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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 December 2019 (12.12.2019)

WIPO | PCT

(10) International Publication Number
WO 2019/234409 A1

(51) International Patent Classification:
G06F 21/72 (2013.01) *G06F 21/62* (2013.01)
G06F 21/34 (2013.01) *G06F 21/60* (2013.01)

(21) International Application Number:
PCT/GB2019/051542

(22) International Filing Date:
04 June 2019 (04.06.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
1809258.5 06 June 2018 (06.06.2018) GB

(71) Applicant: **ISTORAGE LIMITED** [GB/GB]; iStorage House, 13 Alperton Lane, Perivale Middlesex UB6 8DH (GB).

(72) Inventors: **MICHAEL, John**; 21 Glencairn Drive, Ealing, London Greater London W5 1RT (GB). **XU, Quan**; c/o iStorage Limited, iStorage House, 13 Alperton Lane, Perivale Middlesex UB6 8DH (GB).

(74) Agent: **DEHNS**; TAYLOR, Adam, St Bride's House, 10 Salisbury Square, London Greater London EC4Y 8JD (GB).

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

(54) Title: DONGLE FOR CIPHERING DATA

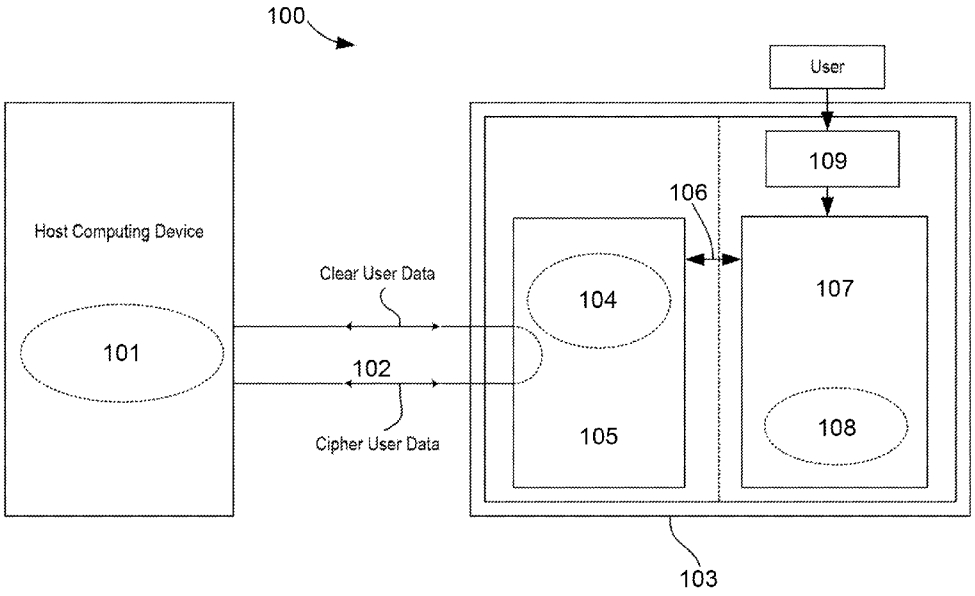


Fig. 1

(57) Abstract: A dongle (103) for ciphering data takes the form of a portable hardware device (103) for receiving and transmitting data to and from an external device. The dongle (103) comprises a user interface (109) for interaction with a user to receive authentication data for confirming the identity of a user, wherein the use of the dongle (103) for ciphering data is not enabled unless an authorised user has been identified. A data transfer channel (102) is included for coupling the dongle to the external device for transfer of data between the dongle and the external device. In order to perform a ciphering transformation on data received by the dongle (103) from the external device the dongle includes a hardware encryption engine (104). The dongle (103) is arranged to use the data transfer channel (102) to receive user data from the external device, perform the ciphering transformation on the user data via the hardware encryption engine (104) and, in a continuous operation, return the transformed user data to the external device via the data transfer channel (102), such

WO 2019/234409 A1

[Continued on next page]

SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Published:

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

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

that the dongle (103) performs real-time hardware ciphering of user data using a single data transfer channel (102) without storage of the user data on the dongle (103).

DONGLE FOR CIPHERING DATA

The invention relates to a dongle for ciphering data, the dongle taking the form of a portable hardware device for receiving and transmitting data to and from an external device, and the dongle requiring authentication of the identity of an authorised user before the ciphering operation is permitted. The invention further extends to related methods and computer systems.

It is often desirable to store data securely or to transfer data securely between various users. When the data is sensitive it can be encrypted before transferring it over potentially insecure networks or before storing it either on a hardware device or on the "cloud". It is then decrypted during future access to the data. Encryption and/or decryption can collectively be referred to as ciphering.

Ciphering to make data more secure can be done via software or via hardware. The use of hardware encryption engines has advantages in relation to the security of the ciphering transformation and it can allow for fast data transfer with effectively real-time encryption. Hardware encryption is more secure than software based encryption in terms of protection of encryption keys, for example since the keys can be stored internally in the hardware (e.g. in a dongle) and accessed only by internal processors. Hardware based systems can also be made yet more secure via authentication systems incorporated in the device, like a keypad or biometric sensor, making the authentication process safer than the keyboard used by software encryption, which is more vulnerable to attacks such as keyloggers. The applicant company, iStorage Limited, of the United Kingdom, markets a range of products making use of hardware encryption such as the diskAshur™ range of portable encrypted hard drives. These products use a USB 3.0 interface and real-time XTS-AES hardware encryption for data stored on the hard drive. Advantageously, the encryption key is securely stored on the device itself, with protection being provided by a user authentication system such as PIN or biometric. Other products are also available making use of various forms of encryption and differing encryption/ciphering techniques.

One known ciphering technique is "public-key cryptography". In public-key cryptography, each user has a pair of cryptographic keys called the public key and the private key. The public key may be publicly shared with anyone wishing to securely communicate with the key's owner, while the private key is for decrypting data that has been encrypted using that public key. When A wishes to communicate securely with B, A encrypts the message (data) using B's public key and sends the

encrypted message to B. B then uses B's private key to decrypt the message. Advantageously, this method does not require the two users, A and B, to secretly exchange an encryption key with one another before communicating over insecure channels.

Another form of encryption is "symmetric-key encryption" in which the same encryption key is used for both encryption and decryption. In such a system, users A and B must first securely exchange the encryption key before they communicate with one another over insecure channels.

Symmetric-key encryption systems have advantages and disadvantages compared to public-key cryptography. Generally, symmetric-key systems are faster for encoding and decoding. In symmetric-key systems encrypted data can be transferred even if there is a possibility that the data will be intercepted. Since there is no key transmitted with the data, the chances of data being decrypted are lower than in public-key systems. Finally, only a recipient who has the key can decrypt the data. Thus senders can be confident that only an authorised recipient can decrypt the data. One compromise arising from the use of symmetric-key encryption is the requirement to provide the initial encryption key to all the users in a secure manner.

Proposals have been made for standalone ciphering devices such as the hardware encryption dongle described in WO2009/136161. This dongle permitted real-time data transfer with simultaneous encryption via a pair of USB interfaces, with one USB connection receiving clear data from a host computing device, and the other USB connection transferring cipher data to an external hard drive or other storage. The diskAshur™ products can be considered as an integration of this type of hardware encryption dongle with a built in hard drive, such that the hardware encryption engine communicates via two data transfer channels, one coupled to a host computer device via a USB 3.0 interface, and the other coupled internally to the hard drive.

Despite the advances provided by the applicant's earlier devices there remains a need for further development of hardware encryption devices such as dongles for ciphering data.

Viewed from a first aspect, the invention provides a dongle for ciphering data, the dongle taking the form of a portable hardware device for receiving and transmitting data to and from an external device, and the dongle comprising:

a user interface for interaction with a user to receive authentication data for confirming the identity of a user, wherein the use of the dongle for ciphering data is not enabled unless an authorised user has been identified;

a data transfer interface for coupling the dongle to the external device for transfer of user data between the dongle and the external device; and

a hardware encryption engine for performing a ciphering transformation on user data received by the dongle from the external device;

wherein the dongle is arranged to use the data transfer channel to receive user data from the external device, perform the ciphering transformation on the user data via the hardware encryption engine and, in a continuous operation, return the transformed user data to the external device via the data transfer channel, such that the dongle performs real-time hardware ciphering of user data using a single data transfer channel without storage of the user data on the dongle.

With this dongle the user, after confirmation of their identity as an authorised user, can perform high-speed ciphering operations via the hardware encryption engine with the data being transferred to and from any data storage located on the external device, or accessible to the external device, including data storage in the "cloud". This may use software on the external device or in the cloud, with the software preferably incorporating a dedicated Application Programming Interface (API) that is necessary to access the dongle. This adds further security. As is known, hardware encryption engines can operate with minimal delays, typically in the micro-second range. It has been realised that this permits a single data transfer channel to be used for both incoming and outgoing data, thus allowing one of the two data transfer channels of the prior art (e.g. as in WO2009/136161) to be omitted. The use of only a single data transfer channel makes use of the device more convenient as well as reducing the size and complexity of the dongle.

In addition, as has been demonstrated by existing devices such as the diskAshur™ products sold by iStorage Limited, suitably arranged user authentication systems, which can include authenticated access for encryption keys stored on the device, allow for a product that is both high security and user-friendly. The combination of features of the first aspect therefore result in the combined capability for both high-speed ciphering and high security ciphering. The dongle is also referred to herein as a hardware encryption token. The external device is hardware external to and separate from the dongle, and may for example be a host computing device.

