Provisional Patent Claim
April 23, 2020

Title:

Protecting to in-the-clear inventory labeling with non-collision, unique random serial numbers in order to prevent forward guessing by bad actors.

Abstract: In an anti-counterfeit system and method, a set of identification codes are generated from data associated with units from a line of items. Universal labeling standards currently require a product to carry within its item unit identification (IUID). The IUID contains a series of identifying strings that are used to ascertain the manufacturer, product type, batch of production, expiration date and a unit serial number that, when concatenated with the batch, allows the identification of one unit from another even if all label and physical attributes are similar to the rest of that batch. Current implementations of IUID's allow bad actors to forward-predict the IUID of future production runs and label their fake products with concocted IUID's. Their counterfeit product labels become indistinguishable from genuinely labeled products because they are using future valid number strings.

In this invention, a non-collision, unique random (NCUR) key replaces the current serial number string. This NCUR key prevents all forward prediction of the IUID. Furthermore, the NCUR key is generated by utilizing the other strings contained in the IUID in combination with uniquely generated but known 'genesis' codes. The known string is used to initiate the controlled start of a rolling encryption key. This key is then used to generate the series of NCUR's that are unique to that set of labels. This invention allows for the remote generation of codes without connection to a controlling server and further allows for the checking of those codes, again without an immediate connection to a controlling server, as long as both the creating and checking systems have, at some point in the past received a synchronized encrypted set of genesis codes in order to start their calculations. The genesis codes can be controlled by a centralized server or can be generated in a fully distributed set of ledgers by a smart contract. In either case, commonly understood public-private keys are then used to encrypt the confidential data. Access to the code data is therefore limited through authentication of any third party such as the creating or checking parties. They must be given the relevant keys in order to use the system – once they have been granted access to and copied the set of keys, they no longer need to be connected to the internet to ascertain that the NCUR in the label was printed by a genuine party.

Background:

Current labeling standards connect something general to something specific along a predefined encoding schema. To understand how these systems connect, let’s propose a set of four (5) alphanumeric character strings, A, B, C, D and E. Think of each character string as representing something important about the item.

Table 1:

String	Data	Example
A	Company Identifier	Cage Code
B	SKU (Store Keeper Unit)	Product or “bar” code
C	Batch Identifier	Lot number
D	Unique Identifier	Serial number
E	Expiration Date	YYMMDD

If you combine String A and String B, otherwise known as concatenation, it is now possible to distinguish between a company’s product lines. Likewise, it is also possible to distinguish it from all other, unalike items. Through concatenation of C and D, it is now possible to distinguish the item from all other like items. When strings A, B, C, and D, are combined, all items are now fully and universally discoverable and traceable but these codes are not yet secured against malicious actions and counterfeit activity. Here is why:

1. A and B are constant, which makes it easy for humans to identify parts by the serial number, and it also means anyone can easily recreate the first half of the code.
2. String C is a batch identifier and so usually follows a progressive sequence or is based on the production timestamp. Once a sequence is found, it is “forward predictable” by date or lot.
3. String D is most commonly only unique within the batch (i.e. when concatenated with String C. This occurs because;
   1. Prior to digital print heads, it was cheaper to repeat a sequence of numbers with a rotating head printer and serial number generation programs were designed around this, no longer applicable, limitation.

2. Prior to inline scanning tools, it was easier to find badly printed labels if they ran sequentially - again this is no longer the case as production lines have far fewer bad prints (near to none) and this rejection process is now automated with inline scanners identifying bad labels both during a print run and during a production run.

Practice 3a and 3b make the entire code forward predictable and non-protectable against counterfeiters and malicious actors.

If you replace the unique identifier string D with a nonsequential identifier you can defeat a counterfeiter's attempts to forward predict the code, however, you then need to distribute the identifiers, or allow permanent access to a centralized database or distributed ledger in order to check the code on the item was truly issued by the company.

However, by utilizing the in-the-clear data in string E in combination with a private but known rolling encryption key, which increments for each code generation, it is possible to both generate a non-collision, non-forward predictable serial number and to then check that number remotely without reference to a centralized database or distributed ledger of codes as long as you have previously been granted access and taken a copy of the correct initiation keys for the presented expiration date.

