

AN OWNERSHIP DATA MANAGEMENT SYSTEM AND METHOD

TECHNICAL FIELD

[0001] This invention relates to an ownership data management system and method. More particularly, this invention relates to an ownership data management system and method for use when trading in collectible items/products.

BACKGROUND

[0002] The following discussion of the background to the invention is intended to facilitate an understanding of the present invention only. It should be appreciated that the discussion is not an acknowledgement or admission that any of the material referred to was published, known or part of the common general knowledge of the person skilled in the art in any jurisdiction as at the priority date of the invention.

[0003] Collectible items/products have been known to be bought and sold. Such items can be expensive, making them a desirable target for counterfeiters. It is not easy for a collector/buyer to distinguish between an original item and a counterfeit one. There can thus be no peace of mind for the collector/buyer when dealing with such collectible items. This may deter some from buying such items. Thus, there is a need for ensuring the originality of the collectible items to create a drive in the market for such items.

[0004] One way of distinguishing between an original item and a counterfeit is to implant an RFID transponder that cannot be easily cloned in the original item. One such RFID transponder is disclosed in European Patent Application EP 2667473 A1 to Barbaric et al. entitled "Production Method, RFID transponder, Authentication Method, reader device and computer program product." The disclosed RFID transponder includes a transponder-specific originality signature that is stored on the transponder by a transponder manufacturer. The transponder-specific originality signature may for example be stored in the nonvolatile memory (EEPROM) of the transponder during the fabrication of the transponder. This transponder-specific originality signature can be checked at any time in a convenient way, which provides an indication of originality of the transponder, and thus the original item.

[0005] Another problem associated with collectible item transactions is the need to verify the ownership of the collectible items, especially during the transfer of a

collectible item from a seller to a buyer. Ownership history has been known to be stored in a computer database which can only be accessed with a proper PIN. However, even with such PIN access, those skilled in the art will know that computer databases can be hacked, and ownership data stored thereon may be compromised.

[0006] To mitigate this problem, different solutions have been proposed. One such solution is disclosed in US 2019/0340623 to Rivkind et al. Rivkind discloses a system for verifying authenticity of products based on proof of ownership and transfer of ownership used in a supply chain application. The system includes a private distributed ledger connected to a public distributed ledger. Although such a system is secure, it can be costly to implement and maintain. It also typically takes a few minutes for a distributed ledger, even a private one, to confirm a transaction; this tends to dampen the user experience when using the system. Furthermore, users of such a system must be members of the private distributed ledger. Although this may suit supply chain members in a commercial setting, it places too much of a burden on casual collectors who trade items only occasionally in a secondary market. Casual collectors also tend to be less tech savvy and sophisticated.

[0007] There is therefore a need for a system which addresses, at least in part, one or more of the foregoing problems.

SUMMARY

[0008] According to an aspect of the present disclosure, there is provided a method of managing ownership data of a product including an electronic tag. The method includes obtaining from the electronic tag a tag-specific identifier, a signature that is generated based on the tag-specific identifier, and a product-specific datum. The method also includes comparing the signature and the product-specific datum with corresponding expected signature and product-specific datum for the product. The method further includes associating the product with a user in an ownership record if the obtained signature and product-specific datum match the expected signature and product-specific datum respectively.

[0009] In some embodiments of the method, the product-specific datum includes a unique product identifier.

[0010] In some embodiments of the method, the unique product identifier includes an SKU and a running number.

[0011] In some embodiments of the method, the product-specific datum further includes a random number.

[0012] In some embodiments of the method, the product-specific datum further comprises a UUID.

[0013] In some embodiments of the method, the product-specific datum from the electronic tag is obtained only after it is determined that there is a match between the obtained signature and the expected signature.

[0014] In some embodiments of the method, the tag-specific identifier, the signature, and the product-specific datum are obtained by a reader in data communication with the product. In these embodiments, the tag-specific identifier and the signature are obtained via a first communication session and the product-specific datum is obtained via a second communication session that is different from the first communication session.

[0015] In some embodiments of the method, the product-specific datum is an encrypted string.

[0016] In some embodiments of the method, the encrypted string is stored in a password-protected memory of the electronic tag. And the method further includes providing a valid password to the electronic tag so as to be able to obtain the encrypted string from the password-protected memory.

[0017] In some embodiments of the method, the ownership record comprises one of an ownership record in a centralized server and a token in a distributed ledger network.

[0018] In some embodiments of the method, the token is a non-fungible token.

[0019] According to another aspect of the present disclosure, there is provided a system for managing ownership data of a product including an electronic tag. The system includes a centralized server that is operable to receive from the electronic tag a tag-specific identifier, a signature that is generated based on the tag-specific identifier, and a product-specific datum. The centralized server is also operable to compare the received signature and product-specific datum with corresponding expected signature and product-specific datum for the product stored thereon. The centralized server is further operable to associate the product with a user in an ownership record on one of the centralized server and a distributed ledger network if

the received signature and product-specific datum match the expected signature and product-specific datum for the product respectively.

[0020] In some embodiments of the system, the product-specific datum includes a unique product identifier.

[0021] In some embodiments of the system, the unique product identifier includes an SKU and a running number.

[0022] In some embodiments of the system, the product-specific datum further includes a random number.

[0023] In some embodiments of the system, the product-specific datum further includes a UUID.

[0024] In some embodiments of the system, the system further includes a distributed ledger network in data communication with the centralized server. The distributed ledger network includes at least two nodes and is operable to execute a smart contract to create a token to associate the product with the user.

[0025] In some embodiments of the system, the system further includes at least one user device that is operable to obtain the tag-specific identifier, the signature, and the product-specific datum from the electronic tag, and send the tag-specific identifier, the signature, and the product-specific datum to the centralized server.

[0026] According to another aspect of the present disclosure, there is provided a program storage device readable by a centralized server, tangibly embodying a program of instructions, executable by the centralized server to perform a method of managing ownership data of a product. The method includes receiving a tag-specific identifier, a signature that is generated based on the tag-specific identifier and a product-specific datum. The method also includes comparing the signature and the product-specific datum with corresponding expected signature and product-specific datum for the product. The method further includes associating the product with a user in an ownership record if the obtained signature and product-specific datum match the expected signature and product-specific datum respectively.

