Authentication Pedigree Scheme for Supply Chain

Manki Min
Computer Science Program
Louisiana Tech University
Ruston, Louisiana 71272
mankimin@latech.edu

Sunho Lim
Dept. of CS
Texas Tech University
Lubbock, TX 79409
sunho.lim@ttu.edu

Yi Liu
Department of EECS
South Dakota State University
Brookings, South Dakota 57007
yi.liu@sdstate.edu

Hyeun Joong Yoon
Department of EECS
South Dakota State University
Brookings, South Dakota 57007
hyeunjoong.yoon@sdstate.edu

***Abstract*—In this paper, we present a novel idea of multilayer aggregate authentication scheme that can be used in any distribution phase of supply chain for more accurate tracking of packages and for innovative authentication of the hierarchy of packages. With minimal modification cost, our scheme can be adapted to any existing cryptographic hash chain based authentication schemes such as RFID-based scheme. By allowing the aggregate authentication of multiple layers (for example a bigger package and its smaller inner packages) at once, our scheme provides more reliable and accurate way to track the packages and effectively thwarts the counterfeiters at every phase of the supply chain. Our scheme also enables any entity in the entire supply chain including the producers, transporters, distributors, sellers, and even consumers to easily authenticate the authenticity of a package/product along its packaging hierarchy.**

***Index Terms*—hash chain, multilayer aggregate authentication, supply chain, pedigree of authenticity**

I. Introduction

The International Chamber of Commerce (ICC) initiative Business Action to Stop Counterfeiting and Piracy (BASCAP), teamed up with The International Trademark Association (INTA), prepared the report on the global economic effect of counterfeiting and piracy in 2016. In their report [1], the estimated value of international and domestic trade of counterfeit/pirated products in 2013 was \$0.9-\$1.1 Trillion and the projected total value of counterfeit/pirated goods in 2020 will be more than doubled (\$1.9-\$2.8 Trillion). Hence, it is unquestionable that anti-counterfeiting has enormous effect onto the global economy.

In order to prevent counterfeiting, many techniques such as Radio Frequency Identification (RFID) [2]–[6] have been used mostly on the packages. There are multiple hierarchies/types of packaging as defined in [7]. Primary packaging (also known as consumer packaging or sales packaging) is the one in contact with the products such as wrapping. Secondary packaging is the one containing multiple primary packages. Tertiary packaging is the one containing multiple primary or secondary packages assembled on a pallet or roll container. Based on the emphasis, there are transport/industrial/distribution packaging, display packaging, and retail packaging of which design concern is emphasized on transport, display, and retail sales, respectively.

In this paper, we present a novel scheme that can authenticate the multilayer descendancy/ancestry of multiple hierarchical packages at once, for example secondary-primary packages, or even tertiary-secondary-primary packages. By doing so, we can better protect the authenticity of the products as well as packages at any phase in the supply chain and hence provide stronger protection against any type of counterfeiting. Our main objective in this work is to design the multilayer descendancy/ancestry authentication scheme that has low computation/communication complexity and also has minimal modification that needs to be applied to existing authentication schemes that are based on cryptographic hash chains.

Mostly the research on supply chain authentication schemes is essentially individual authentication. Either it is simple individual authentication or it is multiple individual authentications for carefully chosen smaller groups. The trade-off between the security and the efficiency of the group authentication (involving individual authentications) has been the obstacle for the design of better group authentication. Our main contribution in this paper is twofold: 1) we proposed the novel idea of multilayer aggregate authentication and 2) we presented two modes of multilayer aggregate authentication. By separating individual authentication from group authentication, our scheme can provide an efficient and effective means to achieve group authentication of multiple levels of package layers.

The remainder of this paper consists as follows: brief overview of the existing authentication schemes for supply chain is discussed in Section II. Our novel scheme of multilayer-aggregate authentication is described in Section III, and its security is analyzed in Section IV. Finally, Section V concludes this paper.

II. Related Work

Different techniques have been studied and used for authenticating different types of products. For example, for the electronic components such as the Integrated Chips (ICs), both chip ID and package ID are protected [8]. For chip IDs, Physically Unclonable Functions (PUFs), hardware metering, secure split test, combating die / IC recovery (CDIR), antifuse-based technology for recording usage time, and Electronic Chip ID (ECID) can be used. DNA markings, nanorods, and magnetic PUFs are used for package IDs.

Pharmaceutical market is one of the major victims of counterfeits and RFID has been actively used in its supply chain. As analyzed in [9], the pharmaceutical supply chain

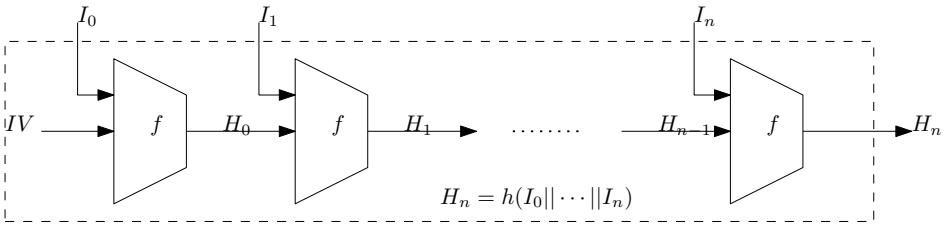

Fig. 1. Cryptographic Hash Structure

requires the following security aspects: anti-adulteration, auto-ID inventory, traceability, accountability, privacy protection, and compliance detection. To achieve the aforementioned security aspects, RFID can be used to generate anti-forgery tags that can be used to authenticate the pharmaceutical products in order to prevent counterfeiting.

With the advance of Internet of Things (IoT) technology, RFID plays an even more important role in securing the whole supply chain. As stated in [10], IoT combined with RFID affects the whole supply chain by optimizing management and resource utilization and improving the transparency in real-time. Many links in the supply chain such as manufacturing links, warehousing links, transportation links, and selling links are affected along the way.

