BMS COLLEGE OF ENGINEERING

(Autonomous College under VTU)

Bull Temple Road, Basavanagudi, Bangalore - 560019

A project report on

"Masquerade - Data Masking Tool"

Submitted in partial fulfillment of the requirements for the award of degree

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

By

Rozelle Jain (1BM13IS066)

Shantanu Das (1BM13IS073)

Majid Nikbakhsh Keivani (1BM13IS098)

Under the guidance of

Vineetha B. Y.

ISE Faculty

Department of Information Science and Engineering

2016-2017

BMS COLLEGE OF ENGINEERING

(Autonomous College under VTU)
Bull Temple Road, Basavanagudi,
Bangalore – 560019

Department of Information Science and Engineering

CERTIFICATE

This is to certify that the project entitled "*Masquerade - Data Masking Tool*" is a bona-fide work carried out by **Rozelle Jain (1BM13IS066), Shantanu Das (1BM13IS073), Majid Nikbakhsh Keivani (1BM13IS098)** in partial fulfillment for the award of degree of Bachelor of Engineering in **Information Science and Engineering** from **Visvesvaraya Technological University, Belgaum** during the year **2016-2017**. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

Vineetha B. Y.	**Dr. Gowrishankar**	**Dr Mallikharjuna Babu**
ISE Faculty	**Professor and HOD**	**Principal**

Examiners

Name of the Examiner **Signature of the Examiner**

1.

2.

ACKNOWLEDGMENT

We are very grateful to our institution BMS College of Engineering, with its ideals and inspiration for having provided us the facilities, for the completion of the project.

We are very happy to acknowledge the numerous people involved in helping us with the project Masquerade – Data Masking Tool.

We would start by expressing our sincere thanks to Dr. Mallikharjuna Babu K, Principal, BMSCE. We extend our thanks to Dr. Gowrishankar, Head of the Department, Information Science and Engineering, BMSCE, for helping us with the encouragement and motivation to implement our ideas into a project.

We wish to express our deep sense of gratitude to our Project Guide Vineetha B.Y., Information Science and Engineering, BMSCE, for her continuous and tireless support. We would also extend special thanks to Dr. Sandeep Verma, Assistant Professor, Information Science and Engineering, BMSCE and Mr. Raghu Rajalingam, for assisting us at various steps of the project and continuous advises and motivation.

We would like to thank the teachers of Information Science and Engineering Department for developing the art of programming in us which was a great help in making of this project. We would also like to thank the teaching and non-teaching staff, laboratory staff for their cooperation and support.

Lastly, we would like to thank our friends and family for their endless support and confidence in us.

Rozelle Jain

Shantanu Das

Majid Nikbakhsh Keivani

ABSTRACT

In today's information age, the data is an important asset of the organization. Enterprises share data from their production applications with other users for a variety of business needs. Most organizations if not all copy production data into test and development environments to allow system administrators to test upgrades, patches and fixes. Retail companies share customer point-of-sale data with market researchers and data miners to analyze customer buying patterns. As a result application developers and data miners require an environment mimicking close to that of production to analyze the data or build and test the new functionality.

As a result of the above, organizations copy tens of millions of sensitive customer and consumer data to non-production environments and very few companies do anything to protect this data.

Here data masking comes to exist where it allows data miners, developers, testers, and administrators to work with data and databases, without exposing them to sensitive data. It is the process of replacing existing sensitive information of test or development databases with information that is realistic but not real. Data masking techniques will obscure specific data within a database table ensuring data security is maintained.

Our application assists organizations in achieving data security at different level. Firstly, at development level. Databases can be masked so that developers are provided with only sufficient amount of information. Secondly, at testing level since masking production data will allow to run various tests with no loss in authenticity of data. Thirdly, at client level since masking sensitive information will bar malicious users from accessing unauthorized data. And lastly, at data mining level. By removing sensitive data while preserving analytic and data-mining capabilities any organization can provide access of database to data-mining researchers.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

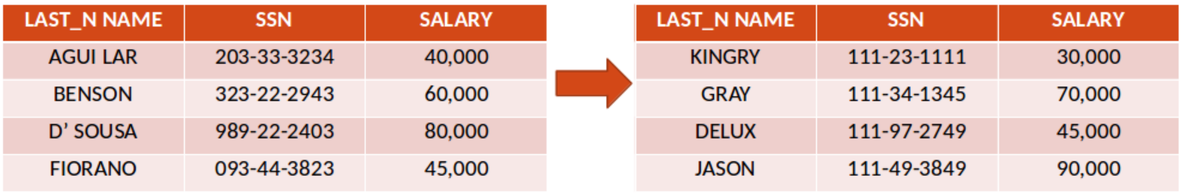
1.1 Defining Data Masking

Data masking is a method of creating a structurally similar but inauthentic version of an organization's data that can be used for purposes such as software testing and user training. The purpose is to protect the actual data while having a functional substitute for occasions when the real data is not required.

Data masking doesn’t prevent actual access to the data; it is a protection mechanism that hides the data itself, like a mask. This is particularly important for private data, like credit card information and personal identifiable information (PII). A developer, tester, or database administrator should be able to access a database in order to complete their daily tasks; however they shouldn’t be able to extract a list of actual credit card numbers or any other sensitive data.

Data masking hides the actual data. There are a variety of different algorithms for masking, depending on the requirements. Simple masking just turns characters to blank, so, for example, an e-mail address would appear as xxxxxxx@xxxxxx.com.

In data masking, the format of data remains the same; only the values are changed. The data may be altered in a number of ways. Whatever method is chosen, the values must be changed in some way that makes detection or reverse engineering impossible. Therefore production like data can be provided avoiding the exposure of original data.



LAST_N NAME	SSN	SALARY
AGUI LAR	203-33-3234	40,000
BENSON	323-22-2943	60,000
D’ SOUSA	989-22-2403	80,000
FIORANO	093-44-3823	45,000

LAST_N NAME	SSN	SALARY
KINGRY	111-23-1111	30,000
GRAY	111-34-1345	70,000
DELUX	111-97-2749	45,000
JASON	111-49-3849	90,000

Fig.1.1 Original vs Masked Data

1.2 The Need For Data Masking

- Compliance to Legal Requirements – There are obligations of a data holder to protect the information they maintain and these are becoming increasingly rigorous in every legal jurisdiction. It is a safe assumption that the standards for the security and maintenance of data will become increasingly strict in the future.

- Preventing Loss of Confidence – In case of a data leak at any organization, the formal legal sanctions applied by governmental bodies is one of the many problems faced. Often the costs of such an event can far exceed any fines levied for the violation of the rules. For example, what will it cost the organization if potential customers are not willing to provide sensitive information to your company because they read an article about a data escape in the newspaper. Dealing with the public relations aftermath of seeing the companies name in the press will not be cheap. The public relations costs of a data escape shows poorly of the organization.

