By means of solutions known for prior art, the user, after having filled in the page, has sent the service page by means of the communicator 1 to a server SERV of the provider of services, i.e., along route 101–101′. This means that the communications link to the Internet should be open for transmission. In a system according to the present invention, the information on the service page is sent (reference 103) in a short message to a short message service centre SM-SC, which sends it further (reference 104) to the server SERV of the provider of services. The transmission of the service page through the Internet, according to prior art, lasts considerably longer than the transmission of a short message and, thus, due to the present invention, a shorter transmission time is achieved, thereby, effecting a saving in power, since, in the communicator, transmission and reception, in particular, consume a lot of power compared to other functions. In addition, in the solution

'117 Claims	Exemplary Disclosures
	according to prior art, a circuit-coupled connection is in use in transmission, in which case the communicator is engaged during transmission. On the other hand, the sending of a short message does not engage the circuit-coupled connection, and an additional advantage is that the short message service centre SM-SC will send the message to a receiver later, if the telephone number of the provider of services happens to be unreachable during the transmission of the message." Isomursu at 20:48-21:20: "In FIG. 9, there is a block diagram of the implementation of a terminal according to the present invention, which in the figure is shown as a device also having means for transmission over the air such as a mobile phone. However a similar implementation can be used for a terminal device of a service provider which usually does not have means for direct radio frequency transmission but is connected to such means (e.g. a base station) over the network as is shown in FIG. 2 (e.g. the personal computer PC or server GTW). The terminal is preferably a mobile phone or communicator, which has circuits and a user interface that enable the processing of different applications. The terminal 1 in FIG. 9 comprises, for communication using radio communication, a radio unit RU (the reference has not been marked in the figure), which comprises a transmitter branch 2, known from an ordinary mobile station, (comprising blocks implementing coding, interleaving, ciphering, modulating, and transmitting), a receiving branch 3 (comprising receiving, de-modulating, de-ciphering, de-interleaving, and implementing blocks) and, for transmission using radio communication, a duplex filter 4 that distinguishes between a received and transmitted message, as well as an antenna 5. The terminal has a main control circuit 6 that controls its operation. Furthermore, the main control circuit 6 comprises still a RU controller 7 that carries out control functions of an ordinary mobile station. In addition, the terminal main control circuit 6 comprises blocks 8–12 for carrying out the functions of a data processing unit of the terminal and for sending application related information as a short

'117 Claims	Exemplary Disclosures
	message according to the present invention. Thus, the blocks 8–12 can be said to form a data processing unit DU of the terminal. The controls of the radio unit RU and the terminal's data processing unit DU do not have to be integrated into the main control circuit but, instead, they could also be implemented apart from each other, so that the RU control circuit 7 is on the radio unit's side, and on the data processing unit's side, there is the DU processor 8, which is in connection with the RU control circuit 7 for establishing communication between the radio unit and the data processing unit." Okonnen at [0011]: "In an embodiment according to the present invention, the network may further comprise at least one of an update server, and a plurality of generators. The generators may be adapted to generate updates able to be processed by at least one provisioned update agent in electronic device. The update server may be adapted to store updates accessible by the plurality of servers. The plurality of servers may be communicatively coupled to the update server." Okonnen at [0013]-[0014]: "In an embodiment according to the present invention, the number assignment module programming activity may comprise at least one of over-the-air service provisioning (OTASP) activity and over-the-air parameter administration (OTAPA) activity. "In an embodiment according to the present invention, the provisioned data unit may be adapted to store at least one of update agent related provisioning information, a universal resource locator of a server used to retrieve updates, and a security key used to authenticate server messages." Okonnen at [0020]:

'117 Claims	Exemplary Disclosures
	"In an embodiment according to the present invention, the network may further comprise update agent provisioning information stored in the electronic device. The update agent provisioning information may comprise a device server URL, an index of provisioned update agents, a security key, and electronic device related information. The device server URL may provide references to servers hosting updates to be downloaded. The updates may be compatible with update agents currently available and provisioned in the electronic device." Okonnen at [0021]: "In an embodiment according to the present invention, the index of provisioned update agents may provide an index value used to compute an address location of a provisioned update agent. The index of provisioned update agents may be an index to a table containing an address for an update agent in non-volatile memory the electronic device." Okonnen at [0052]: "The provisioned data 129 may also be used to store update agent related provisioning information, such as a universal resource locator (URL) of a device server or a management server that may be used to retrieve updates, a security key(s) that may be used to authenticate messages to/from the device server, and selective encryption/decryption of the messages, etc. Each update agent, such as update agent A 113 and update agent B 115, may have corresponding entries in provisioned data 129." Okonnen at [0060]: "In an embodiment according to the present invention, update agent A 113 may be employed to update firmware 117 and operating system 119 during reboot of the mobile handset 107. After a reboot update, update agent B 115 may be used to update other components in the mobile handset 107, for example, configuration