[0027] According to another aspect of the present disclosure, there is provided a method of authenticating a product including an electronic tag. The method includes obtaining a tag-specific identifier from the electronic tag, obtaining a signature from the electronic tag, the signature being generated based on the tag-specific identifier. The method also includes generating an expected signature from the obtained tag-

specific identifier, and comparing the obtained signature with the expected signature. The method further includes providing the tag with a password if the obtained signature matches the expected signature, and obtaining a product-specific datum from the electronic tag. The method yet further includes comparing the obtained product-specific datum with an expected product-specific identifier for the product, and determining that the product is authentic only if the read product-specific datum matches the expected product-specific datum.

[0028] Other aspects and advantages of the invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating by way of example the principles of the invention.

BRIEF DESCRIPTION OF DRAWINGS

[0029] The invention will be better understood with reference to the drawings, in which:

Figure 1 is a block diagram of a system for managing ownership data of a product according to an embodiment of the invention;

Figure 2 is a ladder diagram showing a method of ownership registration in the system in Figure 1;

Figure 3 is a ladder diagram showing a method of ownership transfer in the system in Figure 1;

Figure 4 is a ladder diagram showing a method of proving ownership in the system in Figure 1;

Figure 5 is a flow diagram showing ownership registration by a first collector and ownership transfer between the first collector and a second collector in the system in Figure 1;

Figure 6 is a flow diagram showing a method of verifying the originality of a product in the system in Figure 1 that may be optionally used in any one of the methods in Figures 2-5; and

Figure 7 is a ladder diagram showing a method of verifying the originality of a product and of ownership registration according to another embodiment.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0030] Throughout this document, unless otherwise indicated to the contrary, the terms “comprising”, “consisting of”, “having” and the like, are to be construed as non-exhaustive, or in other words, as meaning “including, but not limited to.”

[0031] Furthermore, throughout the specification, unless the context requires otherwise, the word “include” or variations such as “includes” or “including” will be understood to imply the inclusion of a stated integer or group of integers but not the exclusion of any other integer or group of integers.

[0032] Unless defined otherwise, all technical and scientific terms used herein have the same meaning as is commonly understood by a skilled person to which the subject matter herein belongs.

[0033] As shown in the drawings for purposes of illustration, the invention may be embodied in a novel, user-friendly, reasonably secure and trustworthy product ownership data management system. Existing systems tend to be either not tamper-proof or tamper-proof but diminishing the user experience when using the systems. Referring to Figure 1, an ownership data management system embodying the invention generally includes a centralized server that is operable to receive at least one ownership-related request from a user device, create at least one ownership record corresponding to the at least one ownership related request thereon, and send a request to a distributed ledger network to store the at least one ownership record on the distributed ledger network.

[0034] Specifically, Figure 1 shows a system 10 for managing ownership data related to collectible products, such as toys, antiques, art pieces, wines, etc. according to an embodiment of the invention. However, the system 10 may also be used for managing ownership data of products other than collectible products. The system 10 includes at least one user device 12, a centralized server 14 and a distributed ledger network 16. The centralized server 14 is data communicatively coupled, via the Internet (not shown), to the user device 12 and the distributed ledger network 16. The centralized server 14 is typically built around a single server 14 that handles all processing. The centralized server 14 is controlled by a single administrative entity, such as a collectible product business owner. It operates based on a client-server architecture. The centralized server 14 includes at least one processing unit/processor (not shown) and a program storage device (not shown)

readable by the processing unit that tangibly embodies a program of instructions, executable by the processing unit to provide services of the centralized server 14.

[0035] The centralized server 14 includes an application programming interface (API) gateway 18, a sync service module 20 and a database 22. The centralized server 14 provides ownership-related data capturing services to the users of the system 10 via an App (not shown) running on user devices 12, such as mobile devices 12, of the users. The App is a software application that is downloaded and installed on the users' mobile devices 12. The App pulls content and data from the centralized server 14 through the Internet. In addition to the App, the users may also interact with the centralized server 14 via a collector web portal 24 and a collector mobile portal 26 supported by the centralized server 14. These portals are private locations on the Internet, accessible with unique URLs (web addresses) and unique usernames and passwords. However, the services may also be further provided via websites and mobile websites. Suitable graphical user interfaces (GUIs) are provided on the user devices 12 to enable the users to access the services of the centralized server 14. The collector web portal 24 therefore, is preferably accessible via a web browser on the user device 12 and provides a page on the world wide web, or another access point where a collector can engage with the system 10, for example for the collector to register ownership and transfer ownership of a collectible product. Similarly, the business owner also has a business owner web portal 28 and a business owner mobile portal 30 for writing data to and reading data from a product table 32, a customer table 34 and an ownership table 36 in the database 112, and for making requests, sending data, confirming data, and the like. In addition to the App, these portals are alternative/additional entry points into the system 10 that provide a location for creating and maintaining product and ownership-related data. Partners of the business owner may also access the centralized server 14 via partner web portals 40 and partner mobile portals 42 to provide value added contents associated with the products offered by the business owner.

[0036] As mentioned above, user devices 12, such as but not limited to mobile devices, laptops and personal computers with a suitable user interface are used to access the portals. Each user device 12 may include at least one processing unit and a form of memory. The API gateway 18 on the centralized server 14 allows communication between the user devices 12 and the services provided by the

centralized server 14. An identity services module (not shown) manages an identity for each collector, business owner and partner that logs into the centralized server 14. The identity services module stores or validates data about the collectors, the business owner and the partners. This identity can be utilized to confirm and register a user, in certain applications of the system 10. Thus, to confirm identity, the identity services module can be utilized to confirm the data about the user prior to allowing the user to access services provided by the centralized server 14.

