Persistent Data Masking Best Practices
Abstract
This article discusses the best practices for masking sensitive data using Persistent Data Masking.

Supported Versions
- Persistent Data Masking 9.1.0 - 9.3.0

Table of Contents
- Persistent Data Masking Overview ..................................................... 2
- Masking vs. Encryption ............................................................ 2
- Masking Techniques .............................................................. 3
  - General Data Masking Techniques ................................................... 3
  - Special Data Masking Techniques ................................................... 5
- Best Practices to Ensure Secure Sensitive Data ............................................ 5
  - Designate a Champion .......................................................... 6
  - Define a List of Sensitive Data Domains ............................................... 6
  - Catalog the Data Sources ........................................................ 6
  - Discover, Define, and Augment the Data Model ........................................ 6
  - Define the Data Masking Rules and Policies ............................................ 6
  - Set Up Repeatable Data Masking Plans ............................................... 7
  - Audit the Data Masking Results ..................................................... 7

Persistent Data Masking Overview
Informatica Persistent Data Masking uses proprietary algorithms to permanently and irreversibly mask sensitive data.
These algorithms and associated source code are owned by Informatica. Informatica does not share these algorithms or the associated source code with customers or other third-parties. Informatica customers have successfully used Persistent Data Masking technology to prevent exposure of sensitive data.

Masking vs. Encryption
Data masking is not equivalent to strong encryption. Decryption of encrypted data is not possible without knowledge of the encryption keys. Data masking, on the other hand, allows the possibility of identification of an individual, company, vendor, or other party unless a thorough data masking plan is put in place.
For example, masking only the first name of an employee and leaving the last name, employee number, or other personally identifiable data unchanged makes it easy for someone to identify the employee. This simple example illustrates the point that data masking must be thought of in its entirety and not at the technique level.
Masking Techniques

Informatica Persistent Data Masking techniques allow you to create realistic yet not “real” values for your data to give users the feel of working with real data.

Realistic data allow users such as developers and testers to work with reasonably accurate data values and get more accurate results. Users can produce high quality work without the risk of sensitive data exposure and stay in compliance with regulations.

Each data masking technique utilizes a different algorithm to produce the masked results. Some techniques, such as substitution and random masking, can be used on a broad range of data. Other techniques, such as SSN and credit card masking, are more specific to a data domain. Most of the data masking techniques have options such as repeatability and uniqueness.

Using the data masking techniques correctly can ensure the highest level of security and still retain the business value of the data. In general, a technique with a higher level of security such as nullification or encryption produces the lowest business value. In contrast, a technique with a lower level of security, such as random masking with blurring, produces a higher business value. Validate your masking rules for both business and security value and ensure that you use an appropriate mix of techniques.

Persistent Data Masking provides the following general data masking techniques:

- Key
- Random
- Substitution
- Nullification
- Expression

Persistent Data Masking also provides special data masking techniques specifically for the following types of data:

- Social Security number
- Credit card
- Phone number
- Email address
- URL address
- IP address

General Data Masking Techniques

You can use the following data masking techniques to mask all types of sensitive data.

Key Masking

Key masking produces repeatable results for the same source data, seed value, and masking rule options.

By altering the seed value, you can create a new set of repeatable results. Repeatable key masking is one method to ensure that related columns with the same value are masked the same way. For example, an employee ID column is the primary key of the employee master table. Assign a repeatable key masking rule to the employee master table and all other tables that contain the employee master. As an alternative, use the auto-cascade functionality to assign the masking rule to the employee master data so that the masked results are cascaded to all related tables.

The key masking technique uses a proprietary algorithm to generate new numbers, strings, or dates. Repeatability is likely required for key masking to ensure that referential integrity remains in place and the application logic still functions properly.
Random Masking

Random masking produces random, non-repeatable results for the same source data and masking rules.

Random masking is similar to key masking, but does not require a seed value and does not produce repeatable results. Use the random technique on non-key columns such as quantity or date columns or other columns that do not have additional relationships. Random masking can also be used on string fields to generate random text values.

The random technique offers options such as a range and blurring. Blurring tells the system to return a random masked value within a certain variance of the original value. If the data set contains a few outlier values that could be used for identification, do not use the blurring option to ensure that these outliers cannot be easily detected.

Substitution Masking

Substitution masking replaces a column of data with similar, but unrelated data from a data dictionary.

For example, to mask first or given names, you can use a data dictionary of first names. The dictionary can contain 1 to N number of rows. The higher the number of values, the more distributed the results. Like other techniques, substitution provides the option to make the masked results repeatable.

The substitution technique also provides the option to make the results unique or non-unique. Unique repeatable masking ensures that each original source value produces a repeatable unique masked value.

As is the case with random masking using the blurring option, repeatable substitution using the unique option produces the less secure results. The most secure option is to use non-repeatable masking with a small data set with, for example, 100 rows and configured to produce non-unique values. If users require repeatability and a larger data set, you can use a data set of 10,000 rows configured to produce repeatable non-unique results as a compromise. The non-unique setting is crucial to ensure that any outliers cannot be easily identifiable.