For the authentication of the entire package with multiple smaller packages inside, a multi-batch scheme can be used as in [11]. In the paper, the authors presented a centralized RFID-based scheme that can operate multi-batch authentication. While the security level is highly maintained, the volume of the scanning and the related computation remain large. Another large-scale RFID system authentication scheme based on framed slotted Aloha algorithm is presented in [12], which instead of scanning each tag, scans a partial set of tags determined by a precomputed tree.

Grouping proof methods can be used to check the integrity of the cargo loads in order to thwart the cargo theft during transport as in [13]. The proposed scheme in the paper is using RFID for each tag and construct a tree structure to minimize the number of tags to be scanned in order to verify the integrity of the entire cargo loads. While this approach can reduce the total number of scanning, due to its probabilistic nature, the integrity is not always guaranteed even if the test passes. Moreover, scanning the entire cargo loads does not seem impractical if we check the integrity only when the package of the cargo loads is unpacked at a certain distribution center as long as the package authenticity is strongly maintained.

In [14], a public-key encryption scheme was used to replace the use of the cryptographic hash functions in RFID tags that contain the Electronic Product Code (EPC). While their results show reasonable performance in terms of power consumption and necessary gate counts, still the majority of the RFID-based authentication schemes is using cryptographic hash functions due to their advantageous computational cost.

III. Proposed Scheme

In this section, we first discuss about the cryptographic hash functions' properties and the hash chains' properties. Then we describe our proposed authentication scheme based on the discussed properties.

A. Cryptographic Hash Structure

Cryptographic hash functions (will be referred to as hash functions hereafter) are the algorithms that convert a variable length string into a fixed length string with the following two important properties: one-way property and collision-resistant property. One-way property means that it is computationally difficult, i.e. it takes too much time, to discover the original input string from the output of the algorithm. This one-wayness makes one of the most important characteristics of hash functions because it helps to hide information (the input string). Since the algorithm converts an arbitrary length string into a fixed length string, there are inevitably more than one input strings that map to the same output string which is called collision. Even though collision itself is inevitable, it must be computationally difficult to find such collisions in order to be considered as a hash function.

Hash functions are widely used especially in the field of authentication which does not require confidentiality, i.e. the messages transmitted for authentication do not need to be encrypted, because 1) it has a comparatively lower computational complexity and 2) the two aforementioned properties provide computationally secure separation of the input from the hashed output.

The general structure of hash functions is depicted in Figure 1. Since a hash function must be able to take a variable length input string, any string longer than a predefined block size (may vary depending on the algorithm/version) must be divided into multiple blocks as in the figure. After the proper block preparation, each block (I_0 through I_n in the figure) will be processed by the compressor function f together with another input (IV as the initial value and the output H_0 through H_n of the previous block after that). The final output H_n is called the hash code for the input string $I_0||I_1||\cdots||I_n$ where $||$ denotes the concatenation of strings.

This structure provides an interesting property with respect to appending additional input string when generating the hash code. Let's assume that we have a valid hash code H obtained from the input string $I = I_0||\cdots||I_n$ and a certain initial value IV which is unknown. Then we can freely generate