[0037] During use of the system 10, the business owner registers product information on the centralized server 14 via an App or a suitable portal. For example, the product information may include unique product IDs (PID) and detailed information about each product. This product information is stored in the product table 32 in the database 22 of the centralized server 14. For each original product, the business owner may request a unique product ID (PID) from the centralized server 14 for assigning to the product. The unique product ID (PID) is encrypted and stored in an electronic tag, such as but not limited to, a near field communication (NFC) chip (not shown). The NFC chip also carries an NFC chip unique identifier (UDID) 50 (Figure 6), a digital signature 52 and a 32-bit password 54. Once the NFC chip is programmed with the unique product ID, the NFC chip is attached to or embedded in the original product. Instead of the business owner uploading a list of unique product IDs (PIDs) onto the centralized server 14, the centralized server 14 may generate the unique product IDs (PIDs) as described above. When no product has yet been sold, the ownership table 36 will be empty. Once the products are sold, the system 10 would allow ownership registration, query of ownership data and transfer of ownership. Therefore, the centralized server 14 allows for storage of product data, ownership data, and customer data in the database 22 thereof. The methods for ownership registration and ownership transfer will be described in more details later. For the time being, it is to be noted that ownership records (not shown) are created during ownership registration and ownership transfer and stored in the ownership table 36 in the database 22.

[0038] The centralized server 14 uploads a datum associated with ownership records created and stored thereon at regular intervals onto the distributed ledger network 16 using the sync service module 20. In other words, the centralized server 14 keeps track of ownership records created during an interval, generates a datum

associated with these ownership records at the end of the interval and sends the generated datum to the distributed ledger network 16 for storage thereon such that a copy of the datum is maintained on the distributed ledger network 16. The intervals for sending these data may be twenty-four-hour intervals. However, the interval may also be shorter or longer than twenty-four hours. The datum is stored in a block on the distributed ledger network 104 maintained by nodes 60 of the distributed ledger network 16.

[0039] The distributed ledger network 16 can be based on any distributed ledger technology (DLT) implementation e.g. Ethereum, Quorum, Hyperledger, R3 Corda, or another private or public distributed ledger network 16. As is known to those skilled in the art, the distributed ledger network 16 includes a plurality of nodes 60 associated with different members of the network 16. This plurality of nodes 60 cooperate to provide the required data protection and trust. It allows the data to be stored and securely managed by the members without a central authority. Any attempt to falsify data on the distributed ledger network 16 will be detected and rejected by the members of the distributed ledger network 16, who independently validate and control the correct version of the data. In this manner, ownership records can be registered/stored (“notarized”) on every node 60 of the distributed ledger network 16 and this allows a seller to prove ownership of the product to a buyer.

[0040] The business owner and the users/collectors can examine the full history of ownership of a product using the centralized server 14. This provides a powerful and clear chain of title to a buyer who is interested in buying the product to be confident that he is trading in an original product and ultimately this gives collectors confidence in the originality and/or quality of the products.

[0041] A method 70 of ownership registration using the system 10 is next described with the aid of Figure 2. When a first collector (buyer) 72 purchases a product from the business owner, the buyer 72 may register ownership of the product with the system 10. The ownership registration method 70 starts in a SCAN PRODUCT step 74, wherein the buyer 72 runs the App on his mobile device 12. This step 74 allows the mobile device 12 to be used to scan the NFC chip of the purchased product to obtain at least the ID of the product (PID) stored on the NFC chip. The ownership registration method 70 next proceeds to a CHECK PRODUCT INFO step 76, wherein communication between the centralized server 14 and the

mobile device 12 is established to allow the centralized server 102 to receive information about the NFC chip and verify the originality of the NFC chip, and thus, the originality of the product to which the NFC chip is permanently attached to or embedded in. The verification of originality in this CHECK PRODUCT INFO step 76 will be described in greater detail later. The method 70 ends in a RECEIVE PRODUCT INFO step 78 if the centralized server 14 is unable to verify that the NFC chip is original. In this step 78, the App will indicate to the buyer that the product may be counterfeit. However, if the centralized server 14 is able to verify that the NFC chip is original in the CHECK PRODUCT INFO step 76, the method 70 proceeds to the RECEIVE PRODUCT INFO step 78, wherein the App will indicate to the buyer 72 that the product is original. The method 70 next proceeds to an INITIATE OWNERSHIP REGISTRATION step 80, wherein the buyer 72 is allowed to actuate an appropriate icon on a screen of his mobile device 12 to initiate ownership registration with the system 10. In this step 80, the mobile device 12 sends an OWNERSHIP REGISTRATION REQUEST message 82 to the centralized server 14. The OWNERSHIP REGISTRATION REQUEST message 82 includes the PID and the buyer's collector identity (CID).

[0042] The method 70 further proceeds to a RECEIVE OWNERSHIP REGISTRATION REQUEST step 84, wherein the centralized server 14 receives and processes the OWNERSHIP REGISTRATION REQUEST message 82 from the buyer's 72 mobile device 12. The method 70 further proceeds to an UPDATE NEW OWNER step 86, wherein the centralized server 14, after determining that originality of the NFC chip had been verified earlier, creates an ownership record in the ownership table 36 of the database 22. This ownership record includes the PID and the CID in the OWNERSHIP REGISTRATION REQUEST message 82, and a status field that is set to TO-BE-UPLOADED. This status indicates that the particular record has yet to be uploaded onto the distributed ledger network 16. The ownership record may also, optionally, include other information, such as but not limited to, type/class of product, date and time of the ownership registration, etc.

[0043] In this manner, buyers 72 who purchased products from the business owner may each register ownership of the product they bought, and have respective ownership records created in the ownership table 36 in the database 22 of the centralized server 14. As mentioned above, each of these ownership records created

in an interval would have a status field indicating that the records have yet to be uploaded onto the distributed ledger network 16. In this UPDATE NEW OWNER step 86, the centralized server 14 also sends a first OWNERSHIP REGISTRATION CONFIRMATION message 88 to the mobile device 12 to indicate to the buyer 72 that ownership registration on the centralized server 14 is confirmed. The mobile device 12 receives this OWNERSHIP REGISTRATION CONFIRMATION message 88 in a CONFIRMATION OF REQUEST step 90.