- Avoiding Accidental Exposure – The risk of accidental exposure of information is often neglected when considering the security risks associated with real test data. Often it is thought that there is no point in masking the test data because everybody has access to production anyways. But by masking the most sensitive information such as(credit card numbers, customer email addresses etc) we can mitigate the damage associated with accidental exposure and the masked databases remain just as functional.

- Data breach prevention – Most people think the major risk to the information they hold is external entities out to break in and steal the data. The assumption then follows that protecting the network and firewalls is the appropriate and sufficient response. There is no denying that such protection is necessary however it has been shown that in many cases the data is stolen by malicious insiders who have been granted access to the data. No firewall can keep an insider from acquiring data under such circumstances. However, by reducing the number of databases with unmasked information, the overall risk of exposure is mitigated. The external hackers, if they get through the network security, will have far fewer usable targets and a far greater proportion of the inside personnel will have no access to the real data.

CHAPTER 2

LITERATURE SURVEY

Authors has undergone literature review phase and evolved with the problem statement with the help of work, has published till today in the area of data masking:

Oracle White Paper [1] - Data Masking Best Practices: This paper describes why data masking and challenges of masking production data for non-production environment and the best practices for deploying Oracle Data Masking to protect sensitive information in Oracle databases and other heterogeneous databases such as IBM DB2, Microsoft SQL Server.

Informatica [2] - Best Practices for Dynamic Data Masking and Securing Production Applications and Databases in Real-Time: Sensitive data, such as financial records and personal employee or customer information, needs to be protected, both to safeguard it from unauthorized eyes and to comply with a growing number of privacy regulations around the world. At the same time, enterprise environments are becoming ever more heterogeneous and complex, requiring increasing cost and effort to monitor and protect the data they contain. Dynamic Data Masking (DDM) cost-effectively adds an extra layer of data security by customizing the level of data masking, scrambling, or blocking at the individual level. With DDM, IT organizations can give authorized users the appropriate level of data access without changing a single line of code or the database.

Securosis [3] - Understanding and Selecting Data Masking Solutions: In this paper, they have discussed why customers buy masking products, how the technology works, and what to look for when selecting a masking platform. As with all Securosis research papers, our goal is to help end users get their jobs done. So we will focus on how to help you, would-be buyers, understand what to look for in a product. We will cover use cases at a fairly high level, but we'll also dig into the technology, deployment models, data flow, and management capabilities, to assist more technical buyers understand what capabilities to look for.

Net 2000 Ltd. [4] - Data Masking: What You Need to Know: This paper provide general information about data masking and makes the reader to get answer to some of questions like, Who will be the end users of the masked data? What sort of depth of masking (granularity) of the anonymous data will be required in the resulting databases? How large is the data to be masked? Is there a time window during which the masking operations must be completed? Which data should be masked and in which tables and columns are these data items located? What masking techniques will be used on the various data types? Are the data items to be masked involved in any relationships with other data items?

Muralidhar, et.al [5] - Describes how to maintaining the Relationship between confidential and Non- Confidential Attributes in Statistical Databases for data masking.

Parsa, et.al [6] - Discuss the general method for Data Perturbation. This study also serves to unify commonly used perturbation techniques for numerical confidential variables under one umbrella, and allows for comparison of these techniques using the ideal requirements for data utility and disclosure risk. In developing this new theoretical basis, the authors have define the ideal measures of data utility and disclosure risk. Maximum data utility is achieved when the statistical characteristics of the perturbed data are the same as that of the original data. Disclosure risk is minimized if providing users with micro-data access does not result in any additional information.

Muralidhar, et.al [7] - Presented "The Two Step Data Shuffle: A New Masking Procedure," In this paper, they have described a new shuffling procedure for masking confidential data. The advantages of this approach can be summarized as follows: (1) The released data consists of the original values of the confidential variables (i.e., the marginal distribution is maintained exactly), (2) All pair-wise monotonic relationships among the variables in the released data are the same as those in the original data which we could implement in our application.

Ravi kumar GK, et.al [8] - Proposed a uniform architecture for data masking using random replacement. This paper addresses the necessity of data masking in present information age, the author consolidated all the data masking techniques and importance of data masking for realistic situations. Comparison study of various techniques with the replacement method with respect to response time and our results shown random replacement is strongest method of data masking used across all domain which gives maximum confidence for all the customers and data masking.

After a thorough reading of the papers mentioned above we were able to identify the guidelines following which the web application could be developed. We got an insight of all the techniques and algorithms used for masking. We were also introduced to the conventions followed by large organizations for data security which in turn helped us to develop a solution that could fit in any existing system without compromising on accuracy and efficiency.

CHAPTER 3

AIM AND SCOPE

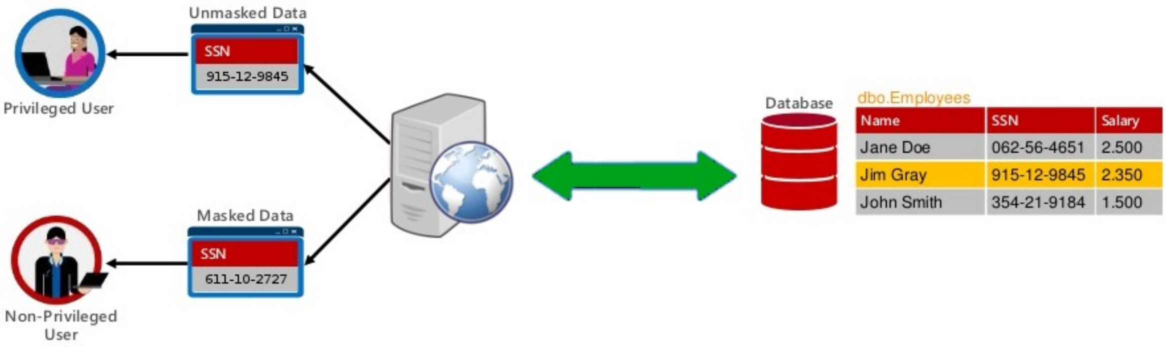
3.1 Aim

The Data Masking tool helps organizations share production data in compliance with privacy and confidentiality policies by replacing sensitive data with realistic but scrubbed data based on masking rules.

The main objectives of the data masking tool are:

- It is necessary to replace sensitive data with non-sensitive production like data that looks and acts like original.
- Data is used to support business function but it's propagation carries serious risk. To reduce the likelihood of theft of this data, it is important to eliminate unwanted access and unnecessary copies of sensitive data..
- The masking tool should be platform independent. It should adapt to any of the underlying operating system present in the organization or department.
- It should be easy for the admin to set rules for each column in each table. There should be an interactive user interface for this data masking tool.
- Time required to query any database is vital in the working of any application designed in the organization. Thus, the masking tool should inject negligible time lag in the process of fetching data.
- Processing and caching the same set of data occupies a considerable amount of space. This is a redundant and repetitive procedure. The data masking tool should aim at using least amount of space yet producing correct data efficiently.
- The masking architecture should be designed to fit within existing data management framework and mitigate risk to information without sacrificing usefulness.