The ciphering transformation may include encryption and/or decryption, for example by using an encryption key stored securely on the dongle. In this context, an encryption key (or “keying material”) may be a random string of bits created explicitly for scrambling and unscrambling data (i.e. encrypting and decrypting data). Encryption keys are usually designed using algorithms that are intended to ensure

that every key is unpredictable and unique. The encryption key may have a length of 256 bytes (2048 bits) or more.

The transformed user data may hence be an encrypted form of original non-encrypted user data, or a decrypted form of original encrypted user data. The references to ciphering and a ciphering transformation are intended to encompass encryption or decryption, or both encryption and decryption. The ciphering transformation may include a conversion between cipher data and clear data, such as between cipher text and clear text, and vice versa.

The dongle may hold an encryption key that is accessible only to an authorised user as identified by the user interface, in which case the encryption key may be stored in a ciphered form with access and decryption of the encryption key only being permitted once the user interface identifies an authorised user. The encryption key may for example be for use in symmetrical key encryption. Thus, the encryption key held on the dongle may be a secret key as used for symmetrical key encryption and the dongle may perform encryption and/or decryption using a single encryption key stored on the dongle.

As noted above, the dongle of the first aspect uses only a single channel for continuously receiving data from the external device and sending data to the external device. The ciphering transformation by the hardware encryption engine may be practically instantaneous and hence in example embodiments the return transmission of a transformed data packet to the external device may occur without delay after the receipt of the data packet from the external device. This, there is no time delay other than the time required for the hardware encryption engine to perform the ciphering transformation. This can be of the order of μs as noted above. Thus, the ciphering transformation by the hardware encryption engine advantageously occurs directly upon receipt of a data packet from the external device, with the return transmission of a transformed data packet to the external device similarly occurring directly upon completion of the ciphering transformation.

The data transfer channel may operate in sequential steps for transmission of user data from the external device to the dongle, and for return of transformed user data from the dongle to the external device. Alternatively, with a suitably high-speed interface there may be simultaneous transport of user data in both directions. Thus, as a data packet *n* is transmitted to the dongle a transformed data packet *n-1* is returned from the dongle, then as a data packet *n+1* is transmitted to the dongle, a transformed data packet *n* is returned from the dongle. The hardware encryption engine may be arranged to continuously transform the data whilst the data transfer

channel is continuously transporting both incoming data and outgoing transformed data to and from the hardware encryption engine.

It will be understood that it is significant advantage of the dongle of the first aspect that only a single data transfer channel is required for the ciphering operation. The dongle may optionally include no further data transfer channels, such that there is only a single data transfer channel, which may be used for during ciphering of data as well as being used for other operations such as for exchange of commands between the dongle and the external device and/or for operations conducted during commissioning of the device, such as enrolment of the user(s) and or transfer of an encryption key to the device.

The data transfer channel is advantageously capable of high speed data transfer rates, and in particular may be capable of data transfer rates equivalent to that available via USB 3.0 "SuperSpeed USB". Thus, the data transfer channel may be capable of data transfer at rates of up to 5 Gbit/s (625 MB/s), or at even higher rates. The data transfer channel may include a data transfer interface such as a USB 3.0 interface, or a successor USB standard such as USB 3.1, etc. The use of a USB interface allows the dongle to be compatible with many computer devices, with the data transfer rates of USB 3.0 and above allowing for advantages in terms of high speed mass data transfer during the ciphering process, resulting in high speed real-time ciphering. Alternatively or additionally, the data transfer channel may comprise one or more of wireless USB, Serial ATA (SATA), High Definition Multimedia Interface (HDMI), Peripheral Component Interconnect Express (PCIe), and radio frequency wireless networks, with the use of suitable interfaces as appropriate.

The dongle of the first aspect does not store the user data during the ciphering process and instead it continuously ciphers data and returns transformed data to the external device. Advantageously there may be no mass data storage device on the dongle, i.e. it is not capable of storage of encrypted or non-encrypted user data transferred to the dongle from the external device. The dongle may hence be a dedicated ciphering device, intended to handle ciphering of user data without having any mass data storage function. It is to be noted that the absence of storage for user data does not mean that the device has no capability for storage of information at all, and naturally there may be some storage enabling the dongle to retain data required for its operation. For example with the use of an encryption key held on the dongle then this may be stored (or encrypted and stored) in a dedicated memory, which may be used only for encryption key storage or may be used for encryption key storage as well as storage of other information necessary for operation of the dongle, such as

authentication data required to confirm the identity of authorised users. However, it is preferred to have no capability to store user data, which instead is transformed by the ciphering operation and returned via the data transfer channel without being stored.

The user interface is arranged for interaction with a user to receive authentication data for confirming the identity of the user. This may include the use of a code such as a PIN and/or the use of biometric data such as a fingerprint. Thus, the user interface may include a keypad for receiving a code and/or a biometric sensor such as a fingerprint sensor. Possible elements of the user interface include one or more of a keypad, biometric sensor, thumb wheel, rocker switch, touch pad, selector switch, encoder, potentiometer, photoelectric switch, or membrane switch, or other suitable device enabling the identity of the user to be confirmed via some form of code or biometric input. As noted above, the user interface may interact with the dongle in order to permit access to an encryption key on the dongle. The dongle may be arranged to require authentication of the identity of the user prior to each use of the dongle, and/or at set time intervals during use of the dongle.

Advantageously, multi-factor authentication may be required in order to enable the use of the dongle for ciphering data, with the multifactor authentication including confirmation of the user's identity via the user interface and confirmation of the dongle's identity via dongle authentication data stored on the dongle. For example, the dongle authentication data may include a suitable digital certificate, such as a certificate with a signature that can be used to check the identity of the dongle. In some embodiments, to allow the hardware encryption engine to perform a ciphering transformation on user data, then the dongle needs to firstly confirm the identity of user through the user interface, and then exchange certificates with server and authenticate the identity of the dongle via the certificate signatures.

The dongle may be arranged to provide real-time ciphering of any type of data, including voice call encryption, e-mail encryption and/or encryption in relation to software as a service (SaaS), i.e. software based on a SaaS architecture.

In a second aspect the invention includes a method of ciphering data using the dongle of the first aspect. The dongle may include any of the other features discussed above. The method can include receiving authentication data for confirming the identity of a user via the user interface, checking the authentication data or authentication data derived data against stored data for authorised users (e.g. a verify a PIN or to perform a biometric identification process), and permitting access to the ciphering function of the dongle only when the received authentication data identifies the user as an authorised user. The method may include transmitting user

data from the external device to the dongle via the data transfer interface, performing the ciphering transformation at the dongle using the hardware encryption engine, and returning the transformed user data from the dongle to the external device via the data transfer interface.

The method may include use of the device for real-time ciphering of any type of data, such as use for voice call encryption, e-mail encryption or encryption in relation to software as a service.

The use of the dongle may include the use of dedicated software services on the external device to facilitate transfer of data between the dongle and the external device and/or to facilitate transfer of data to the cloud. In some examples, the dedicated software services include an application programming interface that can be integrated into the operating system or host application running in the external device, which may be a host computing device. By use of the software services, the external device may be arranged to check peripherals to locate dongles intended for use with the software services. Once a dongle is recognised then the software services may initiate the transfer of data to and from the dongle, and prompt the ciphering transformation via the hardware encryption engine of the dongle (subject to the dongle having recognised an authorised user).

In a third aspect, the invention provides a computer programme product comprising instructions that, when executed, will configure a dongle as in the first aspect to operate in accordance with a method as in the second aspect.

In an extension to the above concepts the dongle may be one of a set of multiple dongles, with each dongle being provided with the same encryption key. Thus, there may be multiple separate dongles capable of ciphering data using the same ciphering transformation. This can allow for multiple users to decrypt data using the same encryption key or for multiple users to contribute different sets of encrypted data that can then be decrypted using one encryption key for each of the different sets of data. One possible application for this is secure access to encrypted data at multiple different geographic locations, enabling secure transmission of data between users without the need to separately share passwords or transmit encryption keys between users. It can also allow for secure global transfer of data via the cloud, with the data being in encrypted form whilst it is stored on the cloud and being decrypted in real-time via a dongle as described herein for access to the data from any location without any concern about third parties accessing the data whilst it is on the cloud.

The multiple separate dongles may include a master dongle and a slave dongle, wherein the slave dongle has been securely provided with a copy of a master encryption key (master keying material) from the master dongle. The slave dongle may have been provided with the master keying material by a method comprising: connecting the master dongle and the slave dongle to a data transfer system; transferring the slave dongle's public key to the master dongle via the data transfer system; using the slave dongle's public key at the master dongle to encrypt the master keying material and hence produce encrypted master keying material at the master dongle; transferring the encrypted master keying material to the slave dongle via the data transfer system; decrypting the encrypted master keying material with the slave dongle's private key at the slave dongle; and storing the master keying material at the slave dongle; such that a user of any of the dongles can use the master keying material to decrypt data encrypted by the same dongle or the other of the dongles. Thus, the present invention may extend to a method of manufacturing a dongle as in the first aspect, this method including providing the dongle with master keying material as set out above.

The master keying material may be a master encryption key, such as for use in the ciphering transformation by the hardware encryption engine.