'117 Claims	Exemplary Disclosures
	parameters, gaming engines, third-party software, etc., that may not be updated during reboot of the mobile handset 107. Both of update agent A 113 and update agent B 115 may be provisioned with security keys for secure communications with device servers 135 and 137 during update agent provisioning activity, such as provisioning activity accompanying NAM provisioning activity and/or incorporated into NAM programming activity." Okonnen at [0066]: "In an embodiment according to the present invention, a separate key may be employed to authenticate updates by a download agent/download client, (e.g., a browser), and by the update agent. In an embodiment according to the present invention, the security key(s) 213 may also incorporate keys employed for secure communications with external systems, such as keys used for encryption and decryption of data and messages during communications with external systems, such as a device server." Hashimoto at [0031]: "In order to solve the second problem, it is made possible to carry out the decryption by using keys that are different for different blocks according to a small amount of secrets." Hashimoto at [0032]: "In order to realize this, the microprocessor is provided with the following. (a) a processor core that issues a reading request by specifying an address of the external memory that stores a plurality of programs that are respectively encrypted according to the unique program keys: (b) an interface for reading a data block of the specified address of the external memory in response to the reading request of the processor core;

'117 Claims	Exemplary Disclosures
	(c) a task key table for maintaining the program key in correspondence to the task identifier assigned to each task which is a unit of the program execution control; (d) an offset table for maintaining a top address at which the program is arranged as an offset value, for each task identifier; (e) a key generation unit for calculating a relative address value from the address specified by the reading request and the offset value specified by the task identifier of the offset table, and generating a block corresponding key in which the relative address value is encrypted by using the program key; (f) an encryption processing unit for decrypting the read out data block by using the block corresponding key; and (g) a cache memory into which the decrypted data block is read in units of cache lines." Hashimoto at [0038]-[0039]: "Fig. 1 shows a multi-party application protection environment to which the present invention is to be applied. "A software vendor 121 produces a program 122, and distributes it to a microprocessor-A 103 provided inside a target system 102 of a system user 112. In order to protect the program to be distributed, the plaintext program 122 is encrypted." Hashimoto at [0040]: "More specifically, the program formed by execution codes and data such as initialization data is encrypted by using a program key selected by the software vendor 121, and the program key used for the encryption is encrypted by using a

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 124 of 442

'117 Claims	Exemplary Disclosures
	public key-A of the processor-A of the target system 102 to generate a distribution key." *See also* Hashimoto at Fig. 1. Fahrny at [0008]: "According to the present invention, a controller for managing media stream decryption keys is provided. The controller comprises a media decryption engine, a table, and a content key list. The media decryption engine generally receives an encrypted media stream from a headend and presents a decrypted media stream to a receiving device in response to a decryption key. The decryption key is generally a function of a content key. The table may contain a content key and index and a plurality of corresponding content keys. Content keys that correspond to a particular encrypted media stream may be selected from the content key list using an index from the content key table, and that is referenced by an identifier received from a headend in connection with the encrypted media stream. The table may optionally (i.e., alternatively) contain initialization vector (IV) values that may be indexed and selected." Fahrny at [0023]: "Keylist: A list of decoder addresses and respective decoder keys in ordered pairs. Keylists may be used by the Uplink Control System (UCS) for generation of authorization messages that are addressed to the diagnostic circuit that is embedded in decoders that are specific to the encoder system." Fahrny at [0037]-[0038]: "For VOD services, a table of keys can be generated and delivered at session setup time. The keys for VOD service may be delivered with synchronization information related to key change as well as other information for short term working key epochs. A VOD Session ID or, alternatively, a Program ID may be

'117 Claims	Exemplary Disclosures
	used as an index to reference the keys list with the appropriate record of information for the VOD transport decryption. In alternative embodiments of reduced hierarchy key management of the present invention, one-way hashing may be implemented in the protection, selection and processing of the decryption key. "The reduced hierarchy key management of the present invention generally provides a new, more secure, and elegant system and method to deliver content keys for decrypting the program streams in conditional access systems (e.g., Broadcast and Video On Demand applications). The key management of the present invention may dramatically reduce the complexity required to deliver new content keys when a first Entitlement Message has been sent (i.e., presented, transmitted, provided, broadcast, etc.) to each set top box. The reduced hierarchy key management system and method of the present invention may be implemented as a portion of a new CAS system. The new CAS system generally provides for the manufacture and distribution of devices that are compatible with infrastructure, regardless of specific content security mechanisms that are used in that infrastructure. The new CAS system may provide far more efficient manufacturing, distribution and operations, and in fact enable new business models, including the retail availability of extremely low cost customer premises equipment (CPE) when compared to conventional approaches." Fahrny at [0041]: "Referring to FIG. 1 a, a diagram illustrating a media decoder (i.e., controller, processor, apparatus, circuit, device, etc.) 100 of the present invention is shown. The decoder 100 may be implemented in connection with a digital media stream distribution system (described in more detail in connection with FIGS. 2(a-b)). The controller 100 is generally implemented as a security processor (or processing system) that provides at least one security feature (e.g., encryption, decryption, authentication, security key management, copy protection, digital rights management, etc.) to at least one digital media input/output stream. The decoder

