



Tokenization Manager v1.0.0

Customer Release Notes

Product Version: 1.0.0

Release Notes Issue Date: 05/27/2010

Last Updated: 05/27/2010

Release Type: ☒ **GA** ☐ **LGA**

1. Product Description

The SafeNet Tokenization Manager helps supplement SafeNet encryption solutions by facilitating smooth application performance and transparent end-user operation while keeping encrypted information secure in one central location. For countries with data privacy laws that require sensitive data remain in-country, tokenization offers the flexibility to offshore storage without compromising on compliance. SafeNet Tokenization offers a range of integration options so you can protect data without affecting critical IT components. SafeNet Tokenization also helps simplify audit compliance by reducing the number of auditable systems.

Format-preserving tokenization

Employed for online credit card transactions or transmission of other sensitive data, tokenization works by replacing sensitive data with tokens that retain essential information. With tokenization, you can enable databases, applications, and users to interact with sensitive data without exposing the clear text. Tokenization technology converts the sensitive data to an encrypted token in the same format as the original data, allowing the application to operate seamlessly. Masking features can also be maintained if a subset of the data needs to be available for authentication.

Audit-scope reduction

When facing an audit for PCI compliance, many organizations must identify and certify regulatory compliance for each server where sensitive data resides. Because SafeNet Tokenization replaces sensitive data in databases and applications with tokens, there are fewer servers to audit. Reducing the scope of audits will help save you time and money.

Flexible data protection

SafeNet Tokenization can protect a broad range of information, from credit card numbers, national IDs, and medical records to vehicle driver information, voter registration, and electronic transactions. A variety of integration options give you the flexibility to choose the right security technique for your environment while enabling you to protect more data types without affecting business logic, database architecture, storage systems, or other critical enterprise components. Production data can be moved or replicated to test environments with no additional process to de-identify or mask data while keeping data protected and not incurring any additional processing costs.

The Tokenization Manager is available as a Java API or as a web service.



Tokenization Manager v1.0.0

Customer Release Notes

2. Version Summary

2.1. Release Description

This release contains the following:

- Command Line Configuration
- Data Encryption and Tokenization
- Luhn Checking
- Data Masking
- Key Rotation
- Token Formatting
- Sample Code

2.2. New Features and Enhancements

Command Line Configuration

The Tokenization Manager includes 2 command line tools that enable users to:

- Configure the database tables needed to store the tokens and keys.
- Select the keys used to encrypt the data.

Data Encryption and Tokenization

Applications can use the Tokenization Manager to:

- Create a token and corresponding ciphertext when sent plaintext data.
- Return plaintext data when sent a token.
- Delete tokens and plaintext values from the database.

The Tokenization Manager leverages the cryptographic features of ProtectApp and the DataSecure. You must install and configure ProtectApp and create keys and users on the DataSecure. Instructions for these procedures are included in the Tokenization Manager User Guide.

Luhn Checking

A Luhn Check can be performed to ensure that the generated token is a valid number combination.

Data Masking

When retrieving plaintext values, the Tokenization Manager can mask data to prevent the calling application from seeing the entire plaintext value. Instead of returning the entire plaintext value, the Tokenization Manager replaces all but the last four numbers with X's.

"1234 5678 9012 3456" becomes "XXXX XXXX XXXX 3456"



Tokenization Manager v1.0.0

Customer Release Notes

Key Rotation

The Tokenization Manager includes a command line tool that enables users to rotate encryption keys in one step. Because the Tokenization Manager uses versioned keys to perform cryptographic operations, there is no need to specify a new key name or generate a new key. The DataSecure creates new key bytes for the new version, but the key keeps its original name.

Token Formatting

The application can create tokens using any of the following formats. These formats give existing applications access to the digits they need, while concealing the rest of the value.

- **Random Numbers** – The entire original value is replaced with random numbers generated by the DataSecure.
“1234 5678 9012 3456” → “6573 0815 3873 8612”
- **Sequential Tokens** – The first token generated for a table is generated using random numbers. Each subsequent token is created using a sequence based on the first token.
“1234 5678 9012 3456” → “6573 0815 3873 8612”
“5634 5678 8011 4457” → “6573 0815 3873 8613”
- **Keep the Last Four Numbers** - The original last four numbers remain. All other numbers are replaced with random digits.
“1234 5678 9012 **3456**” → “5541 9973 5127 **3456**”
- **Keep the First Six Numbers** - The original first six numbers remain. All other numbers are replaced with random digits.
“**1234 5678** 9012 3456” → “**1234 5634** 8341 9432”
- **Keep the First Two and Last Four Numbers** – The original first two and last four numbers remains. All other numbers are replaced with random digits.
“**1234 5678** 9012 **3456**” → “**1287 9811 0948 3456**”

Sample Code

Included in the software download are two sample programs:

- TokenServiceSample.java illustrates how to use the Java API to create and retrieve tokens.
- WSTokenSample.java shows how to call the Tokenization Manager through a web service.



Tokenization Manager v1.0.0

Customer Release Notes

3. Known Issues and Workarounds

Severity Classification	Definition
Critical	No reasonable workaround exists
High	Reasonable workaround exists
Medium	Medium level priority problems
Low	Lowest level priority problems

Issue#	Severity	Synopsis
90347 85298 85297	High	Performance for Bulk Inserts is Approximately the Same as Single Inserts Summary: In this release, bulk inserts do not noticeably outperform single inserts. This issue will be addressed in a future release. Workaround: For higher throughput for bulk operations, SafeNet offers a java-based migration utility. Contact our professional services organization for more information.