With this method, the slave dongle's private key and the unencrypted master keying material are advantageously not transferred across or read by the data transfer system. The method thus allows the secure production of one or more slave dongle(s) that contain the master keying material and hence allows for secure production of a set of dongles that share the same master keying material. The dongles may then provide a secure dongle-based symmetric key system.

The master keying material can be transferred to a plurality of slave dongles using the same method.

The data transfer system may be a personal computer (PC) that, optionally, is connected to multiple slave dongles for rapid production of a set of slave dongles having the same master keying material. Alternatively, the data transfer system may be a dedicated device designed for the secure production of many slave dongles having the master keying material. Such a dedicated device is advantageously not connected to the internet and may be configured to only run the necessary processes for transferring (encrypted) data between master and slave dongles. For example, the data processing routine for performing the method may be hard-coded into a computer chip in the dedicated device. This can provide extra security when transferring the master keying material against attacks via the internet or from

malicious dongles plugged into the dedicated device. In another example, the data transfer system may be a cable or a hub with no internal programs or processor. In this case, the master and slave dongles may be configured to carry out the method using a data processing device in one or each of the dongles. Thus, the dongle(s) may include a processor or similar for performing computing operations.

The dongles include a user interface for identifying an authorised user as discussed above. The master dongle may require a higher level of authorisation before the master keying material can be transferred to another dongle. Thus, the slave dongle(s) and the master dongle when used for encrypting and decrypting data may be accessible to a normal authorised user, whereas there may be a higher level user, such as an administrator, who is permitted access to the master dongle to copy the master keying material and/or access to the slave dongle(s) to access the slave dongle public key and to install new master keying material.

The method may comprise scrambling the slave dongle's public key using a predetermined scrambling algorithm prior to transferring the slave dongle's public key, and unscrambling the slave dongle's public key at the master dongle after transferring slave dongle's public key and prior to encrypting the master keying material. In this case the slave dongle(s) may include a suitable scrambling algorithm and the master dongle may include instructions for unscrambling the scrambled slave dongle's public key. This helps to protect against the case where the data transfer system is compromised and the malicious user has emulated a slave dongle but generated his own public key that is sent to the master dongle for the purposes of discovering the master keying material.

With the added use of scrambling of the slave dongle public key, only the legal slave dongles know the scrambling algorithm and this may be hard-coded onto the dongles, e.g. at the factory when first produced. The data transfer system (and any software thereon for the purposes of performing the method) has no knowledge of the scrambling algorithm, and optionally only the master dongle includes the instructions needed to unscramble the slave dongle public key. This provides a further layer of security against malicious emulated slave dongles, because the hacker has no knowledge of the scrambling algorithm as this is not contained in, nor ever transferred over, the data transfer system, and may only ever be present on the master dongle, where it may be protected via a suitable user authentication system as discussed above. In a case where the length of the public key is 256 bytes (2048 bits), to brute-force crack the scrambling algorithm, an attacker must try 256! times to hack the scrambling algorithm.

As noted above, a plurality of slave dongles may be provided, each having a respective public key and a respective private key; and the method may comprise, for each slave dongle in turn: transferring the slave dongle's public key to the master dongle; encrypting the master keying material with the slave dongle's public key at the master dongle; transferring the encrypted master keying material to the slave dongle; and decrypting the encrypted master keying material with the slave dongle's private key at the slave dongle.

This provides the advantage of rapidly producing a plurality of slave dongles that can be distributed to a plurality of users for the purposes of securely sharing information encrypted with the master keying material.

According to another aspect, there is provided a method of sharing data securely between authorised computing devices, wherein a first computing device is connected to a first dongle as described above and a second computing device is connected to a second dongle as described above, the first and second dongles having been programmed with the same master keying material by any of the methods described above, the method of sharing data comprising: encrypting first data using the first dongle with the master keying material stored in the first dongle; transferring the encrypted first data, optionally across a network, to the second computing device; decrypting the encrypted first data using the second dongle; and transferring the unencrypted first data from the second dongle to the second computing device.

According to another aspect, there is provided a system for securely transferring master keying material between a master dongle and a slave dongle, wherein the slave dongle is a dongle as in the first aspect and optional features thereof, the system comprising: a data transfer system; the master dongle containing a master processor, and master keying material; and the slave dongle containing a slave processor, a slave public key and a slave private key; wherein the data transfer system has a plurality of ports for connecting to the master dongle and to the slave dongle; the data transfer system being configured to transfer the slave dongle's public key to the master dongle; the master dongle being configured to encrypt the master keying material with the slave dongle's public key to produce an encrypted master keying material and to return the encrypted master keying material to the data transfer system; the data transfer system being configured to transfer the encrypted master keying material to the slave dongle; the slave dongle being configured to decrypt the encrypted master keying material using the slave dongle's private key and to store the master keying material at the slave dongle such that a user of any of the

dongles can use the master keying material to decrypt data encrypted by the same dongle or the other of the dongles.

The slave dongle may be configured to scramble the slave dongle's public key using a predetermined scrambling algorithm prior to transferring the slave dongle's public key, such that the data transfer system transfers a scrambled slave dongle's public key; and the master dongle may contain the scrambling algorithm and be configured to unscramble the scrambled slave dongle's public key prior to encrypting the master keying material.

The system may comprise a plurality of slave dongles, each being a dongle as in the first aspect or optional features thereof, and having a respective public key and a private key. The data transfer system may contain at least three ports, wherein one port is for connecting to the master dongle and the at least two other ports are for connecting to at least two of the plurality of slave dongles.

A system according to this arrangement can quickly prepare a plurality of slave dongles, without the user having to remove each slave dongle after it receives the master keying material and attach a new slave dongle to the data transfer system.

According to another aspect there is provided a system for securely communicating data between users, the system comprising: a first computing device coupled to a first dongle, a second computing device coupled to a second dongle, wherein the dongles are dongles in accordance with the first aspect and each hold a common encryption key, optionally wherein the dongles have been configured using any of the aforementioned methods or systems, wherein, when the first computing device is used to send first data to the second computing device, the first computing device is configured to send the first data to the first dongle, the first dongle is configured to encrypt the first data using the encryption key, and the first device is configured to send the encrypted first data to the second computing device, optionally via a network, and wherein the second computing device is configured to receive the first encrypted data and send the first encrypted data to the second dongle, the second dongle is configured to decrypt the first encrypted data using the encryption key and to send the decrypted first data to the second computer device.

This system allows users to communicate securely with one another using symmetric-key encryption which is faster than public-key encryption. Further, as the disclosure provides a secure method for providing the master keying material to users, a known disadvantage of symmetric-key encryption regarding the initial transfer of the master keying material is overcome.

As described above, a plurality of dongles may be configured by software or hardware to hold the same master keying material (e.g. encryption key). This can allow for multiple users to decrypt data using the same master keying material or for multiple users to contribute different sets of encrypted data that can then be decrypted by other users using the master keying material common to the dongles.

In an arrangement where a plurality of dongles are configured by software to hold the same master keying material, the dongles may be used to encrypt and decrypt e-mail or messaging apps where the two communicating parties are the only ones that have the secret master keying material. Messages as well as email attachments may be encrypted by the sending party (who has one dongle), and only the receiving party with the same encryption key (who has a second dongle having the same master keying material) can decrypt those messages and/or attachments.

Such dongles may alternatively or additionally be used for real-time communications such as voice over IP calls, again where the two communicating parties are the only ones that have dongles containing the same secret master keying material.

Such dongles may alternatively or additionally be used for sharing files of any sort, whether the files are stored in media or the cloud, or on a server, or locally shared folders, or transmitted through network.

Such dongles may alternatively or additionally be used for secure distribution of software that is protected from execution by unauthorised users. Dongles are provided to users who are authorised to use a certain software. The software will check for the correct key or a unique identifier (in a dongle) to enable the software to be executed.

Such dongles may alternatively or additionally be used to prevent piracy of entertainment media protected by DRM (digital rights management). Often entertainment media (such as videos or music) are encrypted and cannot be played without the correct decryption keys. Dongles according to the present disclosure may be provided to users who are authorised to play the media; the master keying material in the dongles may then be used by DRM software to decrypt the entertainment media.

Such dongles may alternatively or additionally be used as a second factor authentication for online services. For instance, a user may present a login name and password to sign into an online service. The dongle is pre-registered with the online service as a second factor. The online service asks for the dongle to be connected to the computer after the login details are submitted.

Such dongles may alternatively or additionally be used for document tracking to record users who access and view a certain document. For example, it may be important for legal or Top Secret intelligence documents to be restricted to specific individuals and record the identities of those individuals who have viewed the document. Each authorised individual may be provided with a unique key in a dongle and a protected document can only be viewed if an individual has a dongle containing the authorised key. That is, in this example, the dongle functions as a factor of authorisation. The identity of the dongle may be recorded with the document, e.g. by marking the document with the unique key.

Such dongles may alternatively or additionally be used to authorise online transactions, for example online payments. A payment service may be configured to require that a pre-registered dongle is presented before a payment is authorised.

In addition to encryption keys for symmetric key cryptography, dongles according to the present disclosure may further contain a public/private key pair for asymmetric key cryptography. This public/private key pair may be distinct from, or the same as, the public/private key pair that is used in the transfer of the master keying material to the dongle(s), as described above.