[0044] At the end of the interval, such as after twenty-four hours as described above, the centralized server 14 extracts all records with a status field of TO-BE-UPLOADED from the ownership table 36. The centralized server 14 generates a datum associated with these records and sends an UPLOAD REQUEST message 92 that includes the datum to a public address of a smart contract 100 of the distributed ledger network 16. The datum may be a Merkel root of a Merkel tree of hashes. Each ownership record is cryptographically hashed to obtain a first hash value for the record. Next pairs of first hash values are cryptographically hashed to obtain a second hash value for each pair of first hash values so that the number of second hash values is half that of the first hash values. The process is repeated with the second hash values to obtain third hash values. At the end of such a process, there will be a single final hash value which is known as the Merkel root or root hash. Hashing of an ownership record is applying a function that maps the data of the ownership record to a fixed length output. A hashing algorithm in the function is applied to the input data and the resulting fixed length output is the hash. Many hashing algorithms are available; examples of which include but are not limited to the SHA-1, SHA-2, SHA-3, SHA-256, MD5, RIPEMD-160, Whirlpool, Blake2, Blake3 algorithms. The resulting hash is completely unique to the input data and the function itself is deterministic. By obtaining a Merkel root for all the ownership records created in a twenty-four hour period, huge amounts of data can be identified solely through the Merkel root. The use of the Merkel root creates vast storage savings and helps to increase efficiency. The Merkel root or root hash can be used as a fingerprint for all the ownership records created in the twenty-four hour period. As long as the Merkel root is publicly known and trusted, it is possible to use it to prove integrity of the ownership records.

[0045] Other methods of obtaining the datum are also possible. These includes, but are not limited to, checksums and CRCs obtained based on the ownership records.

[0046] The centralized server 14 associates each ownership record with the final hash value and sends this final hash value in the UPLOAD REQUEST message 92 to the distributed ledger network 16 for storing the final hash value in a block of the distributed ledger network 16. The method 70 next proceeds to a RECEIVE DATUM step 120 wherein the UPLOAD REQUEST message 92 is received by a node 60 in the distributed ledger network 16 to thereby hit that node 60 as a blockchain transaction request. The node 60 registers and propagates the blockchain transaction request to other nodes 60 of the distributed ledger network 16. The transaction is processed by every node 60 independently. The final state (datum update) is validated and synchronized with a consensus protocol in a CONSENSUS BETWEEN NODES IN NETWORK step 122. The consensus protocol is used to validate and synchronize the data (the state) between the nodes 60; it makes sure all nodes 60 are in sync. The distributed ledger network 16 may use different kinds of consensus protocols as is known to those skilled in the art. A consensus simply means that all nodes 60 compare data and each agrees with the data regarding that transaction. Transaction processing also triggers the execution of the smart contract 100 in an UPDATE TRIGGERS EXECUTION OF SMART CONTRACT step 124, wherein members of the network 16 sign off the transaction. The method 70 next proceeds to a BLOCKCHAIN PLATFORM RECEIVES CONFIRMATION OF OWNERSHIP REGISTRATION step 126 via a REGISTRATION OF OWNERSHIP IS CONFIRMED step 128, wherein the distributed ledger network 16 prepares a block ID (transaction hash) of the block in the distributed ledger network 16 in which the hash value is stored.

[0047] After the UPDATE NEW OWNER step 86, the centralized server 14 checks regularly with the distributed ledger network 16 for the availability of the block ID (transaction hash). The method 70 next proceeds to a SERVER OBTAINS CONFIRMATION step 132 at the centralized server 14, wherein upon obtaining of the confirmation message 130 from the distributed ledger network 16, the centralized server 14 changes the status of the records marked TO-BE-UPLOADED to UPLOADED to indicate the completion of the uploading of the hash value associated

with the records created in the interval. The centralized server 14 also links each of these records or the hash value stored thereon to the block ID (transaction hash) obtained from the distributed ledger network 16. The centralized server 14 sends a second confirmation message 134 to the mobile device 12. The method 70 finally ends in a COLLECTOR RECEIVES CONFIRMATION step 136 at the mobile device 12, wherein the mobile device 12 obtains a confirmation that the ownership registration is finally completed at the distributed ledger network 16, and proof of ownership or transfer of ownership can be carried out next by the buyer 72 as a rightful owner of the product.

[0048] After the buyer 72 has registered ownership of the product with the system 10 as described above, the buyer 72 becomes the owner of the product. The owner 72 may now decide to sell the product to another buyer 140 (Figure 4) via some online platforms such as Ebay etc. Such a sale will require the ownership data in the system 10 to be updated to reflect the transfer of ownership from the owner 72 to the new buyer 140.

[0049] A method 150 of ownership transfer according to an embodiment of the invention is next described with the aid of Figure 3. The ownership transfer method 150 starts in the SCAN PRODUCT step 74, wherein the owner 72 runs the App on his mobile device 12. This step 74 allows the mobile device 12 to be used to scan the NFC chip of the product he purchased to obtain an ID of the product (PID). The ownership transfer method 150 further proceeds to the CHECK PRODUCT INFO step 76, wherein communication between the centralized server 14 and the mobile device 12 is established to allow the centralized server 14 to receive information of the NFC chip and verify the originality of the NFC chip, and thus, the originality of the product to which the NFC chip is permanently attached to or embedded in. This verification of originality in the CHECK PRODUCT INFO step 76 will be described in greater detail later. The method 150 ends in the RECEIVE PRODUCT INFO step 78 if the centralized server 14 is unable to verify that the NFC chip is original. In this step 78, the App will indicate to the buyer that the product may be counterfeit. However, if the centralized server 14 is able to verify that the NFC chip is original in the CHECK PRODUCT INFO step 76, the method 150 proceeds to an INITIATE OWNERSHIP TRANSFER step 152, wherein the owner 72 is allowed to actuate an appropriate icon on a screen of his mobile device 12 to initiate ownership transfer. In this step 152,

the mobile device 12 sends an OWNERSHIP TRANSFER REQUEST message 154 to the centralized server 14. The OWNERSHIP TRANSFER REQUEST message 154 includes the PID of the product and CID of the new buyer 140. The method 150 further proceeds to a RECEIVE OWNERSHIP TRANSFER step 160 wherein the centralized server 14 receives and processes the OWNERSHIP TRANSFER REQUEST message 154 from the owner's 72 mobile device 12. The method 150 further proceeds to an UPDATE NEW OWNER step 162, wherein the centralized server 14, after determining that originality of the NFC chip had been verified earlier in the CHECK PRODUCT INFO step 76, creates another ownership record in the ownership table 36 of the database 22.