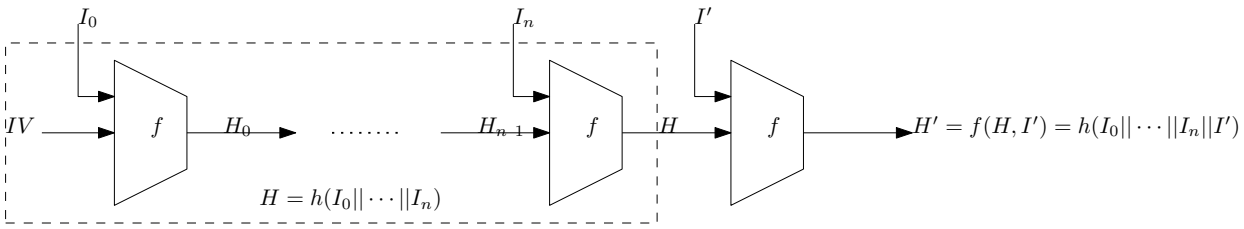

Fig. 2. Generating Hash Code with Additional Input Appended

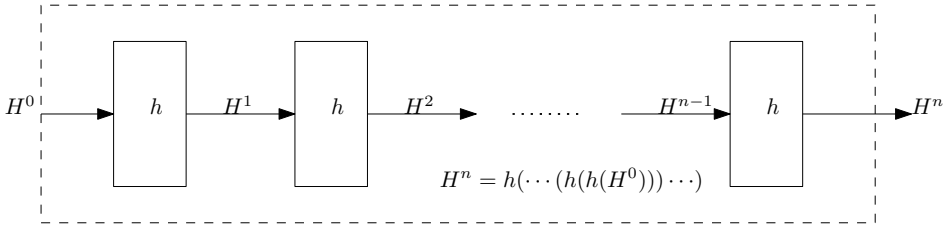

Fig. 3. Hash Chain Structure

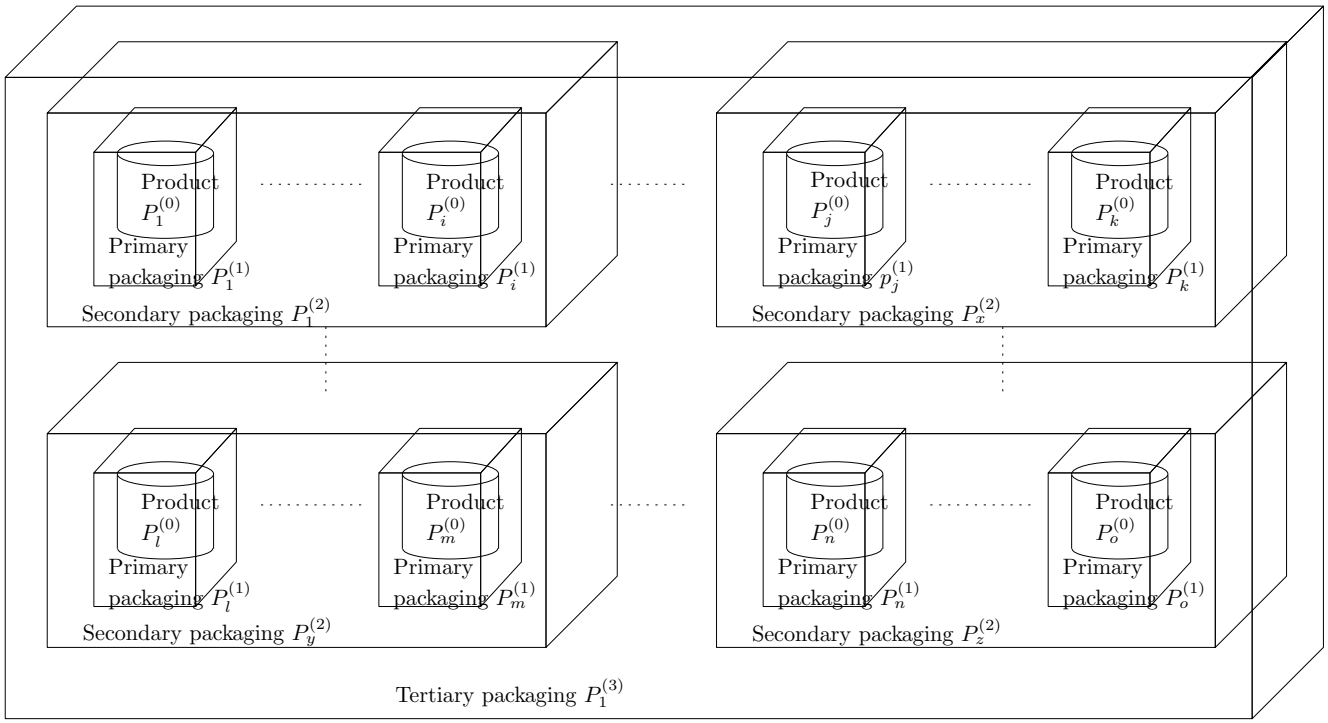

Fig. 4. Multilayered packaging

another valid hash code H' for the input string $I||I'$ for an arbitrary string block I' without knowing IV. The way to construct H' is to use H as the new initial value and to run the hash algorithm on the new input string I' as in Figure 2. Similarly, we can append a variable length additional string to the original input string I and generate its valid hash code. This is the fundamental property of the hash functions that our proposed scheme is based on. The important point in this process is that even without the possession of secret IV, valid hash codes for additional string appended to the original input string can be generated. In many applications such as Hash-based Message Authentication Code (HMAC), a key string is used as IV and this key must remain secret during its lifetime.

B. Hash Chains

The most widely used hash-based authentication method utilizing one-way property and low computational complexity of hash functions is to make a hash chain which is simply a sequence of hash codes where the input of the current hash code is the previous hash code as in Figure 3. In the figure, H^0 represents the initial code and $H^1, \cdots, H^n$ represent the

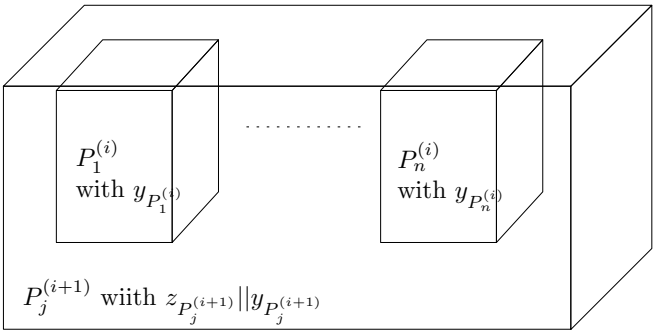

Fig. 5. Assumption for Multilayer Aggregate Authentication

hash codes in the chain, generated as $H^i = h(H^{i-1})$. We call H^n a terminal hash chain code and assume that H^n is public in the sense that it is not kept secret.

Using hash chains, authentication can be performed in the following way: Only one hash code, for example either the hash code H_i that was used last time or the last hash code H^n in the chain, is made public. At every authentication attempt, the hash code in the chain that is the input of the one used last time is given, i.e. H^{i-1} is given if H^i was used last time. The authentication requires either one or $n - i + 1$ times of application of the hash function to H^{i-1} to get H^i or H^n depending on the public hash code and if the computed hash code is identical to the public hash code, the authentication is completed.

Now we will discuss how hash chains can keep security using Figure 3. The one-way property of the hash functions makes the initial code H^0 remain secret, and hence it is computationally difficult to generate the entirely/partially same hash chain using only the public information H^n. If the initial codes are maintained by some trusted third party, the authenticity of the entire hash chain can be verified. The low computational complexity of the hash functions makes it practical to apply the hash function multiple times for the purpose of the authentication. In our previous work [15], we have used the hash chains to print the last hash code as the public hash code in the form of QR codes for the individual product authentication. If used with active RFID devices, the lastly used hash code can be made public to further reduce the computational cost for the authentication.