This may allow the dongle to be used in, for example, the FIDO (Fast IDentity Online) Alliance's Universal 2nd Factor (U2F) specification, which is a specification for a dongle to serve as a second factor authentication to online services. According to the specification, a FIDO U2F dongle contains a public/private key pair. The user registers the dongle's public key with an online service. Later, the user logs into the online service with user credentials, e.g. login name and password. The service then asks for the user's dongle to be plugged into the computer. The dongle creates a signature using the private key and the online service can verify the dongle by the pre-registered public key.

The inclusion of a public/private key pair in a dongle according to the present disclosure may be useful in allowing the dongle to support the following functions (in addition to data encryption/decryption): creating a digital signature for messages/documents to verify the creator and its data integrity; and/or creating a hash-based message authentication code (HMAC) for messages/documents to verify the creator and its data integrity.

Such a dongle can further include a certificate in addition to the symmetric key(s) and the asymmetric (i.e. public/private) key pair. The certificate may be issued by a trusted certificate authority. Applications or services that intend to use dongles for authentication and encryption may also include a certificate issued by the same

certificate authority and can verify the validity of the dongles through mutual authentication based on the certificates. The applications or services can therefore confirm the dongle is genuine prior to using it for authentication or encryption, which further enhances the security of the application scenarios described above.

A unique key may be stored on the dongle that allows the dongle to function as a factor of authorisation.

The dongle may contain a public/private key pair for creating a digital signature for messages/documents to verify the creator and its data integrity and/or for creating a hash-based message authentication code for messages/documents to verify the creator and its data integrity. Optionally, the dongle may further contain a certificate for verifying the validity of the dongle.

The dongle containing a public/private key pair may also create a digital signature for an online service to verify with a pre-registered public key to achieve multi-factor authentication.

Certain preferred embodiments of the invention will now be described by way of example only and with reference to the accompanying drawings, in which:

Figure 1 shows a schematic diagram for a data encryption system;

Figure 2 illustrates the use of the data encryption system during operation;

Figure 3 shows a computer attached to a hub into which are plugged a master dongle and several slave dongles;

Figure 4 shows a sequence diagram of method steps for securely transferring master keying material from a master dongle to a slave dongle;

Figure 5 shows a dedicated device for accepting a master dongle and a plurality of slave dongles; and

Figure 6 shows a system of multiple devices, each device having a slave dongle, wherein users can communicate securely with one another over a network using common master keying material.

Referring to Figure 1, therein is shown a schematic of a data encryption system (DES) 100 using a dongle 103 as a hardware encryption token 103. The DES 100 includes a high speed data transfer channel 102 including a data transfer interface (e.g. USB connector) for transfer of data between the dongle 103 and an external device, such as a host computing device. The dongle 103 is a separate hardware device to the host computing device. It is a portable hardware device that may be disconnected and used with other external devices in differing locations. The DES 100 further includes a DES application programming interface 101.

The DES application programming interface 101 is a group of software services that can be integrated into the operating system or host application running in the host computing device. By initiating the services, the host computing device can check peripherals to search valid DES hardware encryption tokens (dongles) 103, manipulate the host data, and perform the user data encryption/decryption via the token 103.

The data transfer channel 102 provides a means of exchanging data with a host computing device. Universal Serial Bus (USB) is one of the most popular means to connect the DES hardware encryption token 103 to the host. Other examples of the data transfer channel 102 include wireless USB, Serial ATA (SATA), High Definition Multimedia Interface (HDMI), Peripheral Component Interconnect Express (PCIe) and radio frequency wireless networks. Only one data transfer channel 102 is available in the example hardware encryption token 103, and hence the high speed transfer of user data in and out to the token is done through the same data transfer channel 102.

The DES hardware encryption token 103 includes a hardware encryption engine 104, a high speed data bus controller 105, an authentication controller 107, an encryption key 108, and a user interface 109 in the form of a user ID input mechanism 109. To enable the token to be recognised by a host device with the DES application programming interface 101, the user must enter a valid user ID through the user ID input mechanism 109. The user ID input mechanism 109 can include one or more of a keypad, biometric sensor, thumb wheel, rocker switch, touch pad, selector switch, encoder, potentiometer, photoelectric switch, or membrane switch, or other suitable device enabling the identity of the user to be confirmed via some form of code or biometric input.

The authentication controller 107 is operable to control the communication and release of the encryption key 108 to the encryption engine 104 via the internal communication interface 106, based on an input ID entered through the user ID input mechanism 109. Thus, in this example the confirmation of the user as an authorised user both releases the encryption key 108 and also makes the dongle 103 accessible via the DES application programming interface 101. The internal communication interface 106 can be Serial Peripheral Interface bus (SPI), Inter-Integrated Circuit (I2C), General-purpose input/output (GPIO), Universal Serial Bus (USB), ISO/IEC interface. The authentication controller 107 can be a microcontroller or a microprocessor. To provide high security, the authentication controller 107 ideally includes physical security mechanisms that protect it from external violations, side channel attacks, or probing attacks.

The data bus controller 105 is operable to control access by a host device to the DES hardware encryption token 103 so that the token 103 can only be recognised when the authentication controller 107 initiates the communication and releases a valid encryption key.

The hardware encryption engine 104 can be implemented as part of the data bus controller 105. The encryption engine 104 receives data sent by the host computing device via data transfer channel 102 and performs a ciphering transformation. Thus, the encryption engine 104 can receive clear user data and convert it to cipher (encrypted) user data that is sent back to the host device via the same data transfer channel 102. Likewise, the encryption engine 104 can also receive cipher (encrypted) user data sent by host device through the data transfer channel 102 and convert it to clear user data that is sent back to host device via the same channel. The decryption/encryption is an on-the-fly operation done by hardware thus the latency between sending clear user data and receiving cipher user data is negligible, generally in the magnitude of μs. The encryption engine can also be separate controller has the encryption capability to encrypt/decrypt data on the fly along with managing the communication protocol, memory, and other operating conditions.

An encryption key 108 is required by the hardware encryption engine 104 to encrypt/decrypt the information. The encryption key 108 is used in an algorithm (e.g., a 256 bit Advanced Encryption Standard (AES) encryption) that respectively encrypts/decrypts the data from the host device. The encryption key is accessed by the authentication controller 107 when the DES hardware token 103 is online and a valid input ID (e.g. PIN, biometric) is provided by the user. The encryption key can be stored internally or externally but for reasons of security it should not be stored in clear text. The internal or external non-volatile memory that stores the encrypted encryption key shall be protected, controlled access or scrambled.

It has been discovered that, by the employment of hardware based encryption engine 104, a high speed data transfer channel 102, a secure authentication controller 107, and the encryption key 108, the proposed data encryption system can be provide an extremely high speed and massive user data encryption/decryption with both clear and cipher data transferring continuously (and optionally simultaneously) on the same data transfer channel 102.

Figure 2 expands on the content of Figure 1 and shows how a host application 114 may operate the DES application programming interface 101, and how a server and the hardware token (dongle) 103 can verify the certificates of one another.

The host computing device is provided with a host application 114. The host application 114 is software that incorporates the DES application programming interface 101 so by running the application 114, the user can locate a valid hardware token 103 (e.g. upon connection of the dongle 103 via the data transfer channel 102) and then perform data encryption/decryption with the hardware token 103. The host application takes files in any format supported by the application, converts the files to binary data, and calls DES application programming interface services to communicate with the hardware token 103. When receiving cipher/plain user data, the host application converts the data to files in the corresponding formats for specific use cases.

The high speed data transfer channel 102 is used in this illustration as a channel to exchange commands between host computing device and the DES hardware encryption token 103, in addition to performing the data transfer during the ciphering process. A network channel 113 is employed to exchange commands/data between a server and the host computing device.

The server includes a server certificate 111 and a server application 112. The server application is operable to handle certificate verification and communicate with a host application 114. In one example a certificate is issued by the root of trust, or issuer, a trusted third-party capable of generating certificates. All such certificates can include the issuer public key and a signature generated by issuer with the issuer private key. A DES certificate 110 can be held on the dongle 103, with the server certificate 111 and DES certificate 110 having different signatures. The server certificate 111 is stored in server non-volatile memory. The DES certificate 110 can be stored in the internal or external non-volatile memory of the authentication controller 107 on the hardware token 103. A process of certificate verification provides an additional layer of security to the data encryption system in that only a verified DES hardware token 103 can be recognised by the host computing device and used for user data encryption/decryption.

An example process of verifying certificates is as follows.

1. The user enters correct user ID through the user ID input mechanism 109, which is authenticated by the authentication controller 107.

2. The DES hardware encryption token 103 is connected to a host computing device

3. Running the host application 114, the hardware token 103 and server begin to exchange the certificates

4. The server and hardware token 103 verify the signatures in the certificates respectively using the issuer public key.

5. If the signatures are verified then the hardware token 103 is recognized by the host application 114.

6. The host application 114 performs massive user data encryption/decryption using the DES hardware encryption token 103.

The dongle (hardware encryption token) 103 may advantageously be one of several dongles 103 that have the same encryption key, so that multiple users in different locations can encrypt and decrypt commonly accessible data. To securely implement such a system it is necessary to securely transfer the common keying material to multiple dongles. One way to do this is to create a master dongle with master keying material, and to then transfer this master keying material to one or more slave dongles. Figures 3 to 6 show an example of a system and method for doing this. Figure 3 shows a computer 20 and a hub 30 plugged into the computer. A master dongle 10 is plugged into a first port 32a on the hub 30. One or more slave dongles 12a-n, generally designated 12, are plugged into other ports 32b-m on the hub 30. The master dongle 10 and slave dongles 12 are hardware encryption tokens with features as described above in relation to the hardware encryption token 103.