'117 Claims	Exemplary Disclosures
	100 generally has an input 102 that receives at least one signal (e.g., VIDIN and PID), an input/output 104 that receives/presents a signal (e.g., TFHE) as well as additional data, signals, messages, and the like, an input 106 that receives a working key modifier and application function signal (e.g., WKM), and an output 108 that presents a signal (e.g., VIDOUT)." Fahrny at [0052]: "Referring to FIG. 1 c, a diagram illustrating an alternative media decoder (i.e., controller, processor, apparatus, circuit, device, etc.) 100" of the present invention is shown. The decoder 100" may be advantageously implemented in connection with video on demand (VOD) key management. The media stream VIDIN may be an encrypted VOD media stream. The media stream VIDOUT may be clear VOD media stream. The input 102 may receive the media stream VIDIN. In one example, the input 102 may receive the identifier PID. In another example, the input 102 may receive a VOD session identifier (e.g., VODID). The decoder 100" generally does not receive the working key modifier WKM." *See also* Fahrny at Fig. 1a. Fan at 10: FIG. 6 is a schematic structural diagram of a first terminal according to an embodiment of the present disclosure, where the terminal includes: a policy setting module, a policy storage module, and a Push message processing module. a policy setting module, configured to set a Push message control policy including a matching condition and a corresponding processing manner; The policy storage module is configured to store a Push message control policy stored by the policy setting module. Fan at 10:

'117 Claims	Exemplary Disclosures
	The first terminal provided by the embodiment of the present invention further includes an application module, configured to perform corresponding service processing according to the Push message. Based on this situation, the Push message processing module includes: a processing execution module and an analysis control module. Fan at 12: The matching conditions in the set Push message control policy may include the following: user identifier, maximum message length threshold, service control, content control, and control of the trusted PI. The following five Push message control strategies are introduced separately. Fan at 13: First, the user identifier The purpose of setting the matching condition is to determine, according to the user identifier, that the terminal to which the Push message is to be sent is the setting terminal of the Push message control policy when the Push sending agent performs the Push message control. analysis control module. Assmann at 1:39-55: "Today, electronic mail or "e-mail" is a pervasive, if not the most predominant, form of electronic communication. FIG. 1 illustrates the basic architecture of a typical electronic mail system 10. At a high level, the system includes a mail server connected over a network to various e-mail "clients," that is, the individual users of the system. More specifically, the system 10 includes one or more clients 11 connected over a network to at least one SMTP (Simple Mail Transport Protocol) server or "Message Transfer Agent" (MTA) 12 a for routing e-mail. Users write, send, and read e-mail via Mail User Agents (MUA), such as Microsoft Outlook™

'117 Claims	Exemplary Disclosures
	or mutt, present at each client (computer). To send e-mail, an MUA connects to an MTA which receives the e-mail and routes it to another MTA. An intermediary MTA might forward the e-mail to yet another MTA until the e-mail reaches the destination system, where the e-mail is stored in a mailbox accessible by the recipient." Assmann at 4:37-58: "A method of the present invention for sending an e-mail message using a secured connection that employs encryption includes the following method steps. A client's request (e.g., Mail User Agent's request) is received at a Message Transfer Agent (MTA) for establishing a secured connection with the MTA for sending an e-mail message. The method attempts to authenticate the client, through use of a certificate. If the client cannot be authenticated, the method terminates without establishing the secured connection and without sending the e-mail message. On the other hand, if the client can be authenticated, the method establishes the secured connection between the client and the MTA. Additionally, the method (optionally) determines whether the encryption employed for the secured connection meets a predefined minimum encryption strength. If the encryption employed does not meet the predefined minimum encryption strength, the method terminates (including terminating the secured connection without sending the e-mail message). However, if the encryption employed does meet the predefined minimum encryption strength, the MTA will send the e-mail message (for ultimate delivery at a target destination)." Assmann at 7:62-8:19: "Methodology to Enforce Authentication or Encryption to/from an MTA or from a MUA "A. Including TLS Support within the Sendmail MTA "STARTTLS has been implemented in the sendmail MTA for Sendmail Secure Switch 2.0. It is based on existing solutions, including OpenSSL (see e.g.,