C. Multilayer Aggregate Authentication Scheme

Before describing the scheme, we will make assumptions for the packaging in supply chain. Figure 4 depicts our assumption on the multilayered packaging of products as summarized in [7]. Each product is in the primary packaging which may use different authentication method such as RFID, QR code, etc. Multiple products (in its primary packaging) are inside of the secondary packaging which typically uses an RFID-based authentication method. Multiple secondary packages are grouped into an even bigger packaging, tertiary packaging, for better distribution and transportation. Depending on the nature of the product and distribution/transportation requirements, more layer of packaging is possible. Our scheme is based on the assumption that each package can be individually authenticated using hash-chain structures and only focuses on the multilayer aggregate authentication.

Throughout this paper, we will use the following notations to describe and discuss our scheme.

- $\oplus$ is a simple XOR operator.
- $:=$ is a substitution operator.
- $P^{(i)}$ represents a package at the i-th level layer (or simply i-th layer). $P^{(1)}$ represents the primary packaging. We denote the j-th package at the i-th layer by $P_j^{(i)}$.
- $P^{(i)} \in P^{(j)}$ means that the package $P^{(i)}$ is inside of the package $P^{(j)}$ possibly through multiple level of packaging. Note that $j > i$.
- $LP_{P^{(i)}}^{(j)} = \{$ every j-th layer package $p \in P^{(i)}$ $\}$.
- $y_{P^{(i)}}$ is the terminal hash chain code of the package $P^{(i)}$.
- $LY_{P^{(i)}}^{(j)} = \oplus_{p \in LP_{P^{(i)}}^{(j)}} y_p$ is the aggregate code of y hash codes of all j-level lower layer packages $\in P^{(i)}$.
- $z_{P^{(i)}}$ is the aggregate hash code for all multilevel lower layer packages of the package $P^{(i)}$ and is defined recursively as follows:
  $z_{P^{(1)}} =$ NULL, or empty
  $z_{P^{(i)}} = h\big(\big[\oplus_{p \in LP_{P^{(i)}}^{(i-1)}} z_p\big] || LY_{P^{(i)}}^{(i-1)}\big)$, where
  h is a hash function which is based on the compressor chain structure as in Figure 1.
- $LZ_{P^{(i)}}^{(j)} = \oplus_{p \in LP_{P^{(i)}}^{(j)}} z_p$ is the aggregate code of z hash codes of all j-level lower layer packages $\in P^{(i)}$.
- $c_{P^{(i)}}$ is the authentication code verifying that $P^{(i)}$ and all its lower layer packages are authentic.

We assume that each package $p^{(i)}$ keeps $z_{P^{(i)}}||y_{P^{(i)}}$. Here $z_{P^{(i)}}$ recursively contains the z hash codes of all lower layer packages $\in P^{(i)}$ as well as y hash codes of the inside packages at the $i-1$ layer. Hence $z_{P^{(k)}}$ contains every package's y hash code or z hash code in the entire packaging hierarchy. By xor-ing z hash codes instead of y hash code concatenations of multilevel lower layer packages, we can reduce the complexity of this multilayer aggregate authentication scheme while still keeping every y hash codes effective in the authentication procedure.

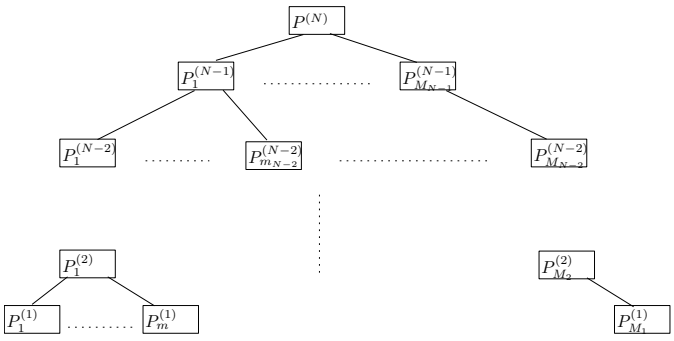

Fig. 6. Tree-like Structure of Packaging

Now we will discuss about the structure of our multilayer

aggregate authentication codes z and y. The hierarchy of the multilayer packaging can be viewed as a tree-like structure as in Figure 6. As in the figure, any package is contained in only one outer (or upper-layer) package. For any package $P^{(i)}$, $z_{P(i)}$ can be considered to include y_p for all $p \in LP^{(1)}_{P^{(i)}} \cup \cdots \cup LP^{(i-1)}_{P^{(i)}}$ either directly or indirectly through a sequence of hash application. Hence $z_{P^{(i)}}$ cannot be modified individually without changing the hierarchical y codes of all lower layer packages $\in P^{(i)}$.

We assume that during the manufacturing process, z and y hash codes are generated in bottom-up fashion in the hierarchy structure and encoded into the corresponding packages. Level-by-level, this procedure goes as follows: y hash codes of all packages at level 1 inside the same outer package are xored and hashed into the z hash code of their outer package at level 2. Then the concatenations of z hash code and y hash code of all packages at level 2 inside the same outer package are xored and hashed into the z hash code of their outer package package at level 3. Generalizing this procedure, we see that the concatenations of z hash code and y hash code of all packages at level i inside the same outer package are xored and hashed into z hash code of their outer package at level $i+1$. This procedure will generate the z code in a different way from the definition, but they are identical as in the following equation:

$$\begin{aligned} z_{P^{(i)}} &= h\big(\big[\oplus_{p\in LP^{(i-1)}_{P^{(i)}}} z_p\big]||LY^{(i-1)}_{P^{(i)}}\big) \\ &= h\big(LZ^{(i-1)}_{P^{(i)}}||\big[\oplus_{p\in LP^{(i-1)}_{P^{(i)}}} y_p\big]\big) \\ &= h\big(\oplus_{p\in LP^{(i-1)}_{P^{(i)}}} \big[z_p||y_p\big]\big)\ . \end{aligned}$$

After z hash code is generated, each package p has the concatenation of the two hash codes encoded: $z_p||y_p$.