The hub 30 may be any standard USB hub, or a hub with another type of port (e.g. FireWire™). Alternatively, the master 10 and slave 12 dongles may be attached directly to ports in the computer 20, without the use of a hub 30. The dongles use the data transfer channel 102 for communication with the hub 30 or the computer 20.

The above described system, comprising the computer 20 and dongles 10, 12, is for securely transferring master keying material, such as a master encryption key, from the master dongle 10 to the slave dongle(s) 12a-n. This allows production of a number of slave dongles that all share the same master keying material. For example, all slave dongles that belong to the same company or working group may use the same master keying material for encryption and decryption when transferring data to one another within the company or working group.

The computer 20 is just one example of a generic data transfer system for connecting the master dongle 10 and the slave dongle(s) 12a-n. However, it is to be understood that other forms of data transfer system may be used.

For example, the data transfer system may be a dedicated device 40 as described below in conjunction with Figure 5.

Alternatively, the data transfer system may be a cable or a hub with no internal processor, in which case one or both of the master and slave dongles 10, 12 may

contain processors configured perform the method steps for securely transferring the master keying material.

When the master dongle 10 is first used, master keying material may be randomly generated by a FIPS approved Random Number Generator and this is stored inside a secure processor in the master dongle 10 in an encrypted form.

Each slave dongle 12a-n has a unique public/private key pair, suitable for use in public key cryptography. As is well understood in public key cryptography, the public key may be publicly shared and seen by others, both friendly and unfriendly. The private key is kept secret and may be used to decrypt data that has been encrypted using the public key.

The computer 20 may run software that is designed for securely transferring the master keying material to each of the slave dongles 12a-n in turn. These slave dongles may then be distributed to e.g. workers in the company or working group for use with their own devices (e.g. PC, laptop, tablet, phone etc.) to allow these devices to securely communicate with one another using the master keying material.

Figure 4 shows a sequence diagram showing the method by which the master keying material may be securely transferred from the master dongle 10 to each slave dongle 12a-n.

Initial steps 201, 202, and 203 are each optional and independent from one another.

In step 201, a PIN or other authentication is entered into the master dongle 10 to unlock the master dongle 10 such that it may send and receive data.

In step 202, a PIN or other authentication is entered into the slave dongle 12 to unlock the slave dongle 12 such that it may send and receive data. Each slave dongle 12a-n may have a unique PIN or other authentication required to unlock the device. Alternatively, a plurality of slave dongles 12a-n may have the same PIN or temporary single-use PIN, for ease of initial setup.

In step 203, a PIN or other authentication is entered into the data transfer system to allow it to transfer data between and/or communicate with the master and slave dongles 10, 12.

Optional step 301: the data transfer system may request the public key from the slave dongle 12. This request may originate from the master dongle 10 (as indicated by the dashed line in Figure 4) or from a computer 20 or dedicated device 40 etc.

Optional step 302: the slave dongle may scramble its public key using a predetermined scrambling algorithm (i.e. such that a scrambled public key is presented to the data transfer system in step 103).

Step 303: The slave dongle's public key (optionally scrambled) is transferred to the data transfer system.

Step 304: the slave dongle's public key is transferred to the master dongle 10.

Optional step 305: if the slave dongle's public key was scrambled at step 302, the master dongle, having been pre-programmed with the scrambling algorithm, unscrambles the public key.

Step 306: the master dongle 10 encrypts the master keying material using the (unscrambled) slave dongle's public key.

Step 307: the encrypted master keying material is transferred from the master dongle 10 to the data transfer system.

Step 308: encrypted master keying material is transferred to the slave dongle 12.

Step 309: the slave dongle decrypts and stores the master keying material using the slave dongle's private key.

According to the method shown in Figure 4, the data transfer system never sees the unencrypted master keying material. Thus, if the data transfer system is compromised by an attacker, the attacker cannot readily discover the master keying material.

The data transfer system may be active in e.g. requesting the slave dongle's public key or may be passive and simply act as a transmission medium for data sent by the dongles 10, 12.

Figure 5 shows an alternative system for securely transferring master keying material from the master dongle 10 to the slave dongle(s) 12a-n. In this system, a dedicated device 40 is provided for performing the above method steps 301-309. That is, the dedicated device 40 is the data transfer system in this example.

The dedicated device 40 contains a plurality of ports 42a-m. A first of the ports 42a is for connecting to the master dongle 10. The remaining ports 42b-m are each for connecting to one of the plurality of slave dongles 12a-n.

The dedicated device further comprises a set of control keys 44, a display device 46, and a control button 48.

The control keys 44 may comprise a numeric keyboard with numbers 0 to 9, a lock button, an unlock button, and an arrow button. The dedicated device 40 may optionally require a numeric code to unlock the device (i.e. step 203) before it will

perform the method steps 301-309. The lock button may be used to re-lock the dedicated device 40 after use, to prevent unauthorized users making their own slave dongles 12.

The display screen may be a LCD screen for displaying messages to a user, such as "Locked", "Unlocked", "Ready to transfer" etc.

Pressing the copy button 48 may cause the dedicated device 40 to start securely transferring the master keying material from a master dongle 10 plugged into the first port 42a to the slave dongle(s) 12a-n plugged into the other port(s) 42b-m.

When multiple slave dongles 12a-n are plugged into the dedicated device 40, the dedicated device 40 may perform the method steps 301-309 for each slave dongle 12 in turn. That is, steps 301 to 309 are performed for a first slave dongle 12a, and once complete, steps 301-309 are performed for a second slave dongle 12b etc.

Similarly, when multiple slave dongles 12a-n are plugged into the computer 20 (or generally, into the data transfer system), the computer 20 may perform the method steps 101-109 for each slave dongle 12 in turn. That is, steps 301 to 309 are performed for a first slave dongle 12a, and once complete, steps 301-309 are performed for a second slave dongle 12b etc.

Once a plurality of slave dongles 12a-n have the master keying material stored thereon, the slave dongles 12a-n may be distributed to a number of users, e.g. employees or member of a working group to allow them to securely exchange data with one another.

Figure 6 shows a schematic diagram in which multiple users may communicate with one another via a network 60 (e.g. via the cloud), using multiple different devices 50a-e. The devices 50a-e may be any of a desktop computer, a tablet computer, a smartphone, a laptop computer and/or any other commonly used computing device capable of connecting to a slave dongle 12.

A respective slave dongle 12a-e is plugged into each device 50a-e. Each slave dongle 12a-e has been prepared with the same master keying material by the method shown above. When a first user 52a wants to share data with another user 52b, the first user 52a instructs his device 50a to send the data to the other user 52b via the network 60. The slave dongle 12a of the first user 52a encrypts the data using the master keying material. The data is then sent via the network 60 to the second user 52b. The second user's device 50b receives the encrypted data and passes it to the second slave dongle 12b. The second slave dongle 12b then decrypts the data using the master keying material and returns the unencrypted data to the second user's device 50b.

In this arrangement, the master keying material is never read by or transferred onto the devices 50a-e, nor is it seen by the network 60. Instead, the master keying material is only ever present on a secure processor in each of the slave dongles 12a-e. Thus, if the network 60 security is compromised, an eavesdropper/hacker will only ever see encrypted data.

Further, if the security of one or more of the devices 50a-e is compromised, the eavesdropper/hacker will only ever see the encrypted and unencrypted data on that device but will not thereby gain access to the master keying material and thus will not be able to compromise the whole network.

Consequently, the compromised device may be repaired/re-secured (e.g. by running an anti-virus program or similar) without requiring that new master keying material is prepared and delivered to each user on a new slave dongle 12.

In an alternative embodiment, not pictured, one of the master dongle 10 or a slave dongle 12 may be configured to perform the method when the two dongles 10, 12 are connected together via e.g. a cable or hub. In this case, the cable is the data transfer system. In this embodiment, all steps requiring a processor are carried out by the master dongle 10 or the slave dongle 12 as appropriate. The method nonetheless works to securely transfer master keying material between master dongle and a slave dongle, as the method does not require processing to be done by the data transfer system. This ensures that an eavesdropper reading signals on the cable never sees the unencrypted master keying material.

CLAIMS:

1. A dongle for ciphering data, the dongle taking the form of a portable hardware device for receiving and transmitting data to and from an external device, and the dongle comprising:

a user interface for interaction with a user to receive authentication data for confirming the identity of a user, wherein the use of the dongle for ciphering data is not enabled unless an authorised user has been identified;

a data transfer channel for coupling the dongle to the external device for transfer of user data between the dongle and the external device; and

a hardware encryption engine for performing a ciphering transformation on user data received by the dongle from the external device;

wherein the dongle is arranged to use the data transfer channel to receive user data from the external device, perform the ciphering transformation on the user data via the hardware encryption engine and, in a continuous operation, return the transformed user data to the external device via the data transfer channel, such that the dongle performs real-time hardware ciphering of user data using a single data transfer channel without storage of the user data on the dongle.

2. A dongle as claimed in claim 1, wherein ciphering transformation uses an encryption key stored securely on the dongle, and the encryption key is accessible only to an authorised user as identified by the user interface.

3. A dongle as claimed in claim 1 or 2, wherein the dongle is arranged such that the data transfer channel is used for continuously receiving data from the external device and sending data to the external device, with the ciphering transformation by the hardware encryption engine occurring directly upon receipt of a data packet from the external device, and the return transmission of a transformed data packet to the external device occurring directly upon completion of the ciphering transformation.

