RFID and Authenticity of Goods[1]

Marlena Erdos

Introduction

Many proponents of RFID claim that RFID tags ensure the authenticity of consumer goods[2], pharmaceuticals[3], and even documents[4]. RFID tags for product authentication are being promoted as "cost-effective to implement, while being prohibitively expensive to replicate illegally[5]."

Low-cost RFID tags emit a unique identifying number[6]—effectively the tag's "name"—when queried by an RFID reader. Given that we are concerned with product authenticity, should we believe a tag (and the good that it is attached to) simply because it claims a particular name? In this chapter, we look at what makes a tag *authenticatable*, or able to prove its identity. We examine the interaction between tag authenticity and authenticity of goods. In particular, we look at an anticounterfeiting scenario that features RFID tags as a centerpiece. We also consider authentication of readers because they are a critical component of any overall system that tries to ensure product authenticity. And we also cover the special issues that arise in authenticating *people* in a multi-enterprise supply chain.

---

[1] This article originally appeared as a chapter in the book "RFID: Applications, Security & Privacy" ed Garfinkel & Rosenberg, 2005

[2] Texas Instruments "Tag-it™" www.ti.com/tiris/docs/news/news_releases/90s/rel01-23-98.shtml.

[3] "COMBATTING COUNTERFEIT DRUGS A Report of the Food and Drug Administration," February 2004. www.fda.gov/oc/initiatives/counterfeit/report02_04.html.

[4] www.compukiss.com/populartopics/travel_transhtm/article960.htm.

[5] Texas Instruments "Tag-it™" www.ti.com/tiris/docs/news/news_releases/90s/rel01-23-98.shtml

[6] The number is often referred to as an electronic product code (EPC), although in reality, an EPC is in a format specified by EPCglobal Inc., a standards body creating a variety of RFID relates technical specifications. www.epcglobalinc.org.

Before we discuss the authenticity of tags and goods, we need examine a few important concepts in authentication that serve as a foundation for the rest of the chapter[7]. (Readers who have security backgrounds should feel free to go on to the following section.)

A Few Important Concepts in Authentication

For each of the following topics, I provide a bit of background before conveying the main point.

Authentication Involves Secret Data

- Background. Colloquially, authentication is simply a matter of proving that you are who you say you are. To say that an entity is authenticated merely means we have gone through a series of checking steps such that we believe the entity's claimed identity with some assurance. Authentication in itself doesn't mean that the authenticated entity has the "right" to do anything.
- Point. Authentication systems almost always involve the use of secret data, whether that data is called a "password," "shared secret key," or "private key"[8]. This is the fundamental concept I want to drive home for our later exploration of authentication issues in RFID.

A lot more than secrets is involved in authentication, but secret data is our main issue. The secret data is combined with some of the message data via an *algorithm* (i.e., a mathematical sequence of steps); the sender and receiver typically engage in a sequence of messages called a *protocol*, whose purpose is to transport the authentication data and also to ensure that a previous legitimate message hasn't been "replayed" by an attacker. Some additional features are usually bundled into authenticated messages, including *integrity mechanisms,* which are the cryptographic equivalent of "tamper evident packaging." Authentication systems can fail because of easy-to-guess secret data, algorithms with "back doors," and "weak" protocols.

---

7 The points below are a "quick and dirty" introduction to some (but not all) important aspects of authentication systems. Readers who are interested in learning more about authentication systems are directed to one of the many fine texts on security.

8 Certain techniques, like biometrics, don't use secret data, but these don't apply to RFID since most of the entities we want to authenticate in RFID (e.g., tags, readers etc.) aren't biological entities.

The "Key Distribution" Problem

- Background. We'll call the secret data a *key*, and we'll call the message data that has been encrypted with the secret data, *authentication data*. We'll also call a "key" the data that a recipient will use to validate the authentication data. (In some systems, such as PKI, this decrypting key is not secret.) Let's say two parties want to communicate: How does the sender securely transfer the validating key to the recipient when the sender and recipient don't already have a secure way of communicating? This is called the Key Distribution Problem.
- Point. Distributing the key for validating authentication data that's been algorithmically combined[9] with a user's key is an outstanding problem in all authentication systems. Solutions to the Key Distribution Problem exist, but all solutions to date carry high costs for the infrastructure and administration needed to support them. Distributing validation keys is definitely an issue in RFID.