Now when a package $P^{(i)}$ is opened, the aggregate authentication hash code $c_{P^{(i)}}$ using the concatenation of z hash code and y hash code of all the packages inside $P^{(i)}$ at level $i-1$ as follows:

$$\begin{aligned} c_{P^{(i)}} &= f(z_{P^{(i)}}, y_{P^{(i)}}) && (1)\\ &= h\Big(\Big[\oplus_{p\in LP^{(i-1)}_{P^{(i)}}} \big[z_p||y_p\big]\Big]||y_{P^{(i)}}\Big) && (2)\\ &= f\Big(h\big(LZ^{(i-1)}_{P^{(i)}}\big), LY^{(i-1)}_{P^{(i)}}||y_{P^{(i)}}\Big)\ . && (3) \end{aligned}$$

As soon as $P^{(i)}$ is opened, $z_{P^{(i)}}||y_{P^{(i)}}$ is read, so we can calculate $c_{P^{(i)}}$ as in equation (1). As soon as scanning of $z_{P^{(i-1)}}||y_{P^{(i-1)}}$ for every inside package $P^{(i-1)}$ is finished, we can calculate $c_{P^{(i)}}$ as in equation (2). If the two c codes coincide, then we can say that the package $P^{(i)}$ contains all the packages $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$ which is denoted as $P^{(i)}$ is **multilayer descendancy authenticated**. If $z_{P^{(i)}}$ or $z_{P^{(i-1)}}$'s are invalid, then the two c codes will be different. Note that we assumed that $y_{P^{(i-1)}}$ is public and can be verified, so the attacker can only change or make up z codes. Even if invalid $z_{P^{(i-1)}}$'s are made up so that $c_{P^{(i)}}$ is verified, due to its hierarchical zipping propertied, $z_{P^{(i-1)}}$ won't work properly with the lower layer packages.

We will discuss more in detail about the z hash codes for the purpose of ancestry authentication after the upper layer boxes are already gone. For this discussion, we assume that the packages $P^{(1)}, \cdots, P^{(k)}$ are in the hierarchy such that $P^{(1)}$ is contained in $P^{(2)}$ and so on and $P^{(k)}$ is the top layer package. The Equation (3) above tells us that $c_{P^{(i)}}$ in another way. If $P^{(i)}$ keeps $h\big(LZ^{(l-1)}_{P^{(l)}}\big)||LY^{(l-1)}_{P^{(l)}}||y_{P^{(l)}}$ for all $i < l \le k$, then we can use them to reconstruct all of $c_{P^{(i+1)}}, \cdots, c_{P^{(k)}}$. When we can verify that the sequence of $c_{P^{(i)}}||\cdots||c_{P^{(k)}}$ corresponds to an already authenticated sequence of packaging hierarchy, we say $P^{(i)}$ is **multilayer ancestry authenticated**. Since each package contains every upper layer packages ancestry codes, using them we can generate $c_{P^{(i)}}||\cdots||c_{P^{(k)}}$ and the validity of this generated codes can be verified through an authentication server.

Now we are ready to discuss two modes of our multilayer aggregate authentication scheme.

1) Multilayer Descendancy Authentication: We can authenticate, upon opening of a package $P^{(i)}$, the multilayer packaging of $P^{(i)}$ as well as all its lower layer packages that are inside of $P^{(i)}$ possibly through multiple layer of packaging. As soon as the authentication is verified, additional codes will be stored into the lower layer packages P^{i-1} that are inside of $P^{(i)}$ which will be used in the next mode of multilayer aggregate authentication. As in Figure 7, the package $P^{(i)}$ contains lower layer packages $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$ and is contained in the upper layer packages $P^{(i+1)}, \cdots, P^{(k)}$. As we discussed above, each package $P^{(i)}$ keeps the z, y codes $(z_{P^{(i)}}||y_{P^{(i)}})$ as well as upper layer ancestry codes:

$$h\big(LZ^{(l-1)}_{P^{(l)}}\big)||LY^{(l-1)}_{P^{(l)}}||y_{P^{(l)}} \text{ for all } i+1 \le l \le k.$$

The reader gets

$$z_{P^{(i)}}||y_{P^{(i)}}$$

from the package $P^{(i)}$ to extract $z_{P^{(i)}}$ and $y_{P^{(i)}}$ and gets

$$z_{P^{(i-1)}_1}||y_{P^{(i-1)}_1}, \cdots, z_{P^{(i-1)}_n}||y_{P^{(i-1)}_n}$$

from all lower layer packages $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$. Now if

$$f(z_{P^{(i)}}, y_{P^{(i)}}) = h\Big(\oplus_{p\in LP^{(i-1)}_{P^{(i)}}} \big[z_p||y_p\big]||y_{P^{(i)}}\Big),$$

then the packages $P^{(i)}$ and its lower layer packages $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$ are descendancy authenticated. When authenticated, a new record of the form

$$c_{P^{(i)}}||c_{P^{(i+1)}}||\cdots||c_{P^{(k)}}$$

is stored into the authentication server and $P^{(i)}$'s all ancestry codes

$$h\big(LZ^{(l-1)}_{P^{(l)}}\big)||LY^{(l-1)}_{P^{(l)}}||y_{P^{(l)}} \text{ for all } i \le l \le k$$

are stored into all lower layer packages $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$. The computational cost for this authentication is as follows: one storing of $O(k)$ codes into each package $P^{(i-1)}_1, \cdots, P^{(i-1)}_n$, three hash computations and $O(k)$ concatenations at the reader, one lookup and storing of $O(k)$-concatenated code, where k is the total number of level of the packaging hierarchy. The protocol for this authentication is given in Figure 8.