4. A dongle as claimed in claim 1, 2 or 3, wherein the dongle is configured to operate the data transfer channel in sequential steps for transmission of data from the external device to the dongle, and for return of transformed data from the dongle to the external device.

5. A dongle as claimed in claim 1, 2 or 3, wherein the dongle is configured such that there is simultaneous transport of data in both directions via the data transfer channel, wherein the encryption engine is arranged to continuously transform the data whilst the data transfer channel is continuously transporting both incoming data and outgoing transformed data to and from the hardware encryption engine.

6. A dongle as claimed in any preceding claim, wherein the data transfer channel is capable of high speed data transfer rates equivalent to that available via USB 3.0.

7. A dongle as claimed in any preceding claim, wherein the data transfer channel is capable of data transfer at rates of up to at least 5 Gbit/s (625 MB/s).

8. A dongle as claimed in any preceding claim, wherein the data transfer channel includes a USB 3.0 interface, or a successor USB standard.

9. A dongle as claimed in any preceding claim, wherein the dongle is not capable of off-line storage of user data transferred to the dongle from the external device.

10. A dongle as claimed in any preceding claim, wherein the user interface is arranged for interaction with a user to receive authentication data for confirming the identity of the user via an input device enabling the identity of the user to be confirmed via a code or biometric input.

11. A dongle as claimed in any preceding claim, wherein the dongle is arranged to provide real-time ciphering of any type of data, such as voice call encryption, e-mail encryption and/or encryption in relation to software as a service.

12. A dongle as claimed in any preceding claim, wherein multi-factor authentication is required to enable the use of the dongle for ciphering data, with the multifactor authentication including confirmation of the user's identity via the user interface and confirmation of the dongle's identity via dongle authentication data stored on the dongle.

13. A dongle as claimed in any preceding claim, wherein a unique key is stored on the dongle that allows the dongle to function as a factor of authorisation.

14. A dongle as claimed in any preceding claim, wherein the dongle contains public/private key pair for creating a digital signature for messages/documents to verify the creator and its data integrity and/or for creating a hash-based message authentication code for messages/documents to verify the creator and its data integrity; optionally the dongle further contains a certificate for verifying the validity of the dongle.

15. A system for securely communicating data between users, the system comprising multiple dongles as claimed in any preceding claim, wherein each dongle shares a common encryption key.

16. A system as claimed in claim 15 comprising: a first computing device coupled to a first dongle of the multiple dongles, a second computing device coupled to a second dongle of the multiple dongles, wherein, when the first computing device is used to send first data to the second computing device, the first computing device is configured to send the first data to the first dongle, the first dongle is configured to encrypt the first data using the encryption key, and the first device is configured to send the encrypted first data to the second computing device, optionally via a network, and wherein the second computing device is configured to receive the first encrypted data and send the first encrypted data to the second dongle, the second dongle is configured to decrypt the first encrypted data using the encryption key and to send the decrypted first data to the second computer device.

17. A method of ciphering data using the dongle of any of claims 1 to 14, the method comprising receiving authentication data for confirming the identity of a user via the user interface, checking the authentication data or authentication derived data against stored data for authorised users (e.g. a verify a PIN or to perform a biometric identification process), and permitting access to the ciphering function of the dongle only when the received authentication data identifies the user as an authorised user; and, after permitting access to an authorised user, transmitting data from the external device to the dongle via the data transfer interface, performing the ciphering transformation at the dongle using the hardware encryption engine, and returning the transformed data from the dongle to the external device via the data transfer interface.

18. A computer programme product comprising instructions that, when executed, will configure a dongle as claimed in any of claims 1 to 14 to operate in accordance with a method as claimed in claim 17.

19. A method of sharing data securely between authorised computing devices, wherein a first computing device is connected to a first dongle as claimed in any of claims 1 to 14 and a second computing device is connected to a second dongle as claimed in any of claims 1 to 14, the first and second dongles having common encryption keying material, the method of sharing data comprising: encrypting first data using the first dongle with the master keying material stored in the first dongle; transferring the encrypted first data, optionally across a network, to the second computing device; decrypting the encrypted first data using the second dongle; and transferring the unencrypted first data from the second dongle to the second computing device.

20. A method of manufacturing a dongle as claimed in any of claims 1 to 14, wherein the dongle is a slave dongle having master keying material provided from a master dongle, and the method comprises providing the master keying material by: connecting the master dongle and the slave dongle to a data transfer system; transferring the slave dongle's public key to the master dongle via the data transfer system; using the slave dongle's public key at the master dongle to encrypt the master keying material and hence produce encrypted master keying material at the master dongle; transferring the encrypted master keying material to the slave dongle via the data transfer system; decrypting the encrypted master keying material with the slave dongle's private key at the slave dongle; and storing the master keying material at the slave dongle.

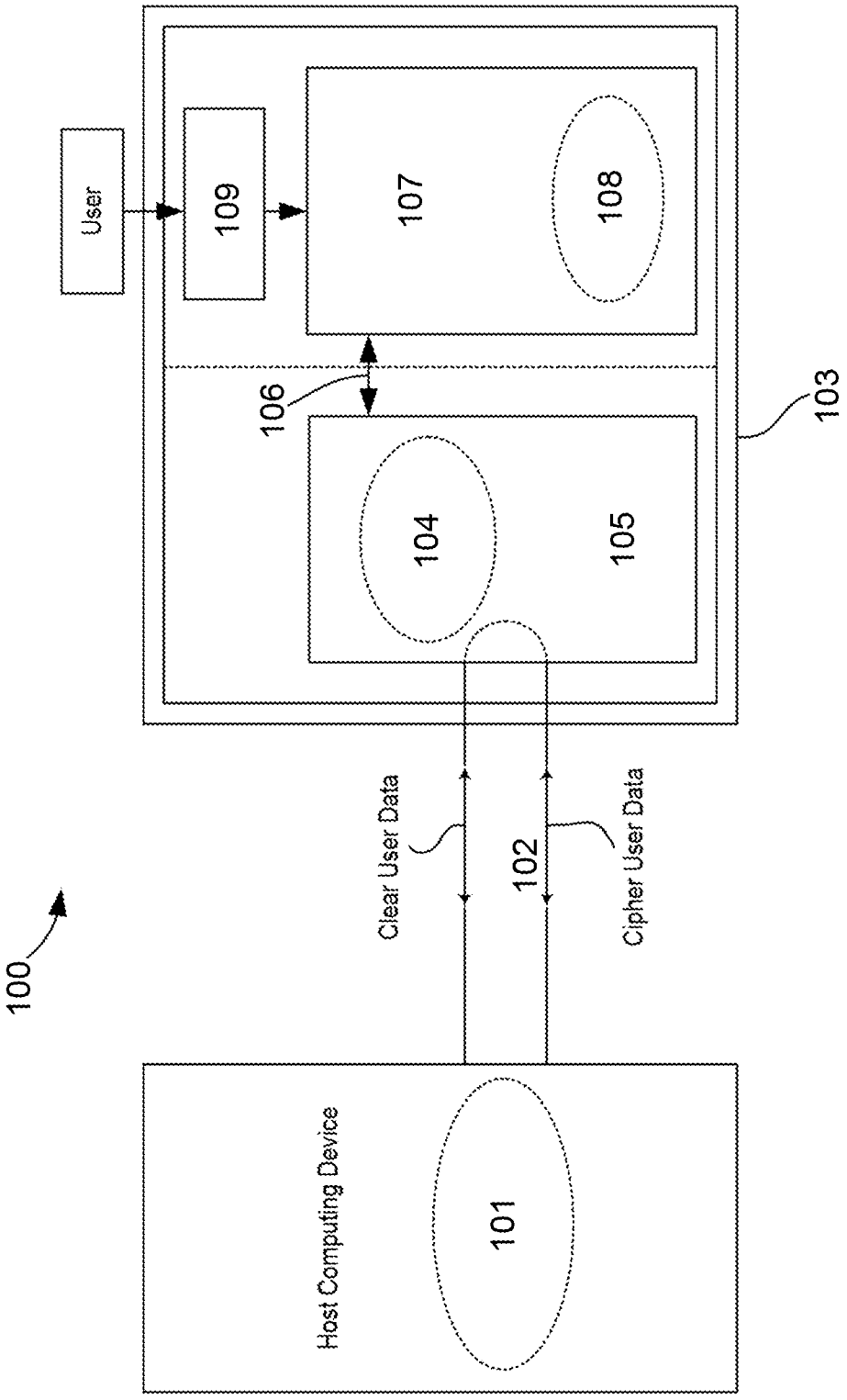

User
109
107
108
106
103
104
105
Clear User Data
102
Cipher User Data
100
Host Computing Device
101