The figure below shows the application design that helps to achieve the underlined aims:



Name	SSN	Salary
Jane Doe	062-56-4651	2.500
Jim Gray	915-12-9845	2.350
John Smith	354-21-9184	1.500

Fig.3.1 Application Design

3.2 Scopes

The Data Masking Tool has application in variety of aspects namely:

- Development of softwares and applications requiring database information
- Analysis of accumulated data where exposure of part of data is sufficient
- Display of statistics to client where distribution and frequency of data in some fields is all the client needs to access. The rest can be dummy production -like data or even mulled out.
- Outsourcing information off-shore where exposure of sensitive data can adversely affect business.
- Testing softwares and application where dummy data is manipulated to quantify various features of the application like integration, scalability, etc. However the dummy data used in the tests should be production-like for accurate results.
- Exposing data within organization in need- to-know basis is a policy of every organization to prevent malicious exposure and leakage.

CHAPTER 4

METHODOLOGY

Data masking can be achieved by implementing various rules but those rules must adhere to the laws of data masking at all times.

4.1 Five Laws Of Data Masking

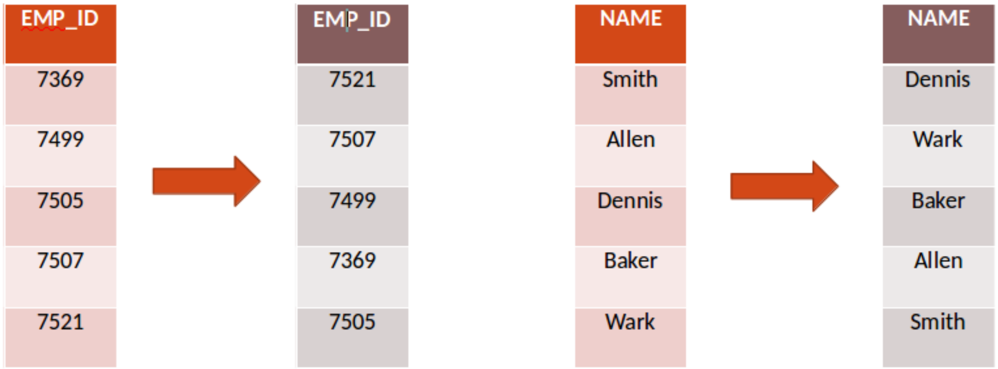
- **Masking must not be reversible** - However you mask your data, it should never be possible to use it to retrieve the original sensitive data.
- **The results must be representative of the source data** - The reason to mask data instead of just generating random data is to provide non-sensitive information that still resembles production data for development and testing purposes. This could include geographic distributions, credit card distributions (perhaps leaving the first 4 numbers unchanged, but scrambling the rest), or maintaining human readability of (fake) names and addresses.
- **Referential integrity must be maintained** - Masking solutions must not disrupt referential integrity; if a credit card number is a primary key, and scrambled as part of masking, then all instances of that number linked through key pairs must be scrambled identically.
- **Only mask non-sensitive data if it can be used to recreate sensitive data** - It isn’t necessary to mask everything in your database, just those parts that you deem sensitive. But some non-sensitive data can be used to either recreate or tie back to sensitive data. For example, if you scramble a medical ID but the treatment codes for a record could only map back to one record, you also need to scramble those codes. This attack is called inference analysis, and your masking solution should protect against it.
- **Masking must be a repeatable process** - One-off masking is not only nearly impossible to maintain, but it’s fairly ineffective. Development and test data need to represent constantly changing production data as closely as possible. Analytical

data may need to be generated daily or even hourly. If masking isn't an automated process it's inefficient, expensive, and ineffective.

4.2. Techniques of Data Masking

Following these laws our objective of masking the data is achieved, by implementing various masking techniques which are cited below:

- **Shuffle -**The Shuffling technique uses the existing data as its own substitution data-set and moves the values between rows in such a way that the no values are present in their original rows i.e shuffling is similar to substitution except that the substitution data is derived from the column itself.



EMP_ID
7369
7499
7505
7507
7521

EMP_ID
7521
7507
7499
7369
7505

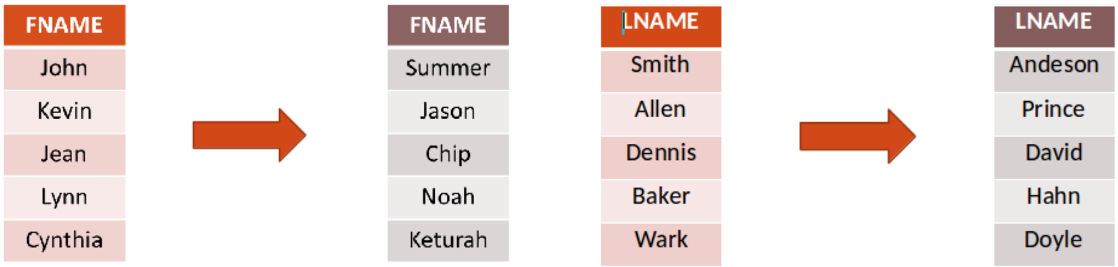
NAME
Smith
Allen
Dennis
Baker
Wark

NAME
Dennis
Wark
Baker
Allen
Smith

Fig.4.1 Shuffling Of Any Data Types

- **Random Substitution -** This technique consists of randomly replacing the contents of a column of data with information that looks similar but is completely unrelated to the real details.

  - **First Names/ Last Names**



FNAME
John
Kevin
Jean
Lynn
Cynthia

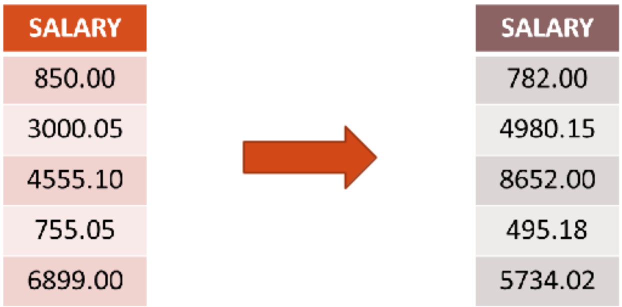
FNAME
Summer
Jason
Chip
Noah
Keturah

LNAME
Smith
Allen
Dennis
Baker
Wark

LNAME
Andeson
Prince
David
Hahn
Doyle

Fig.4.2 Substitution Of First Names And Last Names

- **Integer / Decimal –** This technique makes sure that while replacing the value, the number of digits on the left and right of decimal is maintained.

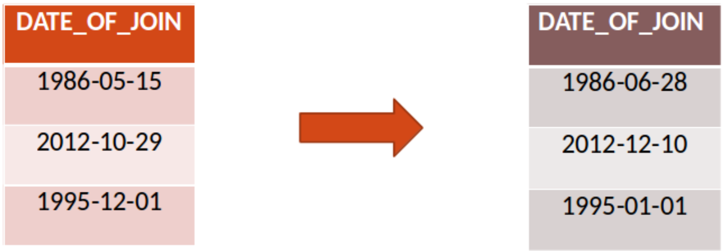


SALARY
850.00
3000.05
4555.10
755.05
6899.00

SALARY
782.00
4980.15
8652.00
495.18
5734.02

Fig.4.3 Substitution Of Integer And Decimal Values

- **Date** - This technique replaces the existing data with random date values, the year is preserved and the day and months are replaced.