'117 Claims	Exemplary Disclosures
	OpenSSL: The open source toolkit for ssl/tls, currently available online at http://www.openssl.org/) and example programs and a patch for Postfix (see e.g., Lutz Jaenicke, Postix, 2000, currently available at http://www.aet.tu-cottbus.de/personen/jaenicke/pfixtls/). X.509 certificates tie a Distinguished Name (DN) together with a public key and those data structures are digitally signed by a Certificate Authority (CA). According to the X.509 standard, a Distinguished Name consists of several named fields, such as Country (C), State (ST), Location (L), Organization (O), and a Common Name (CN). The DN of the CA is called the cert issuer, and the DN of the cert itself is called the cert subject. The sendmail binary (i.e., "sendmail") uses a configuration file (sendmail.cf) to define several options, routing of e-mail, rewriting of addresses, and rejection or acceptance of connections and relaying attempts. The interaction of sendmail with its configuration file is defined by options, macros, classes, rulesets, and mailer definitions. For example, the decision whether an e-mail is relayed is based on a ruleset called check rcpt." Assmann at 8:34-45: "TLS only allows one to control the connection between two hosts: a server and a client. In accordance with the present invention, therefore, support is added to enforce certain restrictions on these connections, depending on whether sendmail acts as a server (receiving e-mail) or as a client (sending e-mail). In this manner, it is possible for each type of connection to enforce an authentication of the other side or at least a certain key length of the symmetric cipher used for encryption. In order to understand how that may be done in accordance with the present invention, it is first helpful to briefly review (E)SMTP, the (Extended) Simple Mail Transfer Protocol." *See also* Assmann at Figs. 1, 2, 3.

'117 Claims	Exemplary Disclosures
	White at [0035]: "Synchronization: Synchronization refers the process of data and/or message synchronization between a client and a server in which data and/or messages are downloaded (i.e., transmitted from the server to the client) and/or uploaded (i.e., transmitted from the client to the server). In the currently preferred embodiment of the present invention, a MobiLink data synchronization mechanism is used for synchronization of data and/or messages. The MobiLink synchronization system is available as a component of SQL Anywhere® Studio (available from iAnywhere Solutions, Inc. of Dublin, Calif.). MobiLink offers bidirectional exchange of information between remote databases (e.g., Adaptive Server® Anywhere or UltraLite databases available from iAnywhere Solutions, Inc.) and a variety of enterprise data sources including Adaptive Server® Anywhere, Sybase® Adaptive Server® Enterprise, Oracle, Microsoft SQL Server, IBM DB2, and other ODBC-compliant databases. Remote systems connect via standard Internet protocols such as TCP/IP, HTTP or HTTPS to the MobiLink synchronization server, which communicates with a back end (consolidated) database. MobiLink is a session-based synchronization system that allows two-way synchronization between a main database, called the consolidated database, and many remote databases. The consolidated database holds the master copy of all the data. Synchronization typically begins when a MobiLink remote site opens a connection to a MobiLink synchronization server. During synchronization, the MobiLink client at the remote site uploads database changes that were made to the remote database since the previous synchronization. On receiving this data, the MobiLink synchronization server updates the consolidated database, and then downloads changes on the consolidated database to the remote database." White at [0065]:

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 131 of 442

'117 Claims	Exemplary Disclosures
	"The present invention also provides network-independent communication. QAnywhere messages can be transported over TCP/IP, HTTP, or HTTPS protocols. They can also be delivered from a Windows CE handheld device by ActiveSync. The message itself is independent of the network protocol, and can be received by an application that communicates over a different network. The present invention also handles the challenges of wireless networks, such as slow speed, spotty geographic coverage, and dropped network connections. It protects proprietary or sensitive information by encrypting all messages sent over public networks. In its currently preferred embodiment, the system compresses and, optionally, encrypts data sent between mobile applications and enterprise servers." Holbrook at [0084]: "Messaging nodes 14 comprise a plurality of instances of messaging node 20, distributed in publicly accessible locations across a geographic region. Each messaging node 20 has access, either continuously or intermittently, to a first communications link 70 with central server 12. First communications link 70 may comprise a dial-up modem connection over telephony lines (either direct point-to-point or via ISP Internet services), Cable/Modem connection, DSL, satellite link (including full-duplex connections using geosynchronous, medium Earth orbit, or Molniya orbit spacecraft; or 'store-and-forward' services from spacecraft in low Earth orbit), radio frequency transceiver link, local area network, or other similar data exchange means known to persons skilled in the art. Individual members of messaging nodes 14 may utilize different data exchange means for first communications link 70 to central server 12." Cooley at ¶[0013]: "FIG. 1 shows one embodiment of a system, such as system 10, in which the concepts of the invention can be practiced. In FIG. 1 there is shown a link from a user device, such as cellular telephone 15 to website 11. This link extends from the user device through a carrier network, such as network 14 and via service