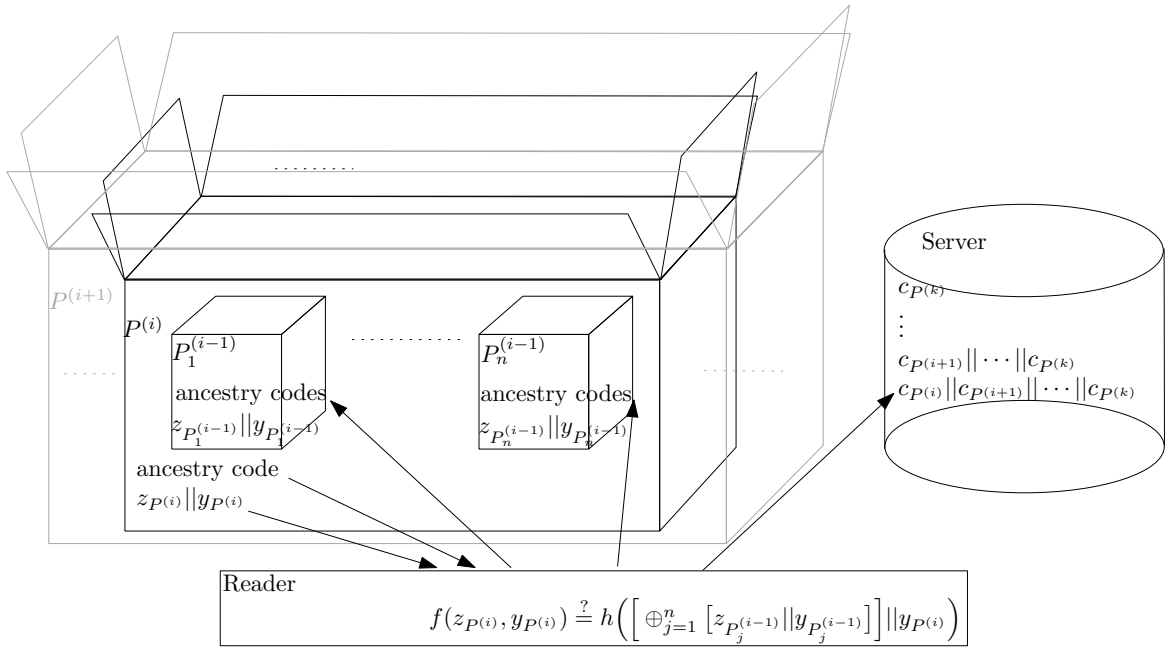

Fig. 7. Multilayer Descendancy Authentication

Server		**Reader**		**Packages** $P^{(i)}$ / $P_1^{(i-1)}, \cdots, P_n^{(i-1)}$
		Scan	$\Rightarrow$	
			$\Leftarrow$	$z_{P^{(i)}}\|\|y_{P^{(i)}}$ from $P^{(i)}$
		$c' := f(z_{P^{(i)}}, y_{P^{(i)}})$	$\Leftarrow$	$w_j := z_{P_j^{(i-1)}}\|\|y_{P_j^{(i-1)}}$ from $P_j^{(i-1)}$ for $1 \le j \le n$
		$c'' := h\Big(\Big[\oplus_{j=1}^{n}\ w_j\Big]\|\|y_{P^{(i)}}\Big)$		
		$if\ c' = c''$		
	$\Leftarrow$	$c_{P^{(i)}}\|\|c_{P^{(i+1)}}\|\|\cdots\|\|c_{P^{(k)}}$		
look up $c_{P^{(i)}}\|\|c_{P^{(i+1)}}\|\|\cdots\|\|c_{P^{(k)}}$				
if not found,				
auth message	$\Rightarrow$			
store $c_{P^{(i)}}\|\|c_{P^{(i+1)}}\|\|\cdots\|\|c_{P^{(k)}}$		if auth message is received		
		// Packages are authenticated		
$else$		$\cup_{l=i}^{k}\{h(LZ_{P^{(l)}}^{(l-1)})\|\|LY_{P^{(l)}}^{(l-1)}\|\|y_{P^{(l)}}\} \Rightarrow$		
error message	$\Rightarrow$	display auth message		store $\cup_{l=i}^{k}\{h(LZ_{P^{(l)}}^{(l-1)})\|\|LY_{P^{(l)}}^{(l-1)}\|\|y_{P^{(l)}}\}$ into $P_j^{(i-1)}$ for $1 \le j \le n$
		$else\ if$ error message is received		
		// Packages are already authenticated		
		display error message		
		$else$		
		// Packages are not authenticated		
		display error message		

Fig. 8. Multilayer Desendancy Authentication Protocol

2) Multilayer Ancestry Authentication: We can authenticate a package $P^{(i)}$'s ancestry after its upper layer packages are all gone. The previous authentication procedure will store the ancestry codes into each lower layer package and hence any package, after opening of its upper layer package, will contain all ancestry codes of its all upper layer packages. As in Figure 9, the reader gets

$$z_{P^{(i)}}||y_{P^{(i)}}$$

and

$$\cup_{l=i}^{k}\{h(LZ_{P^{(l)}}^{(l-1)})||LY_{P^{(l)}}^{(l-1)}||y_{P^{(l)}}\}$$

from the package $P^{(i)}$. Using the codes obtained from the package $P^{(i)}$, the reader computes and sends

$$a := c_{P^{(i)}}||\cdots||c_{P^{(k)}}$$

to the authentication server. If a record of a is found in the server, it means that the multilayer packaging sequence of

$$P^{(i)} \in P^{(i+1)} \in \cdots \in P^{(k)}$$

has been authenticated earlier, so we can ensure that it represents a valid ancestry of $P^{(i)}$. The computational cost for this authentication is as follows: $O(k)$ hash computations