This allows future offline confirmation of codes making this calculation based invention a useful, scaleable and unique.

_____

Drawings:

Please see additional sheet

_____

Background of the invention

While the implementation of standards in supply chain labeling has facilitated the rapid adoption of unique labeling technology and brings benefit to supply chain visibility, it has also offered malicious actors and counterfeiters the capability of understanding the structure of a label and then forward predicting future codes even before manufacturers themselves have issued them.

This invention generally relates to this systemic problem by placing unique, non-collision codes within existing encoding structures and combining that process with well known electronic encryption

process’ to allow the remote verification of any physical item’s unique identifier while concurrently preventing a bad actor from forward predicting future codes in order to generate their own labels for counterfeit product.

Summary of the Disclosure

Certain embodiments of the present invention allow for the replacement of the unique Identifier string commonly known as the serial number product identifier within industry-standard labeling schemas with a derived number that is a non-collision, random and unique based identifier. This new serial number can be calculated remotely by an informed, but only intermittently connected device. The device receives standard labelling data electronically, visually, or by the physical entry of the data. It then extracts and unbundles that item’s strings, adds a secret key that is chosen based on the product’s expiration date and combines the secret key with a short incrementable second string. This combined string is hashed, using a known hash such as SHA 256, to generate the first serial number for this batch of product. The short string is then incremented and the hashing process is repeated and in order to generate enough codes to label all product with that expiration date. These are then used as the serial numbers on any printed label or for electronic identification tags such as RFID or NFC.

In addition a second short incremental value can also then be added as the trailing set of digits of the hashed serial number to provide a quick reference for humans reading the code, allowing activities, such as manual quality control within print runs, to be conducted without change to current practices. This trailing code can be numeric or alphanumeric and alphabetical only and can be of varying length.

This process of acceptable code generation can be replicated by a checking party, who’s own device also has previously been granted access to and has received the secret key for that expiration date. This allows confirmation of a good manufacturing code without the need for constant calls to a remote server and or without needing to maintain a list of all issued serial numbers on the scanning device or system.

Brief description of the drawings

Figures. 1A, 1B, and 1C show a flow chart of the production side of one embodiment of an anti-counterfeiting system in accordance with the invention.

Figures 2A and 2B are a flow chart showing the checking side of one embodiment of an anti-counterfeiting system in accordance with the invention.
Like reference numbers and designations in the various drawings indicate like elements.

A detailed description of the invention

Figs. 1A, 1B and 1C show a flow chart the production side of one embodiment of an anti-counterfeiting system in accordance with the invention. For sake of brevity it is assumed that all communications are properly encrypted and protected from third party interception using standard encryption technologies.

For each item to be marked with a non collision unique random alphanumeric Serial number we first generate a set of primary keys (Step 102) each of which is associated with its own possible future expiration date (Step 104). This set of primary keys is encrypted using well known public/private key or similar encryption techniques (Step 106) by a computer system or piece of executable code based on a distributed set of ledgers (Step 108).

The encrypted primary expiration date keys are downloaded or accessed from a distributed ledger and decrypted (Step 110) within an executable file that does which maintains their secrecy.

Based on the required expiration date the executable selects the appropriate primary key (Step 112) and concatenates that key with the production run information, such as the manufacturer’s identity, the product line and the batch, and any other appropriate metadata that will be included in the final label as well as an initial variable string that increments as each new serial number is generated (Step 114). This string set is then generated (Step 116) by applying a suitable hash function. A hash function is any algorithm or subroutine that maps large data sets of variable length to smaller data sets of a fixed length.

The executable then repeats (Step 122) the process having made an increment to the variable string (Step 118) until there are enough codes for the days production (Step 120) if the delivered codes need to be easily read in order of production by a human (Step 124) a second incremental alphabet or numeric based string of an appropriate length can be added as a trailing set (Step 126). Thes new non-collision, unique random serial number can then be embedded in place of a standard item serial number within industry standard codes, such as Company ID, Product ID, non-collision, unique random

serial number, batch or lot number, expiration date (Step 128). The set of complete codes is then released for use as a physical or electronic label.

Figs. 2A, 2B and 2C show a flow chart for the checking side of one embodiment of an anti-counterfeiting system in accordance with the invention.