'117 Claims	Exemplary Disclosures
	center 12 to the desired website. The message returned from website 11 contains an HTTP request against service center 12. The service center then queues a Short Message Service (SMS) message back to device 15 through an SMS broker, such as broker 13. Broker 13 then routes an SMS message through any carrier network, such as network 14, which need not be the same network that was used for the originating message from the device." Cooley at ¶[0017]: "When the user touches tile 23-1 all the flights that the user has booked on, for example, AMERICAN™ airlines (AA) are displayed. Touching tile 24-1 then yields the screen shown in FIG. 2E which then displays the information that has been received from an external source. In this case this information would have come typically from the AA server in the manner discussed above. There could be any amount of information and tiles (or layers) to this information, all controlled by application 30 residing on the user's device with the updated information coming, from time to time, via messages from various external locations. Note that in FIG. 2F the gate and status of the flight have been updated. This update could have come as a result of a request from the user or could have come as a text message from the airline as part of an update service for customers. In any event, the message would have gone first to service center 12 and then via SMS broker 13 to the user's device where application 30 would have taken the data and placed it in the proper tile." Cooley at ¶[0020]: "Another method for doing this would be to run a Simple Mail Transfer Protocol (SMTP) gateway at service center 12. This would allow application 31 to receive incoming messages in email form. Such email messages could then be parsed to see if they contain certain information, such as the tile ID, etc. This approach would benefit from a fixed format email message. The parsing includes filtering the email

'117 Claims	Exemplary Disclosures
	messages to see if they contain certain information. The filtered information then would be sent to the device by an SMS message. Second, the formatted SMS message could be sent to the service center for delivery to the particular user device." Cooley at ¶[0024]: "FIG. 5 illustrates an example computer system 500 adapted according to one embodiment of the present invention. That is, computer system 500 comprises an example system on which embodiments of the present invention may be implemented (such as servers 11, 12, broker 13, and handheld device 15 of the example implementation of FIG. 1). Central processing unit (CPU) 501 is coupled to system bus 502. CPU 501 may be any general purpose or specialized purpose CPU. However, the present invention is not restricted by the architecture of CPU 501 as long as CPU 501 supports the inventive operations as described herein. CPU 501 may execute the various logical instructions according to embodiments of the present invention. For example, one or more CPUs, such as CPU 501, may execute machine-level instructions according to the exemplary operational flows described above in conjunction with FIG. 4." Cooley at ¶[0028]: While FIG. 5 shows a purpose computer, it should be noted that the exact configuration of a portion of a system according to various embodiments may be slightly different. For example, handheld devices according to one or more embodiments may be any kind of processor-based device, such as a cell phone, a Personal Digital Assistant, a specialized device, and/or the like. Additionally, servers (e.g., servers 11, 12 of FIG. 1) and the SMS broker according to one or more embodiments may be any kind of processor-based device capable of sending messages (e.g., email messages, SMS messages, MMS messages), such as a personal computer, a server-type computer, and the like. Moreover, embodiments

'117 Claims	Exemplary Disclosures
	of the present invention may be implemented on application specific integrated circuits (ASICs) or very large scale integrated (VLSI) circuits. In fact, persons of ordinary skill in the art may utilize any number of suitable structures capable of executing logical operations according to the embodiments of the present invention." Bobo at 6:44-56: "The MSDS 10 is also connected to a network, preferably the Internet World Wide Web 30. Although the Internet 30 has been shown as a single entity, it should be understood that the Internet 30 is actually a conglomeration of computer networks and is a constantly evolving and changing structure. The invention therefore is not limited to the current structure or form of the Internet 30 but encompasses any future changes or additions to the Internet 30. Further, the MSDS 10 is shown as being directly connected to the Internet 30, such as through its own node or portal. The invention, however, may be practiced with any suitable connection to the Internet 30, such as through an intermediate Internet access provider." Bobo at 7:13-25: "After the MSDS 10 receives a message for one of its users, the user can then communicate with the MSDS 10 at any time and at any location by connecting to the Internet World Wide Web 30 and retrieving the message stored within the MSDS 10. With reference to FIG. 3, at step 60 the user first connects to the Internet 30, such as through a personal computer 32 which may be connected to the Internet 30 in any suitable manner, such as through its own portal or node or through some intermediate access provider. The personal computer 32 is not limited to a single computer but may instead comprise a network of computers, such as a local area network within an office." Bobo at cl. 31:

'117 Claims	Exemplary Disclosures
	"31. A network message storage and delivery system, comprising: a central processor for receiving an incoming call, for detecting an address signal on said incoming call, for detecting a message on said incoming call, and for placing said message in a storage area, said address signal being associated with a user of said network message storage and delivery system; a network server for receiving said message from said storage area, for converting said message into a mixed media page layout language, and for placing said message in said storage area; wherein when said network server receives a request from said user over said network, said network server transmits at least a portion of said message over said network to said user and wherein said network comprises the Internet and said network server comprises an Internet server." Generic Content Download Over The Air, Version 1.0, OMA-Download-OTA-V1_0-20040625-A, Open Mobile Alliance (June 25, 2004) at 17 ("The Device hosting the Download Agent MUST at a minimum support either [W-HTTP] or [WSP], and MAY support either [WAPTLS] or [WTLS], as well as other protocols."). Wakasa at ¶[0039]: "Referring to FIG. 1, the entire configuration of an information providing system 10 according to an embodiment is explained below. As shown in FIG. 1, the information providing system 10, for example, includes portable terminal devices 100 *a*, 100 *b* and 100 *c*, a server device 200, a network 300, readers/ writers 400 *a* and 400 *b* and a PC (personal computer) 500. In the following description, the portable terminal devices 100 *a*, 100 *b* and 100 *c* are each denoted by generic reference numeral 100 if it is not necessary to distinguish the portable terminal devices 100 *a*, 100 *b* and 100 *c* from each other. By the same token, the readers/ writers 400 *a* and 400 *b* are each denoted by generic reference