Stolen Keys and Revocation

- Background. An entity's secret data can be compromised, meaning that it's fallen (or even potentially fallen) into the wrong hands. Hence, there's a need to revoke a key just as there's a need to revoke a stolen credit card.
- Point. Dealing with the possible compromise of an entity's key is another outstanding issue in authentication systems. Current schemes for attempting to protect against compromised keys include setting early expiration dates, checking against a locally held list of known compromised keys, and calling out to a validation service. Each solution for dealing with a compromised key requires a fair bit of infrastructure; for example, performing a validation check will typically add many computing cycles and bandwidth, thus making it "expensive." In RFID, time and processing power are both at a premium.

---

[9] Shared secret algorithms (e.g., DES, AES) and PKI are technically encryption techniques. Another method commonly referred to as hashed message authentication code (HMAC) combines the secret data with message data using a type of algorithm that isn't "encryption" but yet is secure.

Comment on Authentication Costs

I've mentioned that significant costs and infrastructure are involved in dealing with both the key distribution problem and compromise of an entity's key. There's no avoiding these costs in an even moderately secure system. This is a seemingly gloomy picture but I prefer to think of it as "fact of life." Authentication is costly, but (to paraphrase a well-known saying) lack of authentication tends to be even costlier!

Authenticity of Tags and Authenticity of Goods

EPC RFID tags identify themselves but do not authenticate themselves [10]. When a reader tries to read RFID tags, it essentially calls out the tag's electronic product code (EPC), and a given tag responds only if the name is its own. However, the reader usually gives only the beginning of the tag's name. This is like the reader calling roll, but saying only, "Speak up if your name begins with H."

The problem here is that any entity—whether legitimate or not—can respond. A tag could claim that its name was the EPC equivalent of anything beginning with H, and the reader would believe it.

Why doesn't an EPC tag authenticate itself? Several reasons: First, for a tag to authenticate itself, it would have to hold secret data. And it would have to use the secret data via a mathematical algorithm[11] to provide proof of identity. The reader, too, would have to have access to the right key for each tag so that it could validate the tag's identity.

Given that the tags are usually provisioned with their EPC by one party (a tag manufacturer or the manufacturer of the goods that are tagged) but are read by readers belonging to entirely distinct parties (e.g., distribution centers and retail stores), there's the matter of distributing the validation keys that are associated with each tag.

But getting the tag keys from the entity that installed them to the business partners that will later read them is not an insurmountable problem. It might cost money and time to distribute the keys needed for validation, but it is

[10] The discussion applies to the low-cost tags that are part of the EPCglobal endeavor. Other tags are being produced, but the EPCglobal tags get the most attention because their low cost makes their use economically feasible for a wide variety of supply chain and retail applications.

[11] Either an encryption or keyed hash algorithm.

certainly feasible: For example, a file containing pairs of EPCs and their associated keys could be sent via secure e-mail or even via postal mail.

In this case, the problem is not key distribution but the inability of tags to use keys even if they had them.

To keep costs low, EPC tags do not contain batteries. They get their power from the RF signals that the reader emits as part of querying the tag name. As it turns out, every operation a tag must do to answer the inquiry from the reader uses up some power. Creating and validating authentication data requires a large number of operations and hence a significant amount of power.

Research is being done in creating authentication algorithms, as well as in designing devices that are especially conserving of power[12]. However, current commercial EPC tags cannot in a practical way create authentication data. Nor is there an obvious solution on the horizon to this problem of authenticating batteryless devices such as EPC tags.

Authenticity of Goods and Anticounterfeiting Measures

The manufacture and distribution of counterfeit goods is a pressing global concern. A particularly worrisome issue is the injection of counterfeit pharmaceutical goods into the supply chain. Counterfeiting is a problem everywhere, but in few other cases are lives more clearly at stake. A number of proposals suggest nonauthenticatable EPC tags, along with tamper-evident packaging and operational controls (such as checking for duplicate EPC tags) to help control counterfeiting.