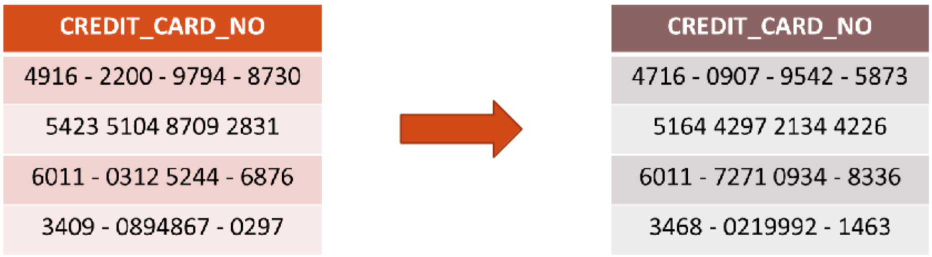


DATE_OF_JOIN
1986-05-15
2012-10-29
1995-12-01

DATE_OF_JOIN
1986-06-28
2012-12-10
1995-01-01

Fig.4.4 Substitution Of Date Values

- **Credit Card** - The credit card numbers are replaced by randomly generated card numbers. However, the first four digits indicating the type of card is maintained. Also Luhn algorithm is used to make sure these random credit card numbers are valid.



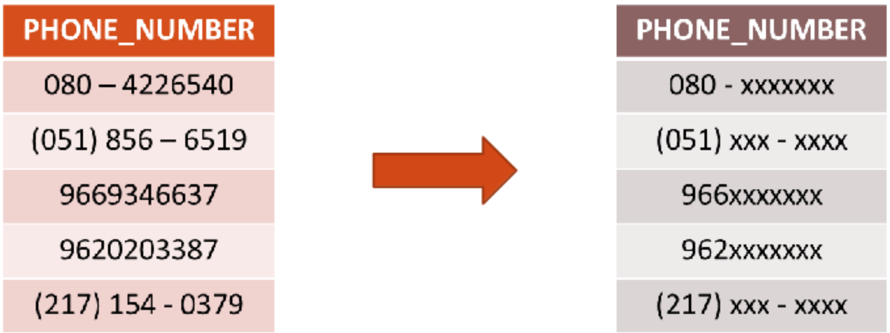
CREDIT_CARD_NO
4916 - 2200 - 9794 - 8730
5423 5104 8709 2831
6011 - 0312 5244 - 6876
3409 - 0894867 - 0297

CREDIT_CARD_NO
4716 - 0907 - 9542 - 5873
5164 4297 2134 4226
6011 - 7271 0934 - 8336
3468 - 0219992 - 1463

Fig.4.5 Substitution Of Credit Card Using Lhun Algorithm

- **Nulling Out** - The Nulling Out technique simply removes the sensitive data by deleting it Simply deleting a column of data by replacing it with NULL values is an effective way of ensuring that it is not inappropriately visible in test environments.

◆ **Right Part Nulling(Varchar)**

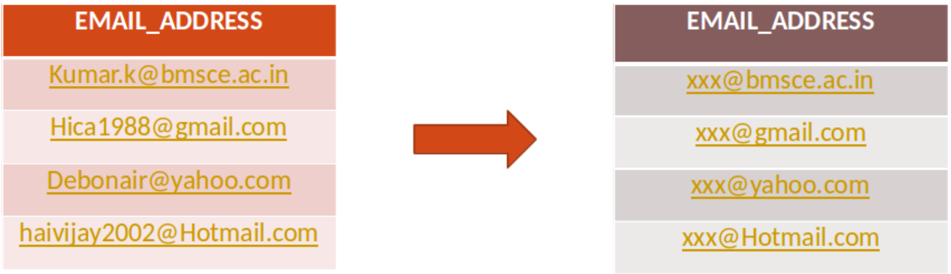


PHONE_NUMBER
080 – 4226540
(051) 856 – 6519
9669346637
9620203387
(217) 154 - 0379

PHONE_NUMBER
080 - xxxxxxx
(051) xxx - xxxx
966xxxxxxx
962xxxxxxx
(217) xxx - xxxx

Fig.4.6 Nulling Out Of Varchar Values

◆ **Nulling Out (Email)**

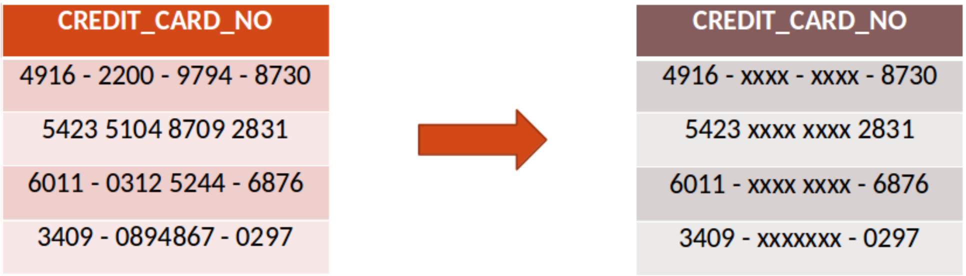


EMAIL_ADDRESS
Kumar.k@bmsce.ac.in
Hica1988@gmail.com
Debonair@yahoo.com
haivijay2002@Hotmail.com

EMAIL_ADDRESS
xxx@bmsce.ac.in
xxx@gmail.com
xxx@yahoo.com
xxx@Hotmail.com

Fig.4.7 Nulling Out Of Emails

◆ **Nulling Out (Credit Card)**



CREDIT_CARD_NO
4916 - 2200 - 9794 - 8730
5423 5104 8709 2831
6011 - 0312 5244 - 6876
3409 - 0894867 - 0297

CREDIT_CARD_NO
4916 - xxxx - xxxx - 8730
5423 xxxx xxxx 2831
6011 - xxxx xxxx - 6876
3409 - xxxxxxx - 0297

Fig.4.8 Nulling Out Of Credit Cards

4.3. Reverse Proxy Server

After all the masking techniques have been implemented we have eliminated unwanted access. All the unauthorized users are bared from connect to database. This is achieved by Reverse-Proxy server.

A reverse proxy (or surrogate) is a proxy server that appears to clients to be an ordinary server.

A reverse proxy taking requests from the Internet and forwarding them to servers in an internal network. Those making requests connect to the proxy and may not be aware of the internal network.