'117 Claims	Exemplary Disclosures
	numeral 400 if it is not necessary to distinguish the readers/ writers 400 *a* and 400 *b* from each other." Wakasa at ¶[0040]: "The server device 200 generates a transmitted message including parameter information and the identification information of an application to be executed by the portable terminal device 100 on the basis of the parameter information. Then, the server device 200 transmits the transmitted message to the portable terminal device 100 via the network 300. The parameter information is typically information used for requesting the application of the portable terminal device 100 to display information desired by the user or carry out predetermined processing. For example, the parameter information is information requesting the application of the portable terminal device 100 to display "Hello" on the display screen. The portable terminal device 100 executes the application corresponding to the identification information of the application included in transmitted message by using the parameter information as input data. For example, by using the parameter information requesting the application of the portable terminal device 100 to display "Hello" on the display screen as input data, the portable terminal device 100 executes the application corresponding to the identification information, displaying "Hello" on the display screen of the portable terminal device 100. As an alternative, the server device 200 holds an application to be provided to the portable terminal device 100 and, in accordance with a request made by the portable terminal device 100, the server device 200 transmits the application to the portable terminal device 100 via the network 300." Wakasa at ¶[0061]: "The transmitted message generation section 202 has a function to generate a transmitted message including parameter information and the identification information of an application to be executed by the portable terminal device 100 on

'117 Claims	Exemplary Disclosures
	the basis of the parameter information. The transmitted message generation section 202 supplies the generated transmitted message to the transmitted message sender section 204." Wakasa at ¶[0062]: "The transmitted message sender section 204 is a communication interface such as a communication line, a communication circuit or a communication device. The transmitted message sender section 204 sends the transmitted message received from the transmitted message generation section 202 to the portable terminal device 100." Satake at ¶[0070]: "The information providing system according to the first aspect of the present invention comprises: a server that is provided with: a message transmitting section that transmits to a mobile communication terminal a message to which a term of validity is attached and that instructs that a predetermined application in the mobile communication terminal be started up; and an information transmitting section that transmits to the mobile communication terminal predetermined information that is requested by the started up application in the mobile communication terminal; and a mobile communication terminal that is provided with: a receiving section that receives a message instructing that an application transmitted by the server be started up; a storage section that stores an application that requests the server for predetermined information and receives the information; a control section that, when the receiving section receives the message, determines whether or not a term of validity is attached to the message, and, based on the term of validity, starts up the instructed application from among the applications that are stored in the storage section." Satake at ¶[0136]:

'117 Claims	Exemplary Disclosures
	"The system shown in FIG. 1 is formed by a stock price information server 1, an SMS server 6 that transmits an SMS message from the stock price information server 1 to a user, a user terminal 7 serving as a mobile communication terminal belonging to the user, and a mobile exchange system 8 that relays communication between the stock price information server 1, the SMS server 6, and the user terminal 7. The mobile exchange system 8 and the user terminal 7 are connected by two transmission paths, namely, a traffic channel that is used for data communication and a paging channel that is used for SMS (i.e., PUSH messages) communication." *See also* Satake at Fig. 1. Satake at ¶[0145]: "If neither a UNTL header nor a WHLE header are recorded, then the application type list shown in FIG. 10 is checked (S15). In the application type list, an effective period is recorded for each type of application. Types of application include, for example,. stock price services, weather forecast services, news services, alarm services, games and the like. If the effective period is recorded, a value obtained by adding this effective period to the stored time of the messages to the server that is recorded in the header of the SMS messages is calculated as the time of the term of validity (S16). If the effective period is not recorded, then either termless, or an impractical term of validity such as, for example, Dec. 31, 2099 is set (S17)." *See also* Satake at ¶[0147].