Injection of Counterfeit Goods into the Supply Chain: Two Scenarios

Generally, goods are packed into cases at the manufacturer, shipped to a warehouse, and then sent to a retail store. Additionally, a case of goods may be opened at the warehouse, with only some of its individual items sent to a given retail store, as needed to manage retail inventory. A case of goods may have tamper-evident packaging and also may be tagged. Each item in a case might have its own RFID tag as well.

[12] Yüksel, K, Kaps, J-P, and Sunar, B. *Universal Hash Functions for Emerging Ultra-Low-Power Networks.* Proceeding of The Communications Networks and Distributed Systems Modeling and Simulation Conference (CNDS), San Diego, CA, January, 2004.

Scenario One: RFID's Role in Anticounterfeiting

As an RFID-tagged case leaves the manufacturer, it is scanned, recording both the case's tag and the tags of the individual items inside. The manufacturer also creates a shipping notice listing the goods (and their EPC tags) and sends the shipping notice to the warehouse (perhaps through secured e-mail or even "Web Services"). The manufacturer then updates a central database that lists the EPCs associated with the shipped items. This database tracks the tags and the tagged items. So, for example, the manufacturer will mark the state of shipped items as "Sent to Warehouse," possibly with the location of the warehouse indicated as well[13].

When the warehouse receives the goods, it scans the case and the item tags. The scanner compares the results of the scan with the shipping notice. The warehouse can thus detect whether it's received the right shipment and whether any goods have been lost or stolen en route (or added, as might happen in a deliberate attempt to harm). The warehouse also checks the central database to make sure the tags (and tagged items) are in the "right" state. In this situation, the state of each item should be "Sent to Warehouse." Any location or identity information in the database should also match the characteristics of the warehouse.

If the warehouse sees an item state that isn't "sent to warehouse," that's a red flag that something is amiss. An item state of "lost," "stolen," or "not yet shipped" might indicate that:

- The goods, while legitimate, have been incorrectly or maliciously removed from the manufacturer site. The check of the shipping notice should have detected this as well.
- The goods are not legitimate. They are counterfeits that have been tagged with EPC tags programmed with the names of legitimate goods. Writable tags make this type of counterfeiting relatively easy.

In this scenario, the tags themselves are not authenticatable. However, the use of the tags in conjunction with a database that tracks the intended state of tagged goods can alert the warehouse to possible criminal activity. A secured database provides an extra level of protection over the electronic shipping notice alone.

---

[13] I'm using item states that make sense for the example scenario. The actual states that will be kept as part of the EPC network are being developed by the EPCIS working group of EPCglobal.

Scenario Two: When RFID Tags and a Database Aren't Sufficient

A trucker delivering pharmaceuticals from a manufacturer to a warehouse (or from a warehouse to a retail outlet) is in league with counterfeiters. While the products are en route, the counterfeiters, with the trucker's cooperation, substitute bottles containing counterfeit drugs for the real ones.

To circumvent the RFID-based controls, the counterfeiters read the EPCs from each legitimate bottle and copy the EPC onto a writable tag on the new bottle. To be sure, the counterfeiters have to counterfeit the bottle and its tamper-evident packaging. But for certain high-priced pharmaceuticals, it may be worth the time and expense needed to do this well.

End result: The counterfeiters have legitimate pharmaceuticals that can be sold on the black market. And the warehouse (or the retail outlet) that receives the counterfeit drugs is none the wiser because the database check on the state of each tag works out just fine.

How Authenticatable Tags Could Help

Authenticatable tags hold secret data that they use (to create authentication data) but that they never release. A tag with secret data cannot be correctly duplicated[14]. Counterfeiters simply would not be able to create tags that properly authenticate themselves to RFID readers. Even if the counterfeiters were in league with the trucker and the database of tags and tagged goods indicated that everything was OK, the recipient of the counterfeit items could know immediately that something was very wrong.