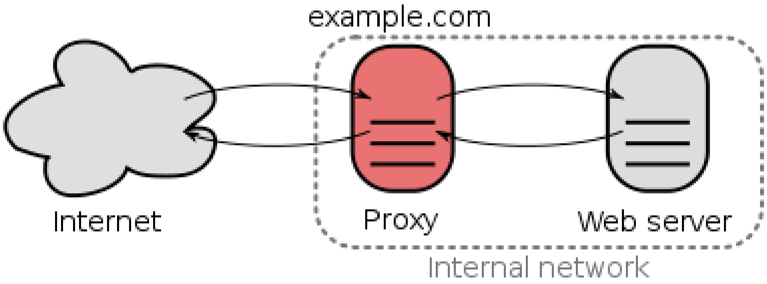


Fig.4.9 Reverse Proxy Server

4.4. Tools Used

Java as programming language: Among all Java is one of the most popular and efficient languages and continues to be one of the most popular programming languages in the world.

We can list some but not all pros of java as:

- Java is an excellent language for developing cross-platform desktop applications.
- Vast array of third party libraries.
- Huge amount of documentation available.
- Large pool of developers available.
- Platform ubiquitous.

- Excellent performance.
- Excellent specification.
- Sturdy garbage collection.
- Managed memory.
- Tool availability - IDEs like Eclipse & NetBeans which are free.

Google Web Toolkit (GWT) as Java framework:

Knowing all the above mentioned advantages of Java we started looking for different Java frameworks and figured out that GWT (Google Web Toolkit) would suit our needs.

GWT is an open source web development framework that allows developers to easily create high-performance AJAX applications using Java.

GWT is a development toolkit for building and optimizing complex browser-based applications. Its goal is to enable productive development of high-performance web applications without the developer having to be an expert in browser quirks, XMLHttpRequest, and JavaScript. The GWT SDK provides a set of core Java APIs and Widgets. These allow you to write AJAX applications in Java and then compile the source to highly optimized JavaScript that runs across all browsers, including mobile browsers for Android and the iPhone.

MySQL Proxy is an application that communicates over the network using the MySQL client/server protocol of which it can be used without modification with any MySQL-compatible client that uses the protocol.

MySQL helps to create a relational database.

By using these tools we achieved a platform independent web application with an interactive user interface, through the admin side and user side interactions which are shown in the figures below:

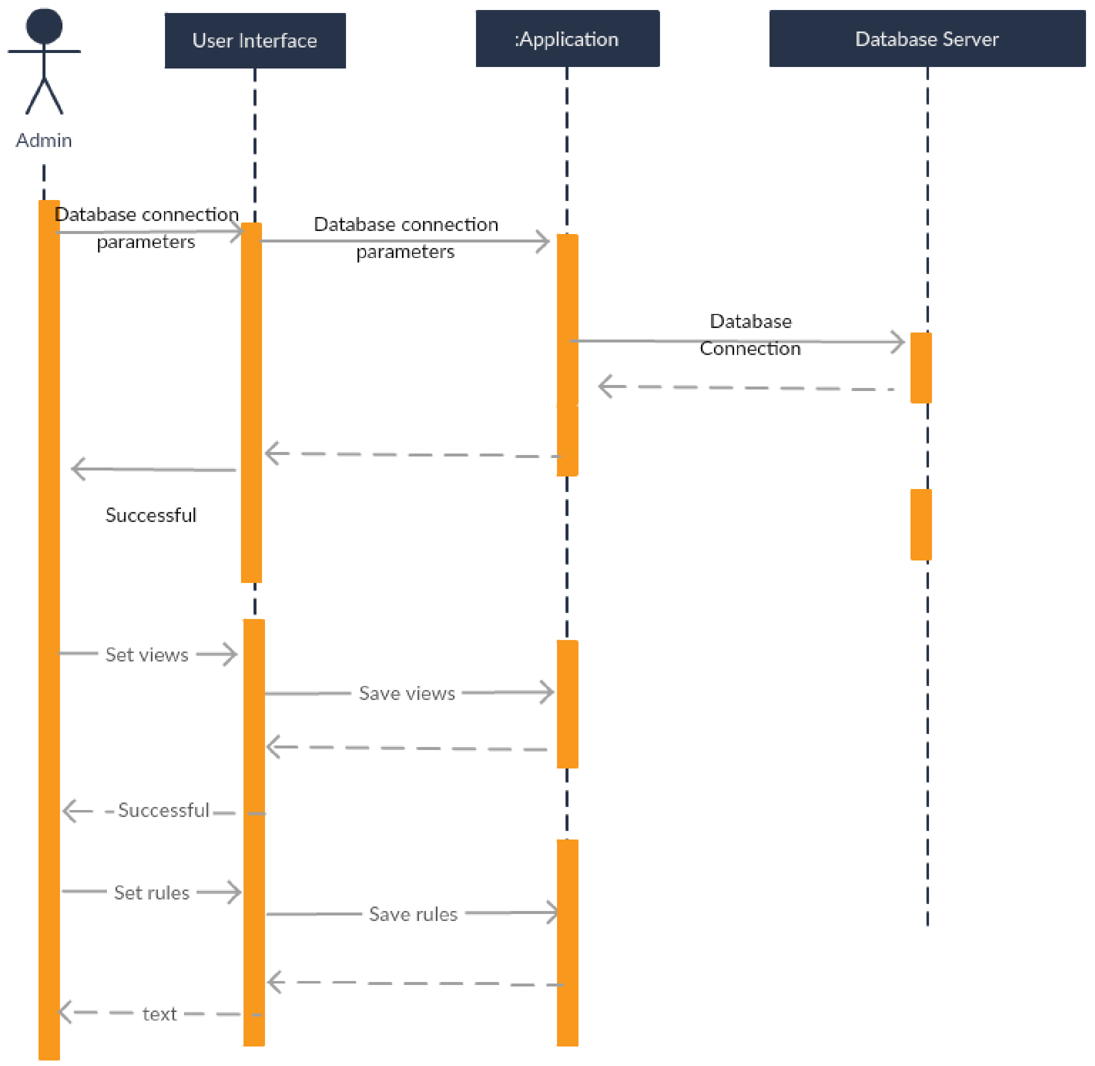


Fig.4.10 Admin Side Interaction Diagram

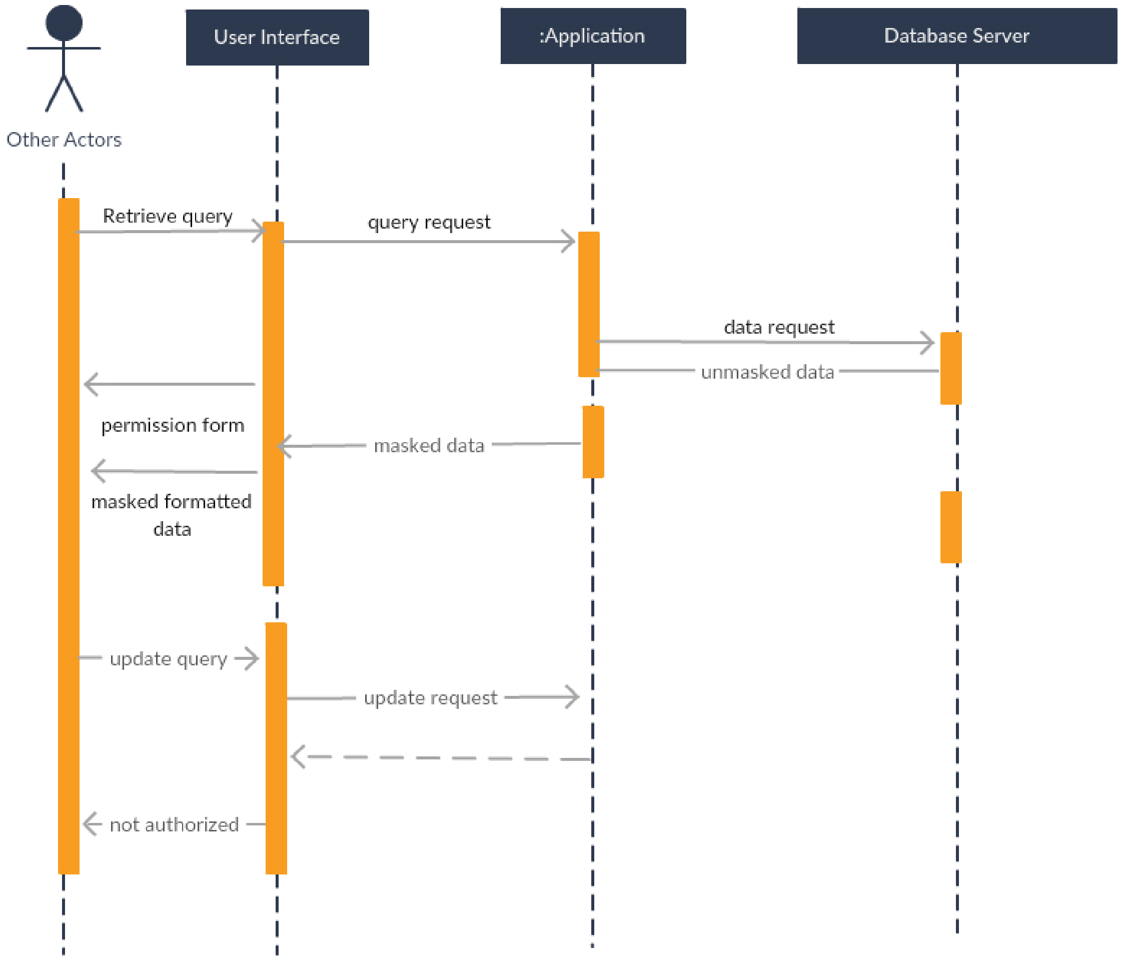


Fig.4.11 User Side Interaction Diagram

4.5. Models of Data Masking

Static data masking procedures involve creating a copy of live database and replacing actual data with fake one. It is the only method of data masking used by companies sending their databases to outsourced software specialists for testing.

With static data masking, most of the DBAs, programmers, and testers never actually get to touch the production database. All of their work is done on the dummy test database.

This provides one level of protection, and is necessary in many environments. However, it is not a complete solution because it does not protect authorized users from viewing and extracting unauthorized information.

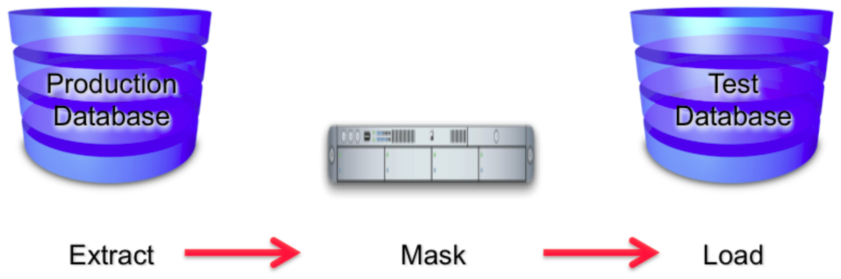


Fig.4.12 Static Data Masking

The following concerns should be noted when using static database solutions.

- With static solutions the database information is extracted as-is from the database, and only then it's transformed. We have to hope or trust that the masking solution will finally delete the real data, and that the static masking solution is working on a secure platform that was not compromised.
- The live database is not protected from those who do have permissions to access the database, like administrators, QA, developers, and others with access to the actual live database. These personnel can access actual data records, which are not masked.
- Having a full test database that is a copy of the full production database, there are costs is in the hardware and maintenance of the second system.

Dynamic data masking, in turn, involves replacing sensitive data with fake values on-the-fly, while the data is being transferred to a client. In other words, data masking software changes the way database responds a query, so it requires no interference to the database itself and the real database entries remain untouched. The data is masked before it exists the database so it is a very secure and reliable method.

When a DBA or other authorized personal views actual data in the production database, data is masked, or garbled, so the real data is not exposed. This way, under no circumstances is anyone exposed to private data through direct database access.

Using a proxy, the dynamic masking tool investigates each query before it reaches the database server. If the query involves any sensitive data, the data is masked on the database server before it reaches the application or the individual who is requesting the data.

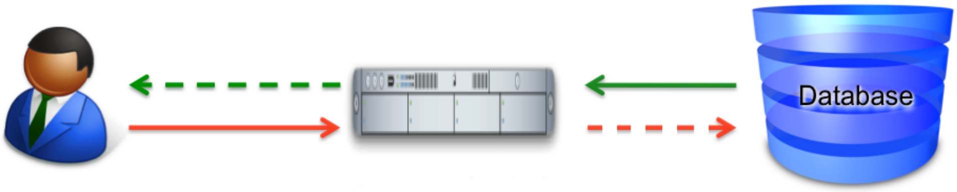


Fig.4.13 Dynamic Data Masking

The following concerns should be noted when using dynamic database masking solutions.

- Response time for real-time database requests. In environments where milliseconds are of crucial importance, dynamic masking needs to be carefully tested to ensure that performance meets the organization standards. Even when a particular item of data is not masked, the proxy does inspect the incoming request.
- Security of the proxy itself. Any type of software installed on the database server needs to be secure. And once a proxy is present, you have to enforce that the entire connections to the database are now passing through this SQL proxy. Bypassing this proxy in anyway, will result in access to the sensitive data without masking.
- Performing of database development and testing on live systems can cause errors in the production system. In many cases, DBAs perform changes on a limited part

of the system before deploying. However, best practices would require a separate database for development and testing.

Advantages of static data masking

- Allows development and testing without affecting live systems
- Best practice for working with contractors and outsourced developers, DBAs, and testing teams
- Provides a more in-depth policy of masking capabilities
- Allows organizations to share the database with external companies

Advantages of dynamic data masking

- The sensitive information never leaves the database
- No changes are required at the application or the database layer
- Customized access per IP address, per user, or per application
- No duplicate or off-line database required
- Activities are performed on real data, saving time and providing real feedback to developers and quality assurance

Weighing the features offered by both the methods of masking we chose to implement the data masking tool using static technique. By doing so we were able to achieve the objective of using minimum amount of space while injecting minimum time lag in producing accurate results.

4.6. System Architecture

4.6.1. High Level Design

Our application sits in between the database and client/users. It provides authentication of users to provide original or masked data based on user login.