[0050] The centralized server 14 updates the status of this ownership record as WAITING FOR CONFIRMATION and sends a CONFIRMATION REQUEST message 164 to the new buyer 140. The method 150 next proceeds to a RECEIVE REQUEST TO TRANSFER OWNERSHIP step 166 at a mobile device 12 of the new buyer 140. In this step 166, the new buyer 140 is notified that the owner 72 of the product has initiated transfer of ownership of the product to the new buyer 140 and requires him to make a payment and to accept the transfer.

[0051] The method 150 next proceeds to a MAKE PAYMENT step 170, wherein the new buyer 140 makes an electronic payment to the centralized server 14. The centralized server 14 receives this payment in a RECEIVE PAYMENT step 172, and sends a first confirmation message 174 to the owner 72. The owner 72 receives the first confirmation message 174 in a CONFIRMATION OF REQUEST step 175 to be informed that payment has been made by the new buyer 140 that is held at the centralized server 14. The owner 72 then sends the product, via courier or post, to the new buyer 140 in a SEND PRODUCT step 176. When the new buyer receives the product in a RECEIVE PRODUCT step 178, the new buyer 140 uses the App in his mobile device 12 to scan the NFC chip in the received product to retrieve the PID of the product for verification of the originality of the product as described above. Once it is verified that the product is original, the new buyer 140 is allowed by the centralized server 14 to accept the transfer of ownership in a CONFIRM CHANGE OF OWNERSHIP step 180 at the mobile device 12. The new buyer 140 accepts the transfer of ownership by actuating an appropriate icon in the App to send a CONFIRMATION ACCEPTED message 182 to the centralized server 14. This

CONFIRMATION ACCEPTED message 182 includes the PID of the received product and the CID of the new buyer 140. The method 150 next proceeds to an UPDATE CHANGE OF OWNERSHIP step 184 in the centralized server 14, wherein the centralized server 14 retrieves the newly created record based on the PID and updates it with the CID of the new buyer 140. The centralized server 14 further changes the status of the record from WAITING FOR CONFIRMATION to TO-BE-UPLOADED so that at the end of the current interval, this record and other similarly marked records can be retrieved for a datum to be generated therefrom and uploaded to the distributed ledger network 104.

[0052] The ownership record for this ownership transfer transaction has exactly the same fields as that of the ownership record described in the ownership registration method 70 described above. However, the ownership record created during an ownership transfer may be of a different format than that used in ownership registration.

[0053] At the end of the interval, the centralized server 14 extracts all records in the ownership table 36 with a status of TO-BE-UPLOADED. The centralized server 14 generates a datum associated with these records, associates each of these ownership records with the datum and sends an UPLOAD REQUEST message 92 that includes the datum to a public address of the smart contract 100 of the distributed ledger network 16. As described above in relation to ownership registration, the centralized server 14 obtains a single hash value as the datum based on all the records and sends this single hash value to the network 16 for storing in a block thereof. The UPLOAD REQUEST message 92 is received by a node 60 of the network 16 in the RECEIVE DATUM step 120 to thereby hit that node 60 as a blockchain transaction request. The node 60 registers and propagates the blockchain transaction request to other nodes 60 of the distributed ledger network 16. The transaction is processed by every node 60 independently. The final state (datum update) is validated and synchronized with a consensus protocol in the CONSENSUS BETWEEN NODES IN NETWORK step 122. Transaction processing also triggers the execution of the smart contract 100 in the UPDATE TRIGGERS EXECUTION OF SMART CONTRACT step 124, wherein members of the network 16 sign off the transaction. The method 150 next proceeds to the BLOCKCHAIN PLATFORM RECEIVES CONFIRMATION OF OWNERSHIP TRANSFER step 190 via the

TRANSFER OF OWNERSHIP IS CONFIRMED step 192 wherein the distributed ledger network 16 prepares a block ID (transaction hash) of the block in the distributed ledger network 16 in which the hash value is stored .

[0054] After the UPDATE CHANGE OF OWNERSHIP step 184, the centralized server 14 checks regularly with the distributed ledger network 16 for the availability of the block ID (transaction hash). The method 150 next proceeds to a SERVER OBTAINS CONFIRMATION step 196 at the centralized server 14, wherein upon obtaining of the confirmation message 194 from the distributed ledger network 16, the centralized server 14 changes the status of the records marked TO-BE-UPLOADED to UPLOADED-COMPLETED to indicate the completion of the uploading of the hash value associated with the records created in the interval. The centralized server 14 also links each of these records or the hash value stored thereon to the block ID (transaction hash) obtained from the distributed ledger network 16. The centralized server 14 sends an OWNERSHIP TRANSFER CONFIRMATION message 198 to both the owner's mobile device 12 the new buyer's mobile device 12. The method 150 finally ends in a respective COLLECTOR RECEIVES CONFIRMATION step 200 at the mobile devices 12 of the owner 72 and the new buyer 140, wherein the mobile devices 12 obtain a confirmation from the system 10 that the ownership transfer between the owner 72 and the new buyer 140 has been completed. The centralized server 14 also releases payment it received earlier from the buyer 140 to the owner 72.