Note that each authenticatable tag must be intimately bound up with the tamper-evident packaging on its associated product so that an attempt to refill (say) a legitimate bottle with counterfeit pills destroys the tag.

Switching the Security Burden

Anticounterfeiting schemes that rely on the database of tagged items rather than the authenticity of tags have switched the security burden from the tags to the database. For the anticounterfeiting scheme described above to work properly, access to the database that contains the tags' state must be tightly and properly controlled. But how can you control access to a database in a multi-enterprise endeavor such as a supply chain?

---

[14] Strictly speaking, devices that hold secret data can in fact be duplicated, but only at a prohibitively high cost in time and money.

It isn't easy: The access control system must allow many people from different companies to update tag (and product) state while denying access to unauthorized persons. Such multi-enterprise-, or "federation-", capable systems are not yet in full commercial production, although much work is being done to develop them. We'll look at this issue—generally referred to as federation—later in the chapter.

Authentication of Readers

Authenticating Readers to Tags

Some vendors of RFID-based systems claim that their systems protect against theft and counterfeiting because their tags respond only to their readers. But if the tags are batteryless EPC tags, this claim is false. Anyone with the ability to listen to radio transmissions of tags and readers can duplicate the protocols used to read a tag and thus impersonate a legitimate reader. There's simply no good way to determine whether or not an entity is entitled to perform an action, such as read a tag, without first performing authentication of that entity.

Can readers authenticate themselves to tags even if the tags can't authenticate themselves to the readers? Currently, the answer is no. Tags can't perform even the "cheapest" types of authentication algorithms (i.e., hashed MACs).

Let's assume for argument's sake that tags can perform hashed MAC authentication. But even in that case, there are problems. Inexpensive EPC tags simply don't have the memory to store lists of reader identities and the corresponding identity validation keys. And the tags don't have the power to call out to an enterprise server to get this information from a database.

A future possibility is that a tag could hold a few reader names and their corresponding validation keys. Note that for such a tag to be read by any of hundreds of readers, these reader "identities" couldn't be unique. Multiple readers would have to have the same identity. The identity is in effect a "role," meaning that the identity describes a job function, such as "incoming stock RFID reader," rather than being a reader's unique name.

This use of reader roles would provide some measure of security: The tag would respond only to readers that could authenticate themselves into a known role. (In this case, the very fact that a reader could authenticate itself to a tag would mean that it was "authorized" to receive the tag's identifier.)

But there's no way ever to revoke a compromised reader identity and key. If an attacker had one of the reader role identities *and* its key, there would be no way to stop the attacker from being mistakenly authorized to read tags. If the compromised reader's identity and corresponding secret data were publicly disseminated, any protection against unauthorized reading of tags would be lost.

Authenticating Readers Within an Enterprise

Enterprises hope to streamline the process of tracking goods as they move through the supply chain. Using RFID readers to read tags on pallets and cases can be faster, cheaper, and more accurate than manually hand-scanning bar codes. Some intended uses for RFID readers and tags include tracking "shrink" (i.e., loss of items), noting entry and exit of items, and locating stored items in a warehouse. The important point is that many organizations will begin to rely on the data from the readers rather than data collected by humans.

Just as tags can be impersonated, so can readers. If an organization intends to rely on reader data for mission-critical applications, it must have some way to ensure that its readers are legitimate and not "rogue" entities installed by attackers (who are possibly company insiders).

A rogue reader could make you think that you've received goods that you've ordered when in fact those goods were stolen en route or simply haven't arrived. Alternatively, a rogue reader could make you believe you've received goods you never wanted.

A rogue reader operating on a more subtle level might simply not read certain EPCs, thus forcing an enterprise to go back to costly hand methods to account for incoming and outgoing stock.

Most likely, a rogue reader will be discovered—for example, when another system has to deal with the phantom goods the reader reported on. But time, effort, and money will have been wasted in the interim.

Authentication of Users Across the Supply Chain (Federation)

Authentication of users in a multi-enterprise supply chain poses some challenges that are distinct from authentication within a single enterprise.