Nullification Masking

Nullification masking replaces a column of data with a null value.

You can use nullification masking on highly sensitive data that does not have a business value for the end users. In isolation, it is impossible to discover the original value of a column based on the null value.

Expression Masking

Expression masking applies expression logic to mask the results for a column in a table.

The expression logic can be simple or complex, depending on your requirements. You can create a simple expression with constants or a complex expression by trimming or concatenating values. You can also configure expression masking to be repeatable.

You determine the level of security in the expression with the functions and logic you use.
**Special Data Masking Techniques**

You can use the following data masking techniques to mask specific types of data.

### SSN Masking

SSN masking produces valid U.S. Social Security numbers based on the latest version of the Social Security Administration’s high value list. The SSN technique produces valid SSN values to ensure calling application logic will not throw errors when interacting with the masked values.

As with other techniques, SSN masking allows repeatability with a seed value. To ensure the highest level of security, turn OFF repeatability. If repeatability is required by the business, then turn ON repeatability, but alter the seed value for every database refresh.

Note that the Social Security Administration no longer produces the list since the number of unused values is diminishing.

### Credit Card Masking

Credit card masking produces valid credit card numbers while leaving the credit card issuer, such as Visa or MasterCard, untouched.

The digits are randomly generated and an appropriate checksum is applied to the final digit. The credit card masking technique produces valid credit card values to ensure that the calling application will not generate errors when it interacts with the masked values.

### Phone Masking

Phone masking generates random phone numbers while keeping the supplied format intact. Phone masking does not allow repeatability.

### Email Address Masking

Email masking generates a random string of characters for the recipient and the domain name. Email masking does not allow repeatability.

### URL Masking

URL masking generates a random URL while keeping the protocol, such as FTP or HTTP, intact. URL masking does not allow repeatability.

### IP Address Masking

IP address masking generates random IP addresses within the range allowed for network IP addresses. IP address masking does not allow repeatability.

### Best Practices to Ensure Secure Sensitive Data

A thorough and well thought out data masking plan ensures maximum security and retains the highest business value.

Use the following best practice guidelines to ensure that data masking results in secure sensitive data:

1. Designate a champion.
2. Define a list of sensitive data domains.
3. Catalog the data sources.
4. Discover, define, and augment the data model.
5. Define the data masking rules and policies.
6. Set up repeatable data masking plans.
7. Audit the data masking results.

These best practices can help you create a high value non-production environment that not only provides substantial value to the users, but also ensures data privacy.

**Designate a Champion**

Designate a data security officer or Chief Information Security Officer (CISO), or other responsible, accountable, and authoritative person as champion of the project.

This person must be responsible for the overall data privacy initiative. This person must be high enough in the organizational level to move the project forward and get buy-in from key stakeholders, including business owners, IT, data analysts, off-shore groups, and QA.

**Define a List of Sensitive Data Domains**

Research which regulations apply to your organization and determine the type of data that is sensitive or confidential to the business.

Define a list of sensitive or confidential data domains, such as first name, last name or credit card, regardless of the database or application. Describe the characteristics of each data domain, including the probable data type, descriptions, and data and metadata patterns. This step ensures collaboration between business, security, data governance, and IT.

**Catalog the Data Sources**

Create a catalog of each data source.

Document the following types of information:

- Accessed from: onshore, offshore, or both.
- Usage: development, QA, testing, training or other types of usage.
- Database type: Oracle, MS SQL Server, IMS, DB2 for z or other databases.
- Data movement: list of data feeds into the environment.
- Frequency of refresh: yearly, quarterly, ad hoc, or other intervals.
- Owners: database and application owners.
- Risk level: high, medium, or low.

**Discover, Define, and Augment the Data Model**

Load the metadata of each physical data source then augment the model with data discovery to uncover logical table relationships and to identify sensitive data. Build and confirm data domain assignments to sensitive data columns. Build cascades into the data model to ensure that the relational integrity remains intact.

**Define the Data Masking Rules and Policies**

Define a set of data masking rules based on the available data masking techniques.

Choose masking techniques and options that ensure the highest level of security required with the appropriate level of business value.
For example, you can use the following techniques:

- Nullify highly sensitive data.
- Use non-unique repeatable substitution with no more than 1,000 values.
- For all numeric sensitive columns use random masking with a range and with no blurring.
- Mask all key fields with repeatable key masking and alter the seed value each time you refresh.
- Use special techniques like credit card masking with the repeatability turned OFF, unless required by the business users. Alter the seed value each time you refresh.

Once you have a vetted inventory of masking rules, build a set of masking policies that must be used for each and every environment.

**Set Up Repeatable Data Masking Plans**

Assign masking policies to all sensitive data columns and set up a masking plan that includes all the policies for each identified data source. Verify that the masking plan execution is part of the standard non-production data provisioning process.

**Audit the Data Masking Results**

Set up independent masking validation rules to verify that all sensitive data are masked in non-production data sources before you turn them over to end users. Run standard reports before and after data masking to show what data masking was set up and what was actually carried out.

**Author**

Jay Hill  
Director of Product Management and Marketing