[0055] It is possible before the new buyer 140 agrees to buy the product from the owner 72 of the product, the new buyer 140 may request that the owner prove that he is the rightful owner of the product. A method 210 of proving ownership 210 is next described with the aid of Figure 4. In the method 210, the new buyer 140 requests the owner 72 to prove ownership in a REQUEST FOR PROOF step 212. The owner 72 scans the NFC chip of the product in the SCAN PRODUCT step 74 as described above. The owner 72 then actuates an appropriate icon on his mobile device 12 to send an OWNERSHIP DATA VERIFICATION message 214 to the centralized server 14. This message 214 includes the PID of the product and the CID of the owner 72. The centralized server 14 upon receiving this OWNERSHIP DATA VERIFICATION message 214 in a RECEIVE VERIFICATION REQUEST step 216 extracts the latest ownership record associated with the PID and compares the CID stored therein with

that received. If the two CIDs match, the centralized server 14 obtains the block ID (transaction hash) associated with the record or hash value and forwards the block ID (transaction hash) to the distributed ledger network 16. Upon receiving the OWNERSHIP DATA VERIFICATION message 214 in a RECEIVE VERIFICATION REQUEST step 218, the distributed ledger network 16 retrieves the datum/hash value based on the block ID (transaction hash). The distributed ledger network 16 next returns the datum/hash value to the centralized server 14 in an OWNERSHIP DATA RESPONSE message 220. The centralized server 14 then sends a result of the CID comparison, the hash value stored thereon and the hash value stored on and obtained from the distributed ledger network to the owner's mobile device 12 in a RECEIVE OWNERSHIP STATUS RESPONSE step 222. The owner 72 upon receiving this OWNERSHIP DATA RESPONSE message 220 in a COLLECTOR CONFIRMS OWNERSHIP step 224 can then present the result to the new buyer 140 to prove his ownership in a SHOW OWNERSHIP STATUS step 226. With two matching hash values, one from the centralized server 14 and one from the trusted distributed ledger network 16, confidence in the ownership information returned by the centralized server 14 will be enhanced.

[0056] Figure 5 is a drawing showing the change of ownership of a product from the business owner to a collector A and finally to a collector B. After collector A purchases the product and has registered ownership 70 with the system 10 as described above, collector A is given access 250 to exclusive content by the system 10. This exclusive content includes, but is not limited to, videos, fan engagement, greetings from favorite artists who designed the products, etc. as a value added service provided by the business owner.

[0057] Figure 6 shows a method 260 of verifying originality of a product as carried out in the CHECK PRODUCT INFO step 76. As described above, when the user uses the App to scan the NFC chip of the product, the mobile device 12 obtains a 32-bit password 54 from the centralized server 14 and sends this 32-bit password 54 to the NFC chip. The NFC chip verifies that the 32-bit password 54 received from the mobile device 12 matches that stored in the NFC chip. The NFC chip next returns to the mobile device 12 an NFC chip unique ID (UDID) 50, a digital signature 52 and an encrypted string 268 containing a product ID (PID). The mobile device 12 forwards these three pieces of information to the centralized server 14. In a first test 269, the

centralized server 14 verifies the NFC chip UDID 50 and the digital signature 52 in a verification algorithm 270 using a public key 272 provided by the manufacturer of the NFC chip.

[0058] The centralized server 14 also decrypts the encrypted string 268 received from the NFC chip using a private key 274. In a second test 275, the PID obtained from the decrypted string is compared to those stored in the product table 32 of the database 22. The centralized server 14 will determine that the product is original if the information returned from the NFC chip passes both the first test 269 and the second test 275 in less than a predetermined number of attempts.

[0059] If the information fails either of the two tests 269, 275 in an attempt, the process of obtaining the information and verification is repeated and the number of attempts is counted. If the number of attempts exceeds the predetermined number of attempts, the centralized server 14 will indicate that the product may be counterfeit and/or there may be a fraudulent activity and forbid the user from proceeding further. On the other hand, if the centralized server 14 determines that the information is verified in less than the predetermined number of attempts, the centralized server 14 will allow the user to proceed with ownership registration, ownership transfer, ownership acceptance, etc. as described above.

[0060] As shown in the drawings for purposes of illustration, the invention may be further embodied in a novel and secure method for verifying originality of a product and managing ownership data of the product. Existing methods tend to be less secure and less tamper-resistant. Referring to Figure 7, the method includes obtaining from an electronic tag of the product, a tag-specific identifier, a signature that is generated based on the tag-specific identifier, and a product-specific datum. The method further includes comparing the signature and the product-specific datum with corresponding expected signature and product-specific datum for the product. The method further includes associating the product with a user in an ownership record if the obtained signature and product-specific datum match the expected signature and product-specific datum respectively.

[0061] More specifically, Figure 7 illustrates a method 290 of verifying originality of a product 300 according to another embodiment of the present disclosure. As shown in Figure 7, the product 300 includes an embedded NFC chip 302 although other ICs supporting wired or wireless communication with a reader may be used. An example

of such a chip is the NXP NTAG21x NFC tag IC that complies to the NFC Forum Type 2 Tag and ISO/IEC14443 Type A specifications.

[0062] The NFC tag 302 has a tag-specific identifier, such as but not limited to, a unique identifier (UID) 50 of the tag 302. This UID may be a universally unique identifier (UUID). The manufacturer generates a signature from the UID using a private key of a private-public key pair according to an asymmetric cryptographic method, such as Elliptic Curve Cryptography (ECC) or another asymmetric cryptographic method, including but not limited to, RSA and NTRU. The UID 50 may also be signed by means of a digital signature in accordance with the Digital Signature Standard (DSS). Those skilled in the art will appreciate that other tag-specific identifiers may be used as well for generating the signature.

[0063] The use of asymmetric cryptography with public keys improves the reliability because the public keys cannot be used for the signing. The private key used for the signing remains stored in a secured environment at the NFC tag manufacturer site. The use of asymmetric cryptography also makes it possible to generate different public keys for different customers and to keep a single private key at the NFC tag manufacturer site. This feature contributes significantly to the relatively low cost of the fabricated NFC tag.