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 139 of 442

'117 Claims	Exemplary Disclosures
[1c] the network message server configured to receive, from each of a plurality of network application servers, multiple requests to transmit application data, each such request indicating a corresponding one of the mobile end-user devices and one of a plurality of applications,	The references below disclose and/or render obvious this limitation. For example, see the following passages and/or figures, as well as related disclosures: Huskins at 4:20-33: "In accordance with one embodiment of the present invention, the server 10 may implement a unidirectional messaging system. In a unidirectional messaging system, the server 10 may transmit messages to a plurality of clients that are unable to respond in any way. One example of such a network is a direct-to-home (DTH) broadcast network that may be compliant with the DVB protocol. The network may use a connection oriented communication protocol or a real time connectionless communication protocol as two examples. There are many applications of unidirectional messaging from server to client such as instant messaging, command and control and notification and signaling, as examples. In other cases, the network may be a bidirectional network, for example with an Internet Protocol (IP) multicast backbone." Huskins at 5:1-18: "In one embodiment of the invention, the server 10 may include a unidirectional messaging server (UMS) 22 that is coupled to the servers 16, 18, 20 and 21 to generate messages in an appropriate format. The messages transmitted by the UMS server 22 may include messages originally generated by one of the servers 16, 18, 20 or 21. The UMS server 22 may then be coupled to an Internet Protocol multicast module 24 that places the messages in an appropriate multicast protocol format. Finally, a DVB Multiprotocol Encapsulation (MPE) 26 is coupled to the Internet protocol multicast module 24. The MPE is described in DVB Specification for Data Broadcasts (EN 301 192) and Specification for Service Information (SI) in DVB Systems (EN 300 468 VI .3.1 1998-02) both available from the ETS. The output of the DVB MPE 26 and a DVB-Service Information (SI) generator 28 are coupled to

'117 Claims	Exemplary Disclosures
	the transport 14. Service Information is digital data describing the delivery system, content and scheduling/timing of broadcast data streams." Chou at 2:19-26: "The enterprise application and the wireless secure server may be implemented on separate processing platforms, or as elements of a common processing platform such as an enterprise communication server. "The enterprise application may comprise a dialogue server configurable for use in conducting a multimodal dialogue between the enterprise application and the mobile client device." Shenfield at [0050]: "Push proxy 410 can, based on the metadata provided, create a content package to send to the client. The content package could include the information requested by the client, as well as a digest or summary of related information that the user may be interested in. The summary is sent to the user, but the deferred retrieval message store block 452 stores the actual data that was received from content provider 110. Thus, if in the future the user wishes to obtain more detailed information about information within the digest, this information is already stored at push proxy 410." Shenfield at [0068]: "Message broker and application queues 540 handle messages from push proxy 410 for delivery to applications 512. An application queue is a queue of messages for applications 512." Mostafa at 732: "Different MMS standardization bodies have recently started working to enrich MMS with the feature of transporting application data. Figure 3 depicts different

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 141 of 442

'117 Claims	Exemplary Disclosures
	layers of communication in transporting application data using MMS, showing the vertical interaction within an MMS-capable terminal and an MMS R/S to achieve communication between two applications residing in two different MMS-capable terminals. The figure assumes a simplified scenario, where the terminals belong to the same MMS R/S. Only HTTP is shown there, as it is the mostly used protocol across the terminal-interface. Lower layer protocols, below MMS in the figure, are MMS-specific, and are beyond the scope of this paper. "MMS architecture has been lately extended to include applications [11]. Application is defined as a system element that may interact with an MMS UA or a VASP to transport its data to its peer application, interacting with another MMS UA or a VASP, using MMS. The extension of MMS architecture is depicted in Figure 4, where unrelated entities and interfaces are not shown. Though an application can interact with either an MMS UA or a VASP to transport its data, MMS UA is mostly mentioned in this regard in this paper to cover both for brevity. Though only one application is shown within a mobile terminal and VASP in Figures 3 and 4, there can be multiple applications, as shown in Figure 5. The application interface between an application and an MMS UA is described in Section 4." *See also* Mostafa at Figs. 1, 3; *see generally* Mostafa at 730-732. Tyhurst at [0026]: The mobile data server 120 may be implemented as one or more server modules, and is typically located behind a firewall 113. The mobile data server 120 manages communications to and from a set of managed mobile communication devices 201 such that the mobile communication devices 201 are each enabled to exchange electronic messages and other information with each other and computers, for example via the Internet 112. The mobile data server 120 also

'117 Claims	Exemplary Disclosures
	provides administrative control and management capabilities over users and mobile communication devices 201 which may connect to the mobile data server 120. Novack at [0015]: "According to an aspect of the present invention, a method is provided for securing inter-process communications between a first process and a second process. The method includes receiving, from a device that supports the first process, a request to initiate inter-process communications. The method also includes replying to the device with a request for information of a digital certificate that authenticates the identity of the first process." Novack at [0021]: "According to an aspect of the present invention, a computer-readable medium is provides that stores a program that secures inter-process communications between a first process and a second process. The computer-readable medium includes a receiving code segment that receives, from a device that supports the first process, a request to initiate inter-process communications. The computer-readable medium also includes a digital certificate requesting code segment that replies to the device with a request for information of a digital certificate that authenticates the identity of the first process." *See also* Novack at [0027]. Novack at [0033]: "FIG. 1 shows an exemplary method for securing inter-process communications. The process starts at S100, and a first process begins to run on a first computer at S102. A determination is made that the services of a second process are needed at S105,