In a supply chain, goods can "flow" in a number of ways—via trucks, rail, ship, airplane, or a combination of these. In our first supply chain scenario, personnel at the manufacturer, warehouse, and retail store looked at and

also updated a central database to keep track of the "state" of the goods. In practice, it's also possible that conveyer personnel (e.g., truckers) might also be charged with tracking tasks. The upshot is that many people from many companies "touch" the database, either to read it or to both read and write it.

Here's the problem: Most existing authentication systems know only how to authenticate users who have *registered* with that system. In general, authentication systems have a registry that holds each user's login name and the validation key that is matched to a user's secret data. In a supply chain, the personnel belong to many different companies and typically are registered with only the authentication system of their employer.

An overly simple solution is to have everyone involved in the supply chain register with the authentication system used by the central tracking database. But this is highly burdensome for both the administrators of the authentication system and the users who need to view and update databases that track goods. Here's why:

Burden on System Administrators

It's highly impractical to register everyone who might need to update the tracking database. First, how does an administrator determine that a given person—for example, one who claims to be an employee of the warehouse—is the correct individual, works at the warehouse, and is entitled to view or update the database? Simply validating that a person deserves a login and access is time intensive and hence expensive.

Also, how would administrators handle employee turnover at each company that accesses the database? They would need to delete users who no longer are employees of partner companies and add users who are new employees or in new roles that would require more or less database access. While individual registration of users might work for a pilot project, it doesn't scale up to a real-life supply chain potentially involving thousands, or even tens of thousands of users.

Burden on Users

While there's a central database for tracking a given item, there will likely be many distinct databases for tracking diverse goods. A user in the supply chain might have to register with multiple authentication systems. As people who have had to "register" with multiple Web sites know, management of all of one's user names and passwords is cumbersome and error prone. Many users, in an attempt to simplify their lives, resort to tactics such as using the same password for most or all entities they communicate with or writing down names and passwords on notes

attached to their computer monitors. Suffice it to say that neither of these tactics is "secure." In the case of deliberate contamination of goods, such lack of security could be catastrophic.

The Answer Is Federation

In a *federated system*, companies register with each other rather than have their users cross-register. Federated registrations occur at the corporate level via both technical means (exchange of validation keys) and legal means (contracts). Individual users no longer need to register at foreign sites that are part of the federation. Instead, when a user wishes to contact a Web page at a partner institution in the federation, the user's own institution sends identity data (and possibly role data or other attributes) to the partner.

Current federation standards (such as SAML, Shibboleth, and Liberty) accomplish this process through clever use of Web-based protocols, making the process "transparent" to the user. Current federation standards also employ secret data and, typically, public key encryption algorithms for both authentication of the communicating institutions and tamper-evident packaging of protocol messages. However, many federation issues still need to be resolved, including agreement on name formats, agreement on what attributes are sent and the meanings thereof, identity mapping procedures, and the ability to maintain accountability.

Conclusions

Authenticity of goods is a pressing problem, particularly in the pharmaceutical industry, where stolen, mislabeled, or falsified drugs could cause catastrophic results. RFID tags can be a very helpful part of an overall system that authenticates and tracks products as they move through the supply chain. But it isn't a silver bullet in the fight against counterfeit goods. Critical aspects in ensuring product authenticity go beyond the RFID tag and include ways RFID readers, databases (and other supply-chain systems), and humans are authenticated and authorized[15].

The multi-enterprise aspect of the global supply chain makes authentication and authorization that much harder. While technology providers know how to solve most of the problems from a technical standpoint, real-

[15] Another hugely important aspect of RFID and product authenticity is how an RFID tag is securely attached to the product. This topic deserves its own full discussion.

world operation of a secure supply chain will require not just technology but many and varied agreements among manufacturers, distributors, retailers, technology providers, and also government agencies.

Much work is being done, but much work needs to be done. In the meantime, any company that claims product authenticity because it attaches RFID tags to goods must be looked at with a jaundiced eye.

[END]

About the author: Marlena Erdos is an independent consultant who has architected, designed, and implemented secure distributed computing systems for well over a decade. Among other things, she is a contributor the original SAML and Shibboleth federation specifications. She is reachable at marlena@acknowledgesoftware.com .