[0064] The NFC tag 302 also has a memory (not shown) that is password protected. In addition to the signature, the tag manufacturer may also assign and store a customer-specific password on the NFC tag 302, and provide the assigned password to the customer who is the manufacturer of the product 300. Alternatively, the customer may write its own password in the NFC tag. The password may be tag dependent or common to all tags of the customer.

[0065] During production of the product 300, the product manufacturer writes a product-specific datum (not shown) in the password-protected memory. This product-specific datum may include, but not limited to, an encrypted string that is made up of a product SKU, a random number, and a running number. The SKU number is assigned to a product in order to identify one or more of the brand, model, colour, size, etc. to indicate different variations of a single product. SKUs are unique to the product manufacturers and are created by them. The SKUs are therefore not universal; and each product manufacturer has their own set of SKUs for their products. The running number is for example an integer that is increased for each

new product item. Therefore, the SKU is used for identifying the type of the product. And the random number and running number are used for identifying a product of the product type. The random number may be a random password which is a cryptographically strong pseudo-random data that is 4 bytes in length. In an alternative embodiment, the encrypted string may further include an RFC-4122 (version 4) compliant random universally unique identifier (UUID). In other words, the encrypted string includes a UUID, random password, product SKU and a running number in that sequence. To generate this encrypted string, a RFC-4122 compliant random UUID is generated and a random password created. A string including the UUID, random password, product SKU and running number is then created. Finally, an encryption system, such as but not limited to, the RSA cryptosystem is used to encrypt the string with a 2048-bit private key to produce the encrypted string.

[0066] The method 290 starts in a LAUNCH APP step 304, wherein a user 72 launches an ownership management app on his mobile device 12. The mobile device 12 includes an NFC tag reader/scanner (not shown). In this step, the user 72 will be prompted by a user interface of the app to read the product 300 by bringing the mobile device 12 in close proximity, e.g. within 10cm, of the product 300 so that communication between the mobile device 12 and the NFC tag 302 embedded in the product 300 can be established for the app to authenticate the NFC tag 302 and thereby the product 300.

[0067] The method 290 next proceeds to an INITIATE COMMUNICATION step 306, wherein the mobile device 12 initiates communication by generating an RF field. The RF field is present with short pauses for data communication. It is used for both communication as well as power supply for the NFC tag 302. In this step 306, the mobile device 12 selects the NFC tag 302 it wishes to communicate with, as is known to those skilled in the art.

[0068] The method 290 next proceeds to a SEND UID step 308 in the NFC tag 302, wherein the selected NFC tag 302 sends its UID to the mobile device 12. The mobile device 12 receives the UID in a RECEIVE UID step 310.

[0069] The method 290 next proceeds to a SEND READ_SIG COMMAND step 312 in the mobile device 12, wherein the mobile device 12 sends a read signature command, READ_SIG, to the selected NFC tag 302.

[0070] The method 290 next proceeds to a SEND SIGNATURE step 314 in the NFC tag 302, wherein the NFC tag 302 receives the READ_SIG command and responds by sending the signature stored therein to the mobile device 12. The mobile device 12 receives the signature in a RECEIVE SIGNATURE step 316. In this manner, the mobile device 12 obtains the UID from the NFC tag 302.

[0071] The method 290 next proceeds to a SEND UID & SIGNATURE step 318 in the mobile device 12, wherein the mobile device 12 establishes communication with a centralized server 14 and sends the UID and the signature received from the NFC tag 302 to the centralized server 14.

[0072] The method 290 next proceeds to a VERIFY SIGNATURE step 320 in the centralized server 14, wherein the centralized server 14 verifies the received signature. The centralized server 14 does this by signing the received UID 50 using the public key 51 stored in the centralized server 14 to generate an expected signature. The centralized server 14 compares the expected signature with the received signature 52 from the NFC tag 302. If it is determined in the VERIFY SIGNATURE step 320 that there is no match between the expected signature and received signature 52, the method 290 proceeds to a SEND ERROR MESSAGE step 322 in the centralized server 14, where the centralized server 14 alerts an administrator of the centralized server 14 to a possible case that the product may be counterfeit and sends an error message to the mobile device 12 that is received by the mobile device 12 in a RECEIVE ERROR MESSAGE step 323. If however it is determined in the VERIFY SIGNATURE step 320 that the two signatures match, the method 290 proceeds to a SEND PASSWORD step 324, wherein the centralized server 14 sends the password associated with the NFC tag 302 that is stored thereon to the mobile device 12.

[0073] The mobile device 12 receives the password in a RECEIVE PASSWORD step 326. The method 290 next proceeds to a SEND PASSWORD step 328 in the mobile device 12, wherein the mobile device 12 sends the password received from the centralized server 14 to the NFC tag 302. The NFC tag 302 receives the password in a VERIFY PASSWORD step 330, wherein the NFC tag 302 compares the received password with the password stored therein. If it is determined that the received password matches the password in the NFC tag 302, the method 290 proceeds to an UNLOCK MEMORY step 332, wherein the NFC tag 302 unlocks the

password-protected memory for reading. If however it is determined in this VERIFY PASSWORD step 330, that the received password does not match the password in the NFC tag 302, the method 290 proceeds to a KEEP MEMORY LOCKED step 334 wherein the NFC tag 302 keeps the password-protected memory in a locked state and increments a count of the number of unsuccessful password verification attempts. If a maximum number of unsuccessful password verification attempts maintained by the NFC tag 302 is reached, memory access is no longer allowed by the NFC tag 302. Any successful password verification, before the limit of unsuccessful password verification attempts is reached, resets the count of the number of unsuccessful password verification attempts to zero.

[0074] The method 290 next proceeds to a SEND READ COMMAND step 335 in the mobile device 12, wherein the mobile device sends a READ command to the NFC tag 302 to read the encrypted string that is stored in the NFC tag 302.