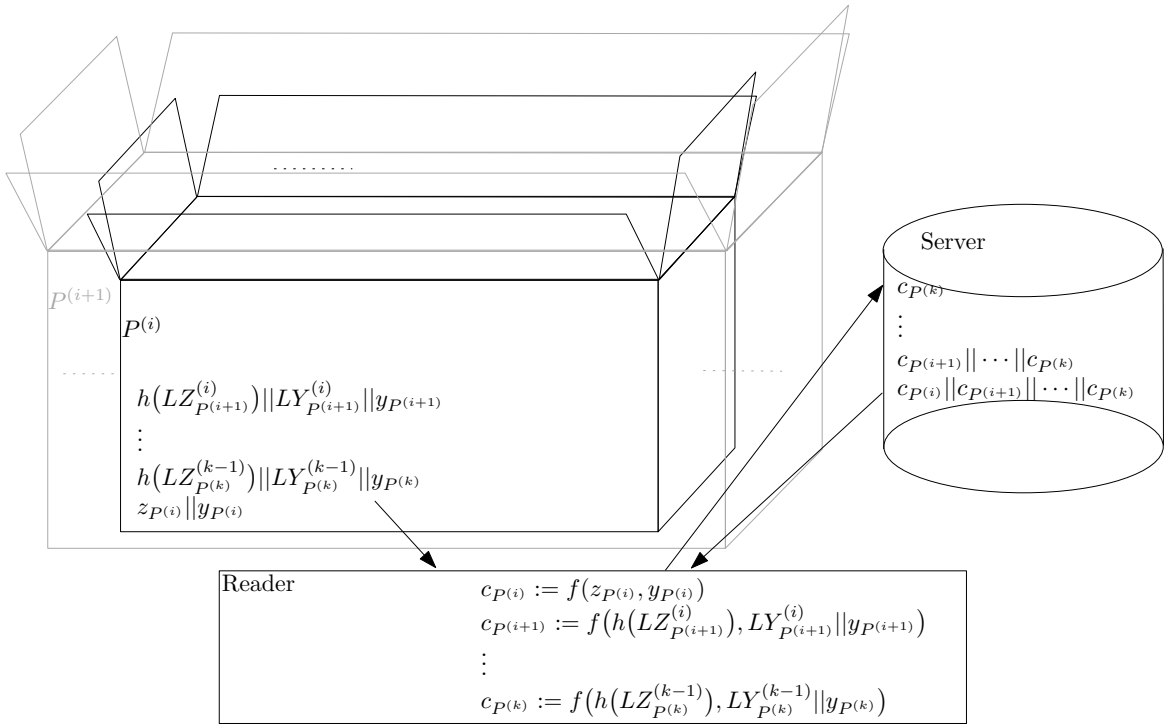

Fig. 9. Multilayer Ancestry Authentication

Server		**Reader**		**Package** $P^{(i)}$
		Scan	$\Rightarrow$	
			$\Leftarrow$	$z_{P^{(i)}}\|\|y_{P^{(i)}}$
		$c'_{P^{(i)}} := f(z_{P^{(i)}}), y_{P^{(i)}})$	$\Leftarrow$	$\cup_{l=i}^{k}\big\{h\big(LZ^{(l-1)}_{P^{(l)}}\big)\|\|LY^{(l-1)}_{P^{(l)}}\|\|y_{P^{(l)}}\big\}$
		$c'_{P^{(l)}} := f\big(h\big(LZ^{(l-1)}_{P^{(l)}}\big), LY^{(l-1)}_{P^{(l)}}\|\|y_{P^{(l)}}\big)$ for $i < l \leq k$		
	$\Leftarrow$	$c'_{P^{(i)}}\|\|c'_{P^{(i+1)}}\|\|\cdots\|\|c'_{P^{(k)}}$		
look up $c'_{P^{(i)}}\|\|c'_{P^{(i+1)}}\|\|\cdots\|\|c'_{P^{(k)}}$				
if found				
auth message	$\Rightarrow$			
		if auth message is received		
		// Ancestry of $P^{(i)}$ is authenticated		
		display auth message		
		$else$		
		// Ancestry of $P^{(i)}$ is not authenticated		
		display error message		

Fig. 10. Multilayer Ancestry Authentication Protocol

and $O(k)$ concatenations at the reader, one lookup of $O(k)$-concatenated code, where k is the total number of level of the packaging hierarchy. The protocol for this authentication is given in Figure 10.

IV. Security Analysis

Our scheme is designed to add additional security on top of the existing security offered by the underlying individual authentication scheme. We assume the same level of security that the underlying mechanism provides for individual authentication, for example secure communication between the tags and the reader in case of RFID-based authentication and secure communication between the reader and the server. In this paper, we don't discuss about the security of the underlying individual authentication and only focus on the following multilayer aggregate authentication related security features only.

1) Authenticity of Entire Packaging: Our scheme is designed to authenticate the entire packaging at any point in the supply chain. Since the z code of $P^{(i)}$ is a hash code in the form of

$$z_{P^{(i)}} = h\big(\oplus_{p \in LP^{(i-1)}_{P^{(i)}}} \big[z_p||y_p\big]\big),$$

revelation of $z_{P^{(i)}}$ does not reveal

$$z_{P_1^{(i-1)}}||y_{P_1^{(i-1)}}, \cdots, z_{P_n^{(i-1)}}||y_{P_n^{(i-1)}}.$$

Hence the counterfeits of $P^{(i-1)}$ cannot contain correct z codes even if $P^{(i)}$ is opened and its lower layer packages

are counterfeited. Or in some other case, the entire package may be replaced with another package full of counterfeits built from the bottom up in the same structure as described in our paper. In that case, the individual authentication won't work correctly, since correct y codes cannot be counterfeited. Therefore any counterfeited package with incorrect z/y codes cannot be descendancy authenticated. In addition, the ancestry codes protect z codes as a hash code in the form of

$$h\big(LZ_{P^{(l)}}^{(l-1)}\big),$$

so the ancestry codes stored in a package do not reveal any z codes. Therefore any counterfeited package with incorrect ancestry codes cannot be ancestry authenticated.