'117 Claims	Exemplary Disclosures
	and the first process requests initiation of an inter-process communications session at S110. The request at S110 marks the beginning of inter-process communications. The first computer may send the request to a second computer that runs a second process in the same network. The first computer receives an acknowledgement and a request for authentication at S115. In the example above, the second computer analyzes the initial request and generates an acknowledgement and request for authentication that is received by the first computer at S115." Novack at [0035]: "In one example, the first process is running on an application server that provides application services over the internet to requesting clients. The application server interacts with a management server that manages multiple databases as a back-end service. The first process may be run when a user contacts an application server and downloads presentation layer elements of an application to interact with the application server. The application includes instructions to invoke additional processes for predetermined user requests. Accordingly, the application will request the services of a second process at S110." Novack at [0044]: "FIG. 2 shows an exemplary network architecture for secure inter-process communications. As shown, a **computer** 201 communicates through a **network** 220 with an **application server** 230. As an example, the **application server** 230 may be a web server that provides internet applications for users. The user downloads a web page onto the **computer** 201 when the user contacts an internet address corresponding to the desired internet application. Typically, the web page is accessed and downloaded using hyper text transport protocol (HTTP)." Novack at [0045]:

'117 Claims	Exemplary Disclosures
	"The application server 230 communicates through another network 240 with management server 250. As an example, the network 240 may be a secure local area network that enables the application server 230 to obtain back-end processing. Of course, the network 240 may be the same network as the network 220." Novack at [0046]: "The management server 250 manages databases 254 and 256. The databases 254 and 256 store data associated with the service provided by the application server 230. As an example, the application server 230 may provide internet services for accounts of a service provider. Thus, the application server 230 may provide services for a bank, a utility, or a content provider such as a news service. While the user of the computer 201 is intentionally provided with direct access to the application server 230, the configuration of the network shown in FIG. 2 is intended to help deny the user of the computer 201 direct access to the management server 250 and the databases 254, 256. Further, to ensure that the communications between the application server 230 and the management server 250 are secure, a digital certificate is installed on the application server 230, and the digital certificate is bound to the application server 230 and/or a process running on the application server 230." Novack at [0047]: "In an embodiment, the application server 230 and the management server 250 communicate using the secure sockets layer (SSL) protocol (i.e., HTTPS), for secure and trustworthy communications. A secure sockets layer session begins with an exchange of messages known as a handshake. In the embodiment of FIG. 2, the handshake occurs between the application server 230 and the management server 250. The handshake includes the application server 230 contacting the management server 250. In response to the application server 230 contacting

'117 Claims	Exemplary Disclosures
	the management server 250, the management server 250 sends the application server 230 a digital certificate. The application server 230 responds by providing the management server 250 with a digital certificate. Of course, an exchange of two digital certificates does not need to occur in every case; rather, a single digital certificate may be used when only one party (e.g., the application server 230) to a communications session needs to be authenticated." Novack at [0048]: "The management server 250 may require authentication of the application server 230 for each instance of inter-process communications. In this regard, even if the management server's 250 only function is to process requests for the application server 230, the management server 250 may need to authenticate the application server for each request for inter-process communications when, for example, the application server 230 widely interacts with the public and is therefore subject to unknown risks. The management server 250 may request verification of the application server's 230 address, the particular requesting function, and any other information that will help authenticate the request. Of course, the management server 250 may demand a digital certificate that is bound to the application server 230 or even a particular process that runs on the application server 230." *See also* Novack, FIG. 2. Novack at [0064]: "The device in FIG. 4 may include multiple processes that communicate with each other using the secure inter-process communications of the present invention. Additionally, although the terms "client" and "server" are used to refer to devices, the devices may be defined and distinguished by a set of software instructions installed as a pre-packaged program. Further, while examples of the uses of the present invention include secure client/server networks and peer-to-peer networks,

'117 Claims	Exemplary Disclosures
	secure inter-process communications can be used to enhance security in any inter-process communications environment. Further, while the embodiment shown in FIG. 2 only shows a single computer 201 and a single application server 230, secure inter-process communications can be used to enhance security for multiple application servers 230 belonging to a single entity or multiple users using computers 201. Additionally, while networks 220 and 240 are separately designated in FIG. 2, the networks may be the same, such as an internet protocol (IP) network. Further, while a certification authority is discussed in the context of the present invention, a certification authority is not shown in the figures. In this regard, a certification authority may communicate over the network 220 and/or the network 240 to provide digital certificates to the application server 230 and the management server 250. Of course, the digital certificates may be provided by the certification authority as portable memory modules that are installed on the application server 230 or the management server 250. Furthermore, the entity that provides the application server 230 and/or the management server 250 may also act as a certification authority for its customers using the computers 102." Santamaria at [0049]: "According to one embodiment of the invention, each notification service client on a mobile device performs the operations of FIG. 4, independently. At block 410, a notification service client (e.g., a push notification service client or a pull notification service client) registers with a notification service connection manager. For example, with reference to FIGS. 3A and 5, the notification push notification services 110, 120, and the notification pull notification service 130 (each a notification service client 310) each independently register with the notification service connection manager 360. From block 410, flow moves to block 412." Isomursu at 2:56-3:33:

Headwater Research LLC
Ex. 2011, IPR2024-00003
Page 147 of 442