[0075] The method 290 next proceeds to a SEND ENCRYPTED STRING step 336 in the NFC tag 302, wherein the NFC tag 302 sends the encrypted string stored therein to the mobile device 12 if the password-protected memory has been unlocked in the UNLOCK MEMORY step 332. The NFC tag 302 will send an error message or a default string to the mobile device 12 if the password-protected memory remains locked. The mobile device 12 receives the encrypted string in a RECEIVE ENCRYPTED STRING step 338.

[0076] The method 290 next proceeds to a SEND UID & ENCRYPTED STRING step 340 in the mobile device 12, wherein the mobile device 12 sends the last received UID and encrypted string to the centralized server 14. The method 290 next proceeds to a VERIFY ENCRYPTED STRING step 342 in the centralized server 14, wherein the centralized server 14 compares the encrypted string 268 received from the mobile device 12 with the encrypted string stored thereon corresponding to the UID. If it is determined that there is a match between the two encrypted strings, the method proceeds to a CALL SMART CONTRACT step 344, wherein the centralized server 14 calls a smart contract to register ownership of the product in a ERC721 non-fungible token (not shown) in the distributed ledger network 100. In this manner, the physical product 300 can be tokenized by associating its UID with an on-chain reference of the distributed ledger network 100. Non-fungible tokens contain identifying information recorded in the smart contracts. This information makes each

non-fungible token different and as such, they cannot be directly replaced by another token. They cannot be swapped like for like, as no two tokens are alike.

[0077] The non-fungible token is capable of having the token holder's rights and legal responsibilities embedded directly onto the token, along with an immutable record of ownership. This immutable record means that nobody can "erase" the ownership even if the information is not registered in another registry or database. These characteristics add transparency by tracking and recording the history of the product every single time it changes hands. Preferably, the non-fungible token is transferred when the physical product 300 is sold. The non-fungible token will be mapped to the Ethereum address of the new owner in the smart contract as is known to those skilled in the art.

[0078] The method next proceeds to a SEND CERTIFICATE step 346 in the centralized server 14, wherein the server 14 sends a certificate of authenticity to the mobile device 12. The certificate of authenticity may be simply text or an image including information, such as but not limited to, one or more of a certificate number, a date of ownership registration and owner's information. The certificate of authenticity may be sent with or without confirmation from the distributed ledger network 100. Sending the certificate to the mobile device 12 without the need for confirmation from the distributed ledger network 100 however provides for a better user experience. The mobile device 12 receives the certificate of authenticity in a RECEIVE CERTIFICATE step 348 and displays the certificate of authenticity in the app accordingly. The method 290 ends in this RECEIVE CERTIFICATE step 348.

[0079] However, if it is determined that the two encrypted strings do not match in the VERIFY ENCRYPTED STRING step 342, the method 290 proceeds to a SEND ERROR MESSAGE step 350, wherein the centralized server 14 sends an error message to the mobile device 12. The mobile device 12 receives the error message in a RECEIVE ERROR MESSAGE step 352 and displays an appropriate message in the app to inform the user accordingly. The method 290 then ends in this RECEIVE ERROR MESSAGE step 352.

[0080] Advantageously, the ownership data management system 100 improves the user experience by providing a timely response to an ownership registration or ownership transfer request through the introduction of the centralized server 14. Since management of private and public keys are handled by the business owner,

casual collectors will find the system less cumbersome and more user friendly. At the same time, since the ownership records are also uploaded onto a distributed ledger network 16 to be in sync with those stored on the centralized server 14, the ownership records are made more secure and tamper-resistant. And together with the use of the electronic tag, the system 10 therefore gives the collectors the consumer confidence that products are original and assurance that they are dealing with true owners of the products. Furthermore, with bulk uploads to the distributed ledger network 16 that charges per transaction carried out only once in an interval, the running cost of the system 10 is reduced significantly for the business owner.

[0081] Furthermore, tokenizing the product on a distributed ledger network that cannot be easily altered provides for independent and decentralized verification. This contributes to a new level of security, especially with the way authentication of the NFC tag 302 is carried out. The distributed ledger-based security also provides increased transparency and a clear record of beneficial ownership with certainty at any point in time.

[0082] Although the present invention is described as implemented in the above described embodiments, it is not to be construed to be limited as such. For example, although it is described that a datum that is associated with ownership records created in an interval is sent to the distributed ledger network 16 at the end of the interval to be uploaded/stored thereon, a datum for each ownership record may be sent to the distributed ledger network 16 as and when the ownership record is created by the centralized server 14.

[0083] As another example, it is described that the centralized server 14 checks regularly with the distributed ledger network 16 for the availability of the block ID (transaction hash). The distributed ledger network 16 may however be configured to send the block ID (transaction hash) to the centralized server once the block ID (transaction hash) is available. In such a case, obtaining the block ID (transaction hash) by the centralized server 14 is merely receiving it from the distributed ledger network 16.

[0084] As another example, instead of sending a datum that is associated with ownership records created in an interval to the distributed ledger network 16, the centralized server 14 may send a datum associated with ownership records to the

distributed ledger network 16 when the number of yet to be uploaded ownership records in the centralized server 14 reaches a predetermined number.

[0085] As yet a further example, it is described that the centralized server 14 sends a CONFIRMATION REQUEST message 164 to the new buyer 140, such a message 164 may be sent outside of the system 10 to the new buyer 140 by the owner 72. For example, the message 164 may be sent as a short message from the owner's mobile device 12 to the new buyer's mobile device 12.

[0086] As another example, although NFC is described as a communication protocol between the mobile device and the tag, it should not be construed to be limited as such. Those skilled in the art will recognize that any suitable wired or wireless communication protocols can be used.

[0087] Also, a specific sequence of authentication is described, but again it should not be construed to be limited as such. For example, the encrypted string could have been read and verified before verification of the digital signature. The encrypted string may also be stored in a memory that is not password protected.

[0088] As yet another example, the UID and signature may be obtained simultaneously and not one after another as described.

[0089] It should be further appreciated by the person skilled in the art that one or more of the above modifications or improvements, not being mutually exclusive, may be further combined to form yet further embodiments of the present invention.