Fig. 1

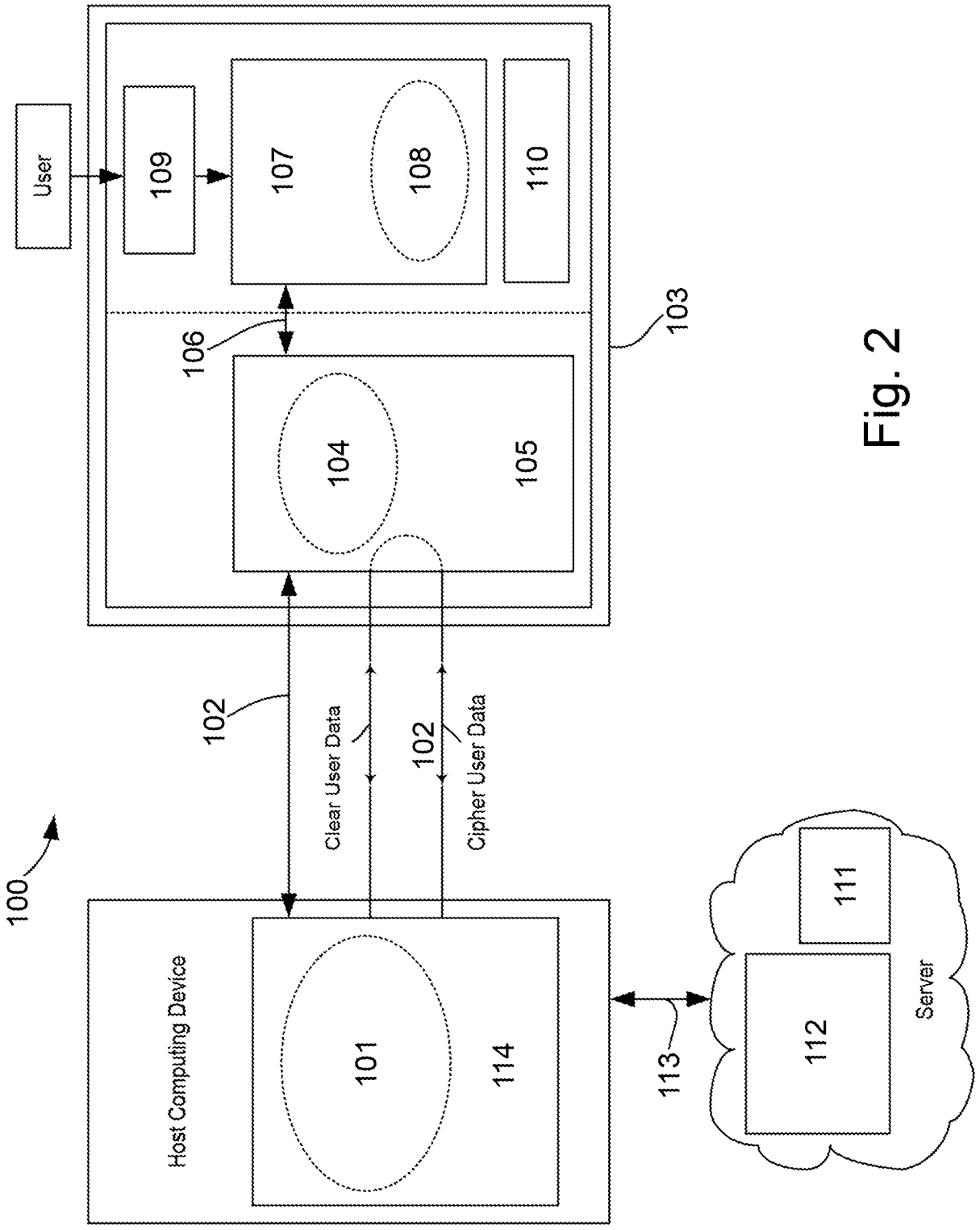

User
109
107
108
110
106
103
104
105
102
Clear User Data
102
Cipher User Data
100
Host Computing Device
101
114
113
111
112
Server

Fig. 2

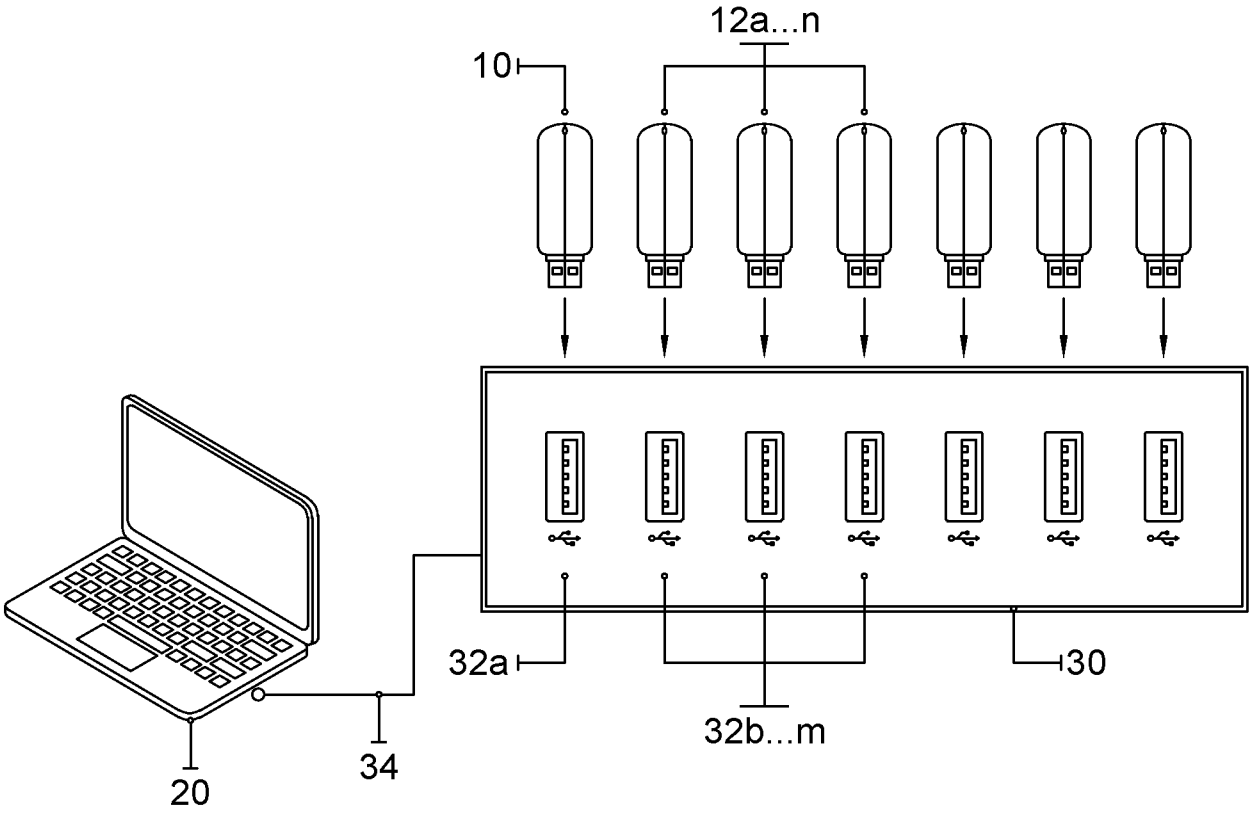


Fig. 3

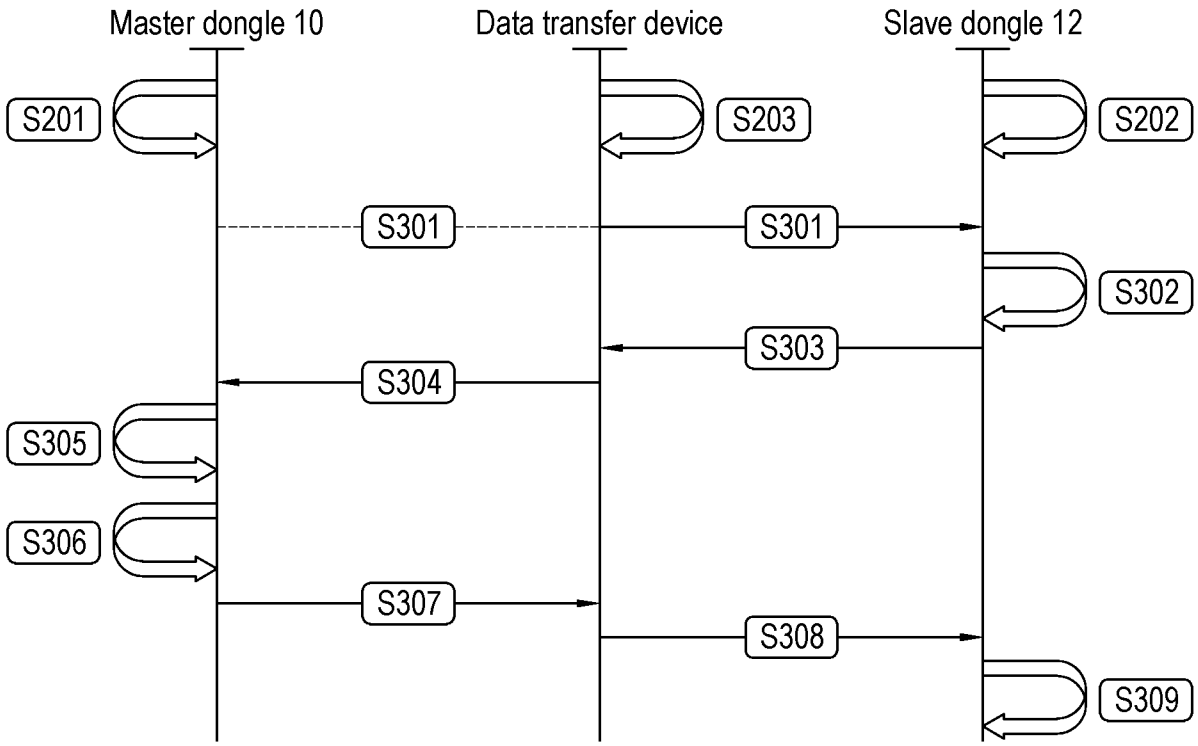

Master dongle 10
Data transfer device
Slave dongle 12
S201
S203
S202
S301
S301
S302
S303
S304
S305
S306
S307
S308
S309