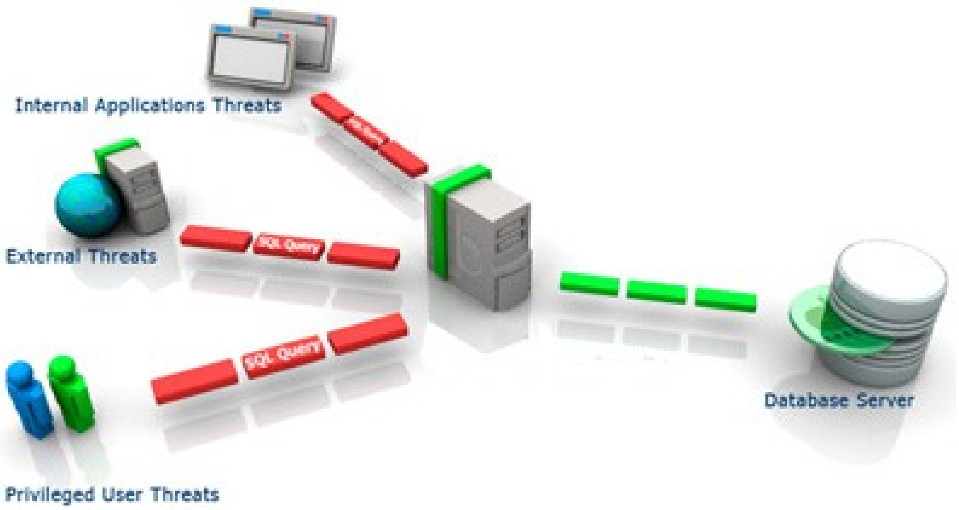


Fig.4.14 High Level Design

4.6.2. Detailed Design

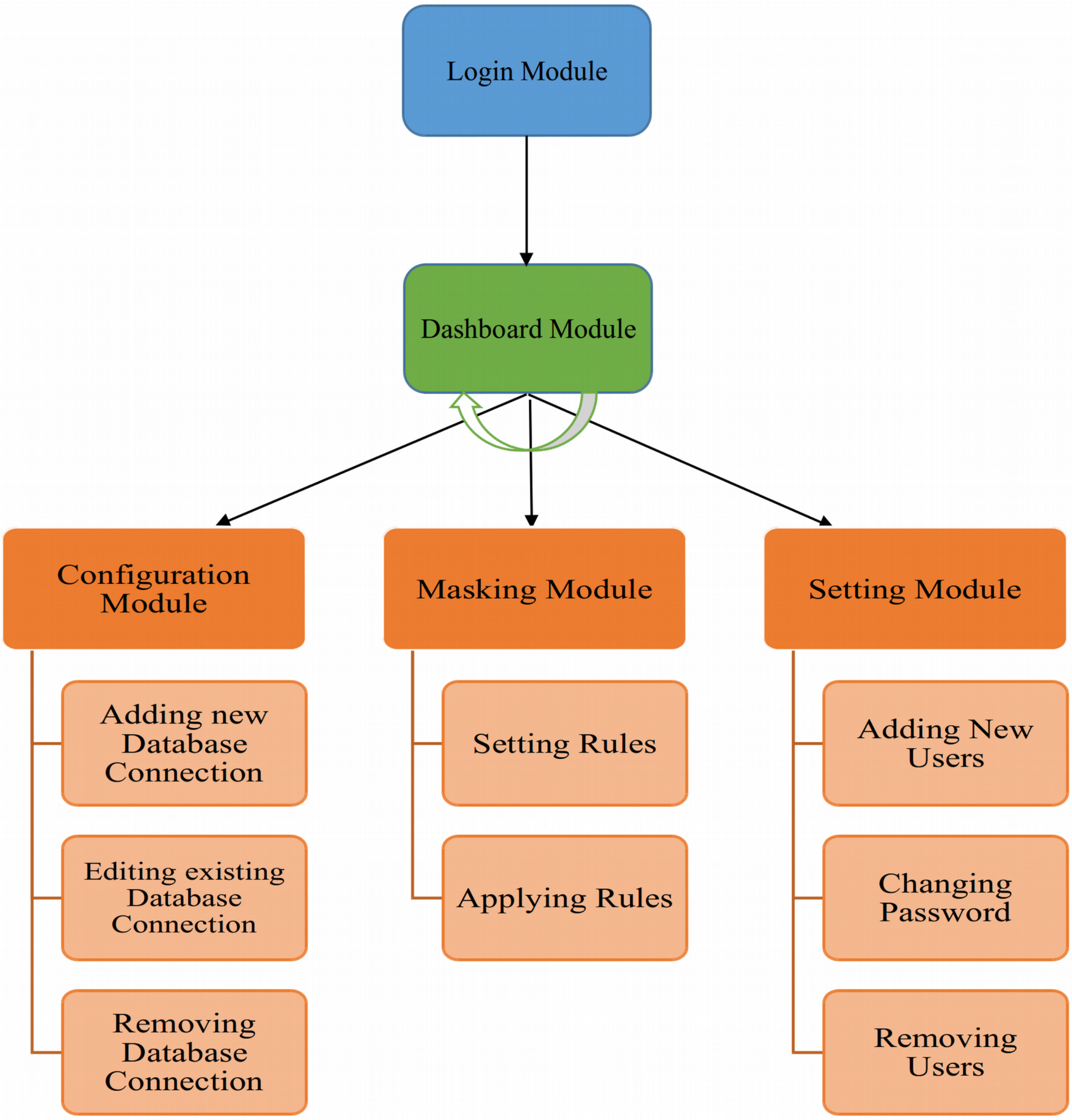


Fig.4.15 Detailed Design

Login Module:

In this module the user get authenticated based on one's user name and password. If it's first time login, the preset username and password is "admin" and "password" so that admin can have first time access to all other module and can edit username and password. If authentication pass successfully, he/she directed to next module (dashboard module).

Dashboard Module:

This is the main module after login module where the admin can select any of three sub-modules (Configuration, Data Masking, Setting). Also at any time the admin can switch between modules.

Configuration Module:

In configuration module the admin can:

- **Add new database connection:**

  Admin enter the details of new database (host address, port, username, password) and the test the connection if every details is correct then only the connection can be saved where later to be used in Data Masking module.

- **Edit existing database connection:**

  Admin can edit the connection that has been saved earlier. The editing can be used to change if any details of existing connection is need to be changed.

- **Remove database connection:**

  Admin delete the saved database connection.

Data Masking Module:

In data masking module the admin can:

- **Set the masking rules:**

  For each connection made earlier the admin get suggested a list of type of masking for each column for all the tables in database schema based on the tables description where he/she chooses one for each.

- **Apply Masking:**

  By applying mask, the rules set earlier will be applied on the schema tables and the “masked” values will be saved on database.

Setting Module: This module is for setting the application level setting as below:

- **Adding new user:**

  Here the admin can add new username/password for login into application which is used in case there arr more than one person doing admin job.

- **Changing password:**

  Admin can change the password for any user including user “admin”.

