First Data™

A First Data White Paper

July 2010

# Implementing Tokenization Is Simpler Than You Think

A surprisingly simple service-based approach makes implementing end-to-end encryption and tokenization in your payment environment simpler than you might think.

By:
**Bruce Dragt,** Division Manager, Merchant Services – First Data
and
**Rob McMillon,** Director of Solution Development – RSA, The Security Division of EMC

# Executive Overview

In recent years, most merchants have followed the Payment Card Industry Data Security Standard (PCI DSS) guidelines to institute numerous security measures that have helped reduce the risk of loss or theft of sensitive cardholder data. Nevertheless, data vulnerabilities still remain and costly breaches still occur. This situation is leading many merchants to turn to solutions that exceed the current PCI DSS guidelines.

A new service-based solution set now available from major players in the payments processing industry addresses many merchant concerns. End-to-end encryption (E2EE) combined with data tokenization provides enhanced security by protecting sensitive cardholder data from the point of capture through delivery to the payment processor, and by eliminating cardholder data from the merchant's environment post-authorization. With these two technologies in place, the data handled by a merchant is far less vulnerable in the event of a breach, simply because encrypted or tokenized data is useless to a thief.

Large merchants who feel battle scarred after years of implementing new security technologies and procedures may be hesitant about undertaking yet another implementation to add E2EE and tokenization to their systems. In this case, however, the hesitation is unjustified; the service-based approach to bringing E2EE or tokenization or both to a merchant's environment is surprisingly simple.

There are only four potential activities for a merchant to do in order to add the benefit of these technologies in a service-based scenario, and depending on the merchant's current data environment and business processes, it may not even be necessary to undertake all four steps. The implementation process goes like this:

1. Discover and convert legacy data stored in a data warehouse to token numbers only if needed
2. Modify the message specification that is sent to the processor
3. Embed encryption if needed or desired
4. Make minor modifications to business processes only if needed

The implementation process is merchant friendly and can be done in a manner and on a schedule that makes the most sense for the merchant. Tokenization can be implemented independent of encryption, and vice versa, at the merchant's discretion.

The remainder of this paper describes the merchant's implementation process when E2EE and tokenization are delivered as a service from the merchant's payment processor. In most cases, there is no investment to make in new hardware, which minimizes the merchant's costs. At the same time, the merchant can reasonably expect long-term saving based on reduced PCI compliance requirements.

## Step 1: Discover and Convert Legacy Data

The merchant's first step is to inventory where legacy cardholder data resides within its network and convert the data to token numbers. This step is needed only if a merchant currently stores actual cardholder data after the payment authorization process.

Large multi-store merchants often use real post-transaction card data in back-office applications such as data analysis, marketing and customer loyalty programs. Once this legacy data is stored, it has a way of proliferating throughout the merchant's network (for example, to spreadsheets on individual workers' PCs). Unfortunately, all these places are part of the cardholder data environment (CDE) that is subject to PCI audits—even if the data is encrypted.

Merchants that do not store data long term can skip this step if desired, although the data discovery process is still recommended just to verify that sensitive data is not located in unexpected places. A data inventory gives peace of mind and may even enable a warranty from the tokenization provider.

There are automated tools (such as data loss prevention software) that can discover cardholder data wherever it's located in a merchant's environment. Once the data is discovered and inventoried, it should be converted to token numbers. The data conversion serves several purposes. One, it helps reduce the CDE for PCI compliance. Two, it yields cost saving by reducing both the need to protect the data and the scope of future audits. And three, it reduces the merchant's risk posture while protecting the merchant's customers. In this way, the merchant gets the most benefit from implementing tokenization. If, however, the legacy data is not replaced with token numbers, the merchant doesn't actually reduce its liability or its PCI burden very significantly.

The tokenization service provider should be able to conduct the entire data discovery and token conversion process. The merchant can submit the legacy data file(s) to the provider and receive tokenized data file(s) in return. In most cases, the tokenized data fit right into the back-office applications as a viable substitute for real cardholder data without disrupting current business processes.

## Step 2: Modify the Message Specification

Because tokenization is a new service that offers security features previously unavailable, it requires modifications to the message specification between the merchant and the payment processor. It's critical for the merchant to provide clear instructions to the payment processor regarding what data is being sent upstream, the format it is in and what should be returned to the merchant post-authorization. Therefore, the step of modifying the message specification is required for every merchant implementing tokenization.

Every merchant that has an established method for credit card payment processing already uses a message specification to tell the payment processor about the inbound data. For example, the specification may essentially say something like "Here is a clear text PAN and card number. Send back this number along with the authorization code." If the merchant already encrypts data at the point of sale, the message might read: "Here is an encrypted PAN and card number. Send the number back in encrypted form along with the authorization code." These critical instructions let the payment processor know how to work with the data.

To implement tokenization, the message must be modified to include processor-defined tokenization instructions such as, "Send back a token number along with the authorization code."If the merchant also chooses to embed encryption in the upstream data, the merchant must signal through the message specification that encrypted data is now taking the place of what was previously clear text data.

## Step 3: Embed Encryption for Additional Security

The process of encrypting cardholder data before sending it to the payment processor is an optional yet highly recommended step that can be applied when implementing a tokenization solution. The vendor that provides the tokenization service also is most likely the one to offer an encryption service for additional security.

Merchants that use a secure private line such as a frame relay to transmit data to the processor may not feel the need for encryption. Moreover, some merchants have already implemented encryption independent of tokenization, so they obviously would not need to acquire the service from the tokenization provider.