Fig. 4

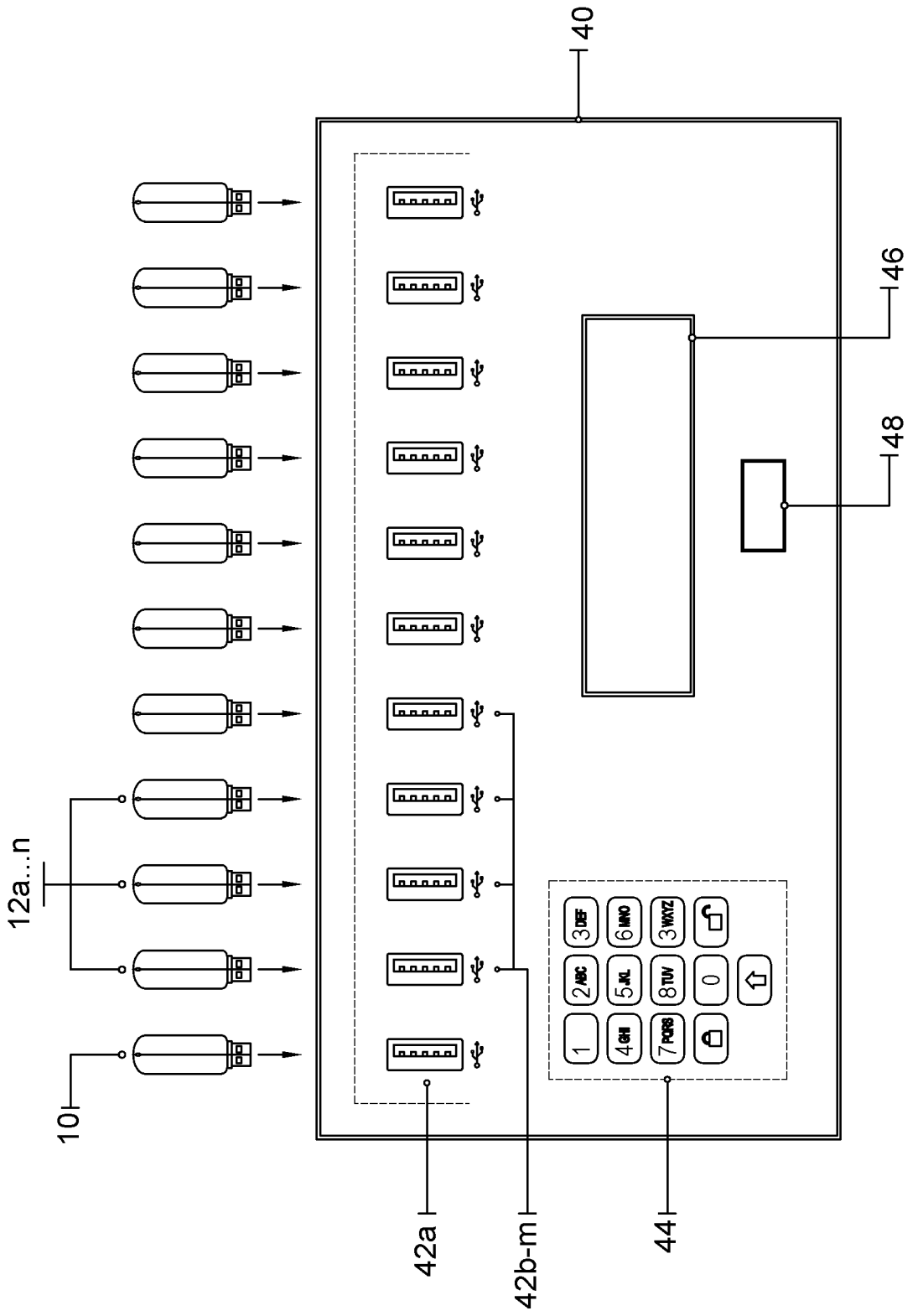


Fig. 5

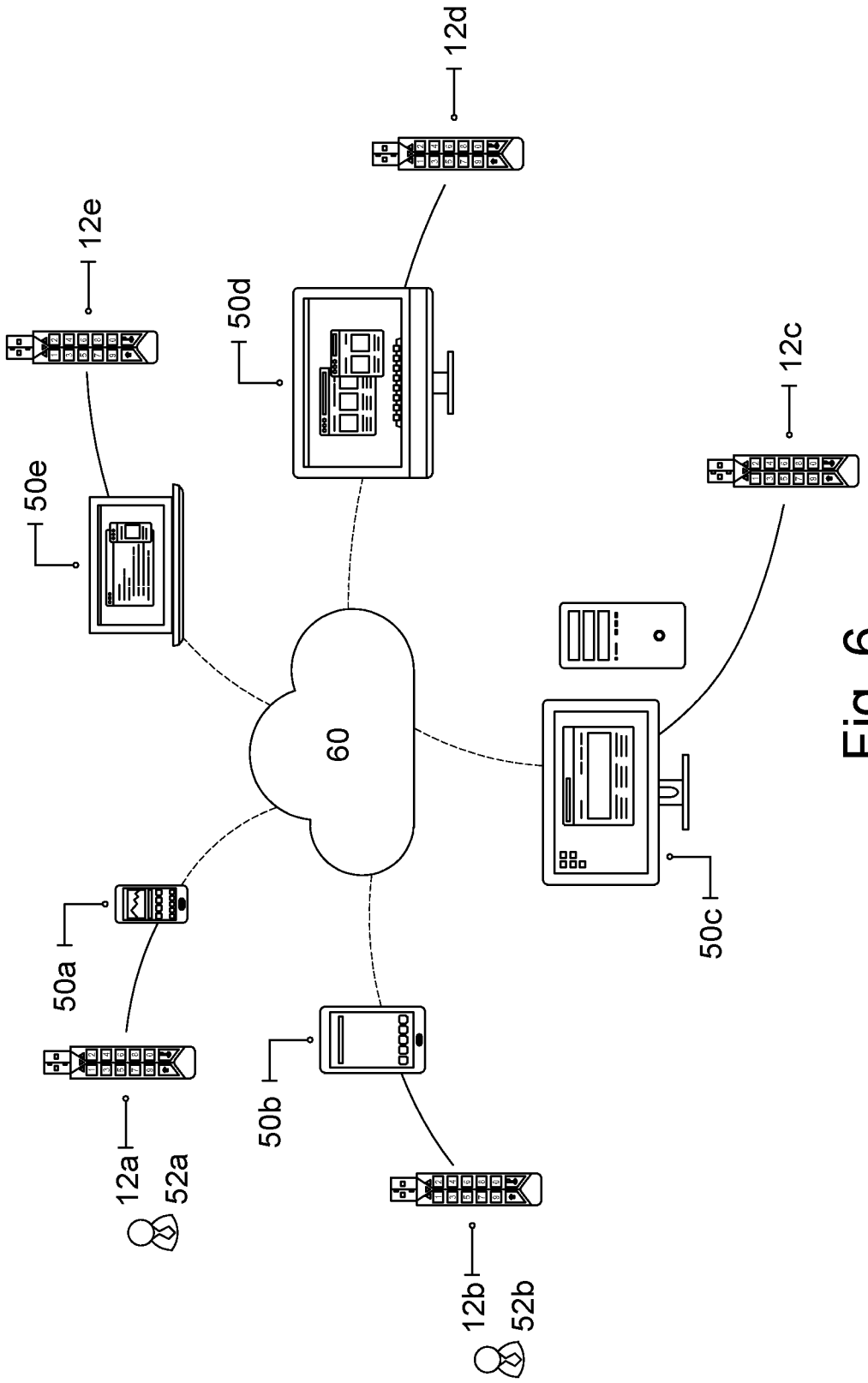


Fig. 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/GB2019/051542

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F21/72 G06F21/34 G06F21/62 G06F21/60
ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 2 902 934 A1 (NXP BV [NL]) 5 August 2015 (2015-08-05) paragraph [0020] - paragraph [0033] figures 1,2	1-19

X	WO 98/07255 A1 (INFORMATION RESOURCE ENGINEERI [US]) 19 February 1998 (1998-02-19) page 8, line 17 - page 9, line 8 page 12, line 15 - line 25 page 14, line 15 - page 18, line 29 page 19, line 22 - page 20, line 29 page 25, line 14 - page 30, line 8; figure 2	1-19

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
9 August 2019	09/10/2019

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

INTERNATIONAL SEARCH REPORT

International application No.
PCT/GB2019/051542

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1-19

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-19

   Apparatus and method for ciphering data

   ---

2. claim: 20

   Method for securely initializing security dongles.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/GB2019/051542

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2902934 A1	05-08-2015	CN 104821878 A	05-08-2015
		EP 2902934 A1	05-08-2015
		US 2015222607 A1	06-08-2015
WO 9807255 A1	19-02-1998	AU 726397 B2	09-11-2000
		CA 2263991 A1	19-02-1998
		EP 0916210 A1	19-05-1999
		US 5778071 A	07-07-1998
		WO 9807255 A1	19-02-1998