2) Denial of Service: Our scheme is using an authentication server which stores additional information that is required for multilayer aggregate authentication hence the availability of the server is critical to our scheme. Since our scheme requires minimal computation at the server for the authentication as discussed in the previous section, Denial of Service (DoS) attacks to the server can be effectively thwarted.

3) Replay Attack: This is one of the most critical attacks against our scheme because replay attacks can be used to reuse the known hash codes to authenticate counterfeits. If the attacker counterfeits just one level of packaging, then correct z codes and ancestry codes cannot be generated. Hence our scheme can easily detect such counterfeiting attempt. The only way of counterfeiting is to construct the entire packaging hierarchy exactly described in our paper. In that case, y code cannot be correctly generated (without registered in the authentication server), and hence such packages cannot be aggregate and/or ancestry authentication.

Other types of attacks are assumed to be defended by the underlying mechanisms such as individual authentication, secured communication, and secured database and hence are out of the scope of this paper.

V. Conclusion

In this paper we proposed a novel idea of multilayer aggregate authentication and presented two modes of the scheme that authenticate multiple layers of packaging for supply chain together at the same time. Our scheme can be easily added to the existing individual authentication schemes with minimal modification as long as the authentication scheme is based on hash chain structure. With the help of computation-efficient hash functions and communication-efficient RFID devices, we can cross authenticate multiple layers of packages all together as described in this paper.

Another important feature of our schemes is that the additional computational cost required for the implementation on top of existing individual authentication schemes is minimal. With only a small number of hash code generations, our scheme adds a new level of security to the existing authentication schemes. When upper-layer packages are opened, our scheme can efficiently and effectively authenticate that the inner packages are the ones that are supposed to belong to the opened package.

Lastly our scheme provides the capability to cross authenticate a package (or a product) even without the upper-layer packages at hand. This allows even stronger authenticity of a package/product at any point in the supply chain by providing a pedigree of authenticity (in the sense that an authentication code in our scheme contains in itself the authentication codes of all upper-layer-packages in the hierarchy).

The future work will include the actual implementation of the scheme and the study of its performance and security. Another direction is the feasibility study of other cryptographic functions such as symmetric/public-key encryptions to partially/entirely replace the hash functions that we used in our scheme to further reduce the burden on the server side.

References

[1] "The economic impacts of counterfeiting and piracy." [Online]. Available: https://cdn.iccwbo.org/content/uploads/sites/3/2017/02/ICC-BASCAP-Frontier-report-2016.pdf

[2] "Information technology – radio frequency identification for item management – part 6: Parameters for air interface communications at 860 mhz to 960 mhz," International Organization for Standardization, Geneva, Switzerland, ISO/IEC 18000-6:2010.

[3] M. Lehtonen, T. Staake, and F. Michahelles, "From identification to authentication–a review of rfid product authentication techniques," in *Networked RFID Systems and Lightweight Cryptography*. Springer, 2008, pp. 169–187.

[4] S. Kwok, J. S. Ting, A. H. Tsang, W. Lee, and B. C. Cheung, "Design and development of a mobile epc-rfid-based self-validation system (mess) for product authentication," *Computers in Industry*, vol. 61, no. 7, pp. 624–635, 2010.

[5] M. O. Lehtonen, F. Michahelles, and E. Fleisch, "Trust and security in rfid-based product authentication systems," *Systems Journal, IEEE*, vol. 1, no. 2, pp. 129–144, 2007.

[6] M. Ohkubo, K. Suzuki, S. Kinoshita *et al.*, "Cryptographic approach to privacy-friendly tags," in *RFID privacy workshop*, vol. 82. Cambridge, USA, 2003.

[7] M. Saghir, "The concept of packaging logistics," in *Proceedings of Second World Conference on POM and 15th Annual POM Conference*, 2004, p. 22.

[8] U. Guin, D. Forte, and M. Tehranipoor, "Anti-counterfeit techniques: From design to resign," in *Proceedings of the 2013 14th International Workshop on Microprocessor Test and Verification*, ser. MTV '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 89–94. [Online]. Available: http://dx.doi.org/10.1109/MTV.2013.28

[9] B. King and Z. Xiaolan, "Securing the pharmaceutical supply chain using rfid," in *2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*, April 2007, pp. 23–28.

[10] C. Sun, "Application of rfid technology for logistics on internet of things," *AASRI Procedia*, vol. 1, pp. 106 – 111, 2012, aASRI Conference on Computational Intelligence and Bioinformatics. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212671612000200

[11] Y.-H. Lien, C.-T. Hsi, X. Leng, J.-H. Chiu, and H. K.-C. Chang, "An rfid based multi-batch supply chain systems," *Wireless Personal Communications*, vol. 63, no. 2, pp. 393–413, 2012.

[12] F. Rahman and S. I. Ahamed, "Efficient detection of counterfeit products in large-scale rfid systems using batch authentication protocols," *Personal and ubiquitous computing*, vol. 18, no. 1, pp. 177–188, 2014.

[13] M. H. Yang, J. N. Luo, and S. Y. Lu, "A novel multilayered rfid tagged cargo integrity assurance scheme," *Sensors*, vol. 15, no. 10, pp. 27 087–27 115, 2015. [Online]. Available: http://www.mdpi.com/1424-8220/15/10/27087

[14] A. Arbit, Y. Oren, and A. Wool, "Toward practical public key anti-counterfeiting for low-cost epc tags," in *2011 IEEE International Conference on RFID*, April 2011, pp. 184–191.

[15] H. Keni, M. Earle, and M. Min, "Product authentication using hash chains and printed qr codes," in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan 2017, pp. 319–324.