If a merchant chooses to implement encryption, there isn't necessarily a need to invest in new swipe terminals. Very reliable encryption schemes are available as a software-only solution. The encryption routine is added to the point of sale (POS) so that sensitive data is encrypted as closely as possible to the point of presentment by the customer. From that point on, the card data remains encrypted until it is received by the payment processor, where it is decrypted in order to route it through the processing network and complete the authorization process.

When a transaction authorization is being sent back to the merchant, the response includes a token number instead of the clear text or encrypted card data.

## Step 4: Make Minor Modifications to Business Processes if Necessary

This is a useful time for a merchant to gain a good understanding of its internal rules and processes for working with card data and to assess how (or whether) tokenized or encrypted data can impact them. To have a truly seamless implementation of tokenization, the merchant and tokenization provider need to discuss the systems that touch credit card data and understand the business requirements of those systems. This step would primarily be for large merchants that use card data for ancillary purposes beyond payment authorization.

For example, say a merchant wants to perform post-authorization bank identification number (BIN) analysis to determine who the bank is or what the card type is. Tokenization will affect this process because the BIN isn't maintained with the token number. In order to continue doing this type of analysis, the merchant would need to make slight modifications to the application.

Another more common example involves credit card validation with check digits. Some merchants perform a Mod 10 check to validate the authenticity of a card after receiving it back from the payment processor. This process is simply one more way to ensure that the data has not been corrupted before concluding a transaction. A post-authorization Mod 10 check fails every time once token numbers are used in place of real card data. Such failure is an intentional design of the token value to ensure that the token, which is a randomly generated number, does not accidentally match a real credit card number. The merchant's business process of performing a Mod 10 check post-authorization needs to be modified in such cases.

There's no reason why token numbers should break a valid business process. A merchant should work with its tokenization provider to understand what processes might be affected and how token numbers can accommodate the business need.

# What to Expect From a Tokenization Provider

Implementation can be simple—even seamless in many cases—if the tokenization provider has designed its solutions around merchants' common needs. Ideally, the tokenization provider should deliver a solution that minimizes the impact on a merchant's existing environment. Examples of what to expect are explained below:

## Data discovery

The provider should help a merchant conduct a data inventory in order to discover the places where sensitive legacy data is stored today.

## Legacy data conversion

The provider should accept the merchant's data warehouse files, convert the cardholder data to token numbers and return a fully tokenized file back to the merchant for use in back-office applications. This process is important if the merchant wants to reduce its risk as well as its PCI compliance requirements.

## Token composition and format

The tokenization provider should compose its token numbers in such a way that they easily fit into a merchant's existing environment and can never be mistaken for a real card number. For example:

- → The tokenized number should have the same number of digits as a real card number
- → There is some degree of card number preservation, such as the last four digits being the same as in the real card number
- → The randomly generated token numbers should not have any possibility of matching real card numbers
- → The tokenized numbers shouldn't start with any of the traditional numbers of the major brands: 3, 4, 5 and 6
- → The numbers of a token always fail a Mod 10 check

## Business rules modifications

In the event that application business rules require modification, the tokenization provider should assist the merchant in making the changes. If the POS application must be modified, the tokenization provider should assist in recertifying the application.

## Encryption process

The tokenization provider's encryption process should fit into the POS without significant modification or disruption. Ideally, the merchant would not need to replace existing card swipe devices to enable encryption. The service provider should assume responsibility for the encryption key management process.

## Conclusions

The steps to implement encryption and tokenization can truly be minimal and straightforward with a service-based solution. Depending on the merchant's existing environment, implementation can be relatively simple. The tokenization provider should help ensure a seamless implementation experience.

## Recommended Reading

To learn more about end-to-end encryption and tokenization and how both can help merchants secure their cardholder data and reduce their PCI liability, please see the resources below:

First Data white paper: Data Encryption and Tokenization: An Innovative One-Two Punch to Increase Data Security and Reduce the Challenges of PCI DSS Compliance

First Data white paper: Where Security Fits in the Payments Processing Chain

RSA's Speaking of Security blog: What is Tokenization and How Does It Work?

RSA's Speaking of Security blog: Business Impacts of Tokenization

PCI Data Storage Do's and Don'ts, published by the PCI Security Standards Council

First Data™

**The Global Leader in Electronic Commerce**

First Data powers the global economy by making it easy, fast and secure for people and businesses around the world to buy goods and services using virtually any form of payment. Serving millions of merchant locations and thousands of card issuers, we have the expertise and insight to help you accelerate your business. Put our intelligence to work for you.

## About the Authors

**Bruce Dragt** is division manager of Merchant Services at First Data and is responsible for driving product development globally across the company's suite of merchant products. He has held roles in many facets of the payment life cycle from card issuing to traditional merchant processing to alternative payments and mobile commerce. Prior to joining First Data, Dragt worked for i2 Technologies and Financial Settlement Matrix as vice president of product management, where he created payment solutions associated with business-to-business transactions originated through electronic exchanges. In addition, he has worked at Wells Fargo Bank and held roles in the business direct division and the wholesale Internet services group.

**Robert McMillon** is director of solution development for the joint partnership between First Data and RSA. McMillon has extensive experience in security strategy, corporate governance, business process transformation and product development. He helped start the first independent security team in the Southeast, founded the managed security services company that eventually became Verizon's managed security business, and has brought many successful security products to market.

For more information, contact your First Data Representative
or visit firstdata.com.