A physical or electronic label is scanned by an authorized device or the data is entered into an authorized device by an authorised use (Step 200). The data is analyzed and the expiration date is extracted (Step 202). If there is no expiration date the code is rejected (step 204). If the expiration date is valid the associated primary key is found from a previously downloaded set of master keys (Step 206). The primary key and the production run information, such as the manufacturer's identity, the product line and the batch, and any other appropriate metadata as ascertained from the physical or electronic label and each of the rolling incremental strings are then used to creat a full set of possible good codes (Step 208). The presented non-collision, unique random serial number from the physical or electronic label is then compared to the set of good codes generated by the local checking device (Step 210). If the presented code is not present then it is rejected. If the code is present then it is accepted by the checking device. In both cases the device can be set to choose to inform the user (Step 214) or to leave them ignorant of the true state of the presented item.

Once an item has been analyzed the outcome can then be sent to the original issuer (Step 218) of the code or an external authority or any third party if desired. If there is no current internet connection (Step 220) the notification can be queued for later dispatch (Step 224) allowing the issuer or other interested party to be informed that the label has been verified (step 226).

Claims

What is claimed for:

1. An anticounterfeiting method, that allows creating parties to calculate their own non-collision unique random serial numbers in

order to prevent forward guessing of those codes by bad actors who wish to make counterfeit labels.

1. By generating a series of series of non-collision, unique random alphanumeric strings on a computer system, server
2. or by executing a piece of code, commonly known as a smart contract which , on a set of distributed ledgers generates a series of series of non-collision, unique random alphanumeric strings , with one code to represent each future expiration date projecting forward from todays date
3. protecting that series of non-collision, unique random alphanumeric strings strings by encryption and the issuance of private public keys that then only allow access by holders of those keys,
4. Then utilizing the expiration date of the item being labelled to pick the relevant non-collision, unique random alphanumeric string
5. Combining the string with the batch, the batch and expiration date, the batch, expiration date and the product identifier or the batch, expiration date, product identifier, and the company identifier,
6. Further combining it with a numerical or alphanumeric string that can be incremented,
7. hashing the concatenated code using a standard hashing function (such as SHA 256 or similar) to create the first non collision unique random key for that batch or expiration expiration print run of labels,
8. If required, adding a further, human readable trailing string of letters starting with AAAA, then AAAB to allow manufacturing order and immediate visual identification of product for QA purposes
9. Recombining the hash value and trailing string with company identifier product identifier, batch, expiration date, to create the first items new, non forward predictable unique identifying string.
10. Then incrementing the numerical alphanumeric or numerical string within the concatenated code and Repeating steps process ‘c’ through ‘i’ until enough codes for the production run are available for delivery to the print heads.

2. And furthermore allows checking parties through utilization of a designated app on a smartphone or a service within a warehouse management scanning system to

1. Download to a computer or server, subscribe to or otherwise record, act as a simple node within a series of distributed ledgers and receive a copy of or act as a full node within a seriesof distributed ledgers, and receive a copy of the encrypted set of expiry date dependant genesis codes
2. By use of a key shared with them electronically or via other communication medium, unlock the encrypted genesis codes
3. Visually scan, manually type, tap the NFC, receive the RFID or somehow otherwise take the contents of the label and allow the designated app on a smartphone or a service within a warehouse management scanning system to utilize the passed expiration date to select the correct key from previously ascertained master list
4. the designated app on a smartphone or a service within a warehouse management scanning system will then re-generate the set of acceptable non collision unique random keys for that expiration

date using the same process as described in claims 1a through 1j.

5. the designated app on a smartphone or a service within a warehouse management scanning system can then compare the non collision unique random key extracted from the presented data to its generated set of non collision unique random keys and if it matches one of the keys that label was originally generated by the authorized creator.

3. The designated app on a smartphone or a service within a warehouse management scanning system can then

1. Be set to notify its user or users that the product is good or can be set to tell the user or users that all products are good even if the check has failed, or not tell the users anything about the checking process

2. Be set to send a notification to the creator that the product has been checked immediately or when or if an internet connection becomes possible

3. or not communicate that the label has been checked with the creator.

4. Publish to a private or public record that the item has been checked

5. signify a change of state such as but not limited to purchased, shipped or used to a public or private record or the creator.