- **Removing user:**

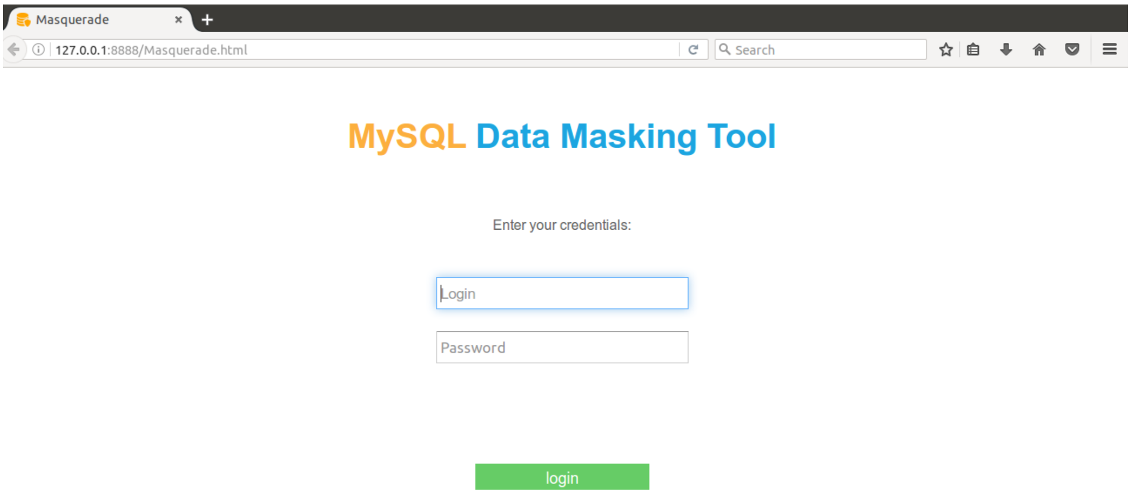
  Any previously added username can be deleted to revoke access to application.

CHAPTER 5

RESULT & DISCUSSIONS

5.1 Admin side Front End

Login page as per Login module explained in part 4.6.2.

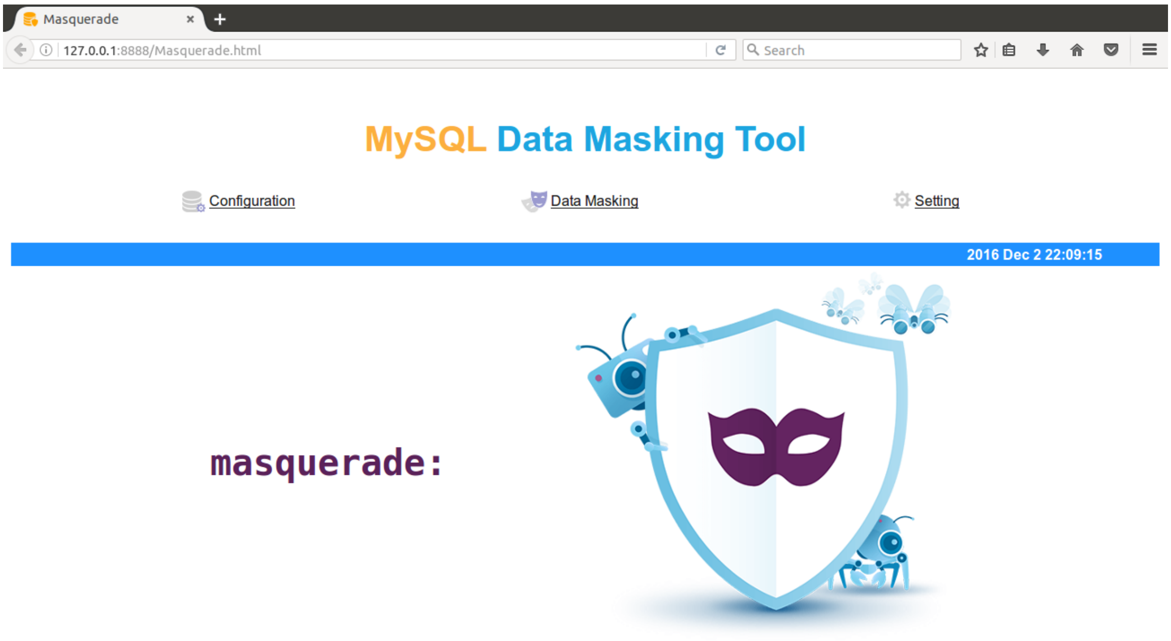


MySQL Data Masking Tool

Enter your credentials:

Fig.5.1 Login Page

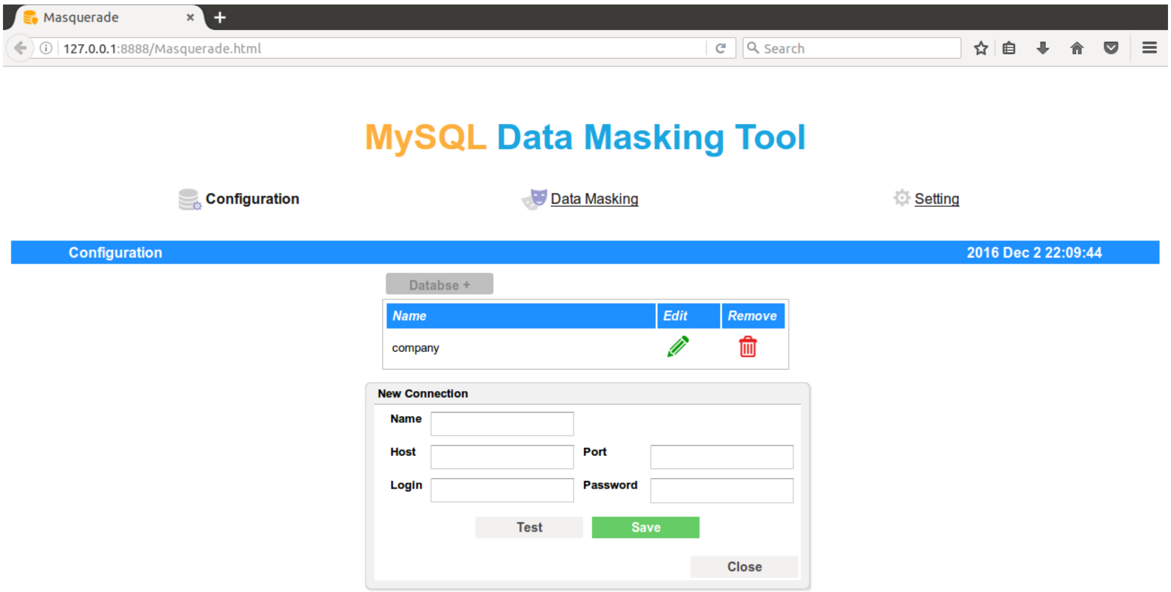
Dashboard page as per Dashboard module explained in part 4.6.2.



MySQL Data Masking Tool

Fig.5.2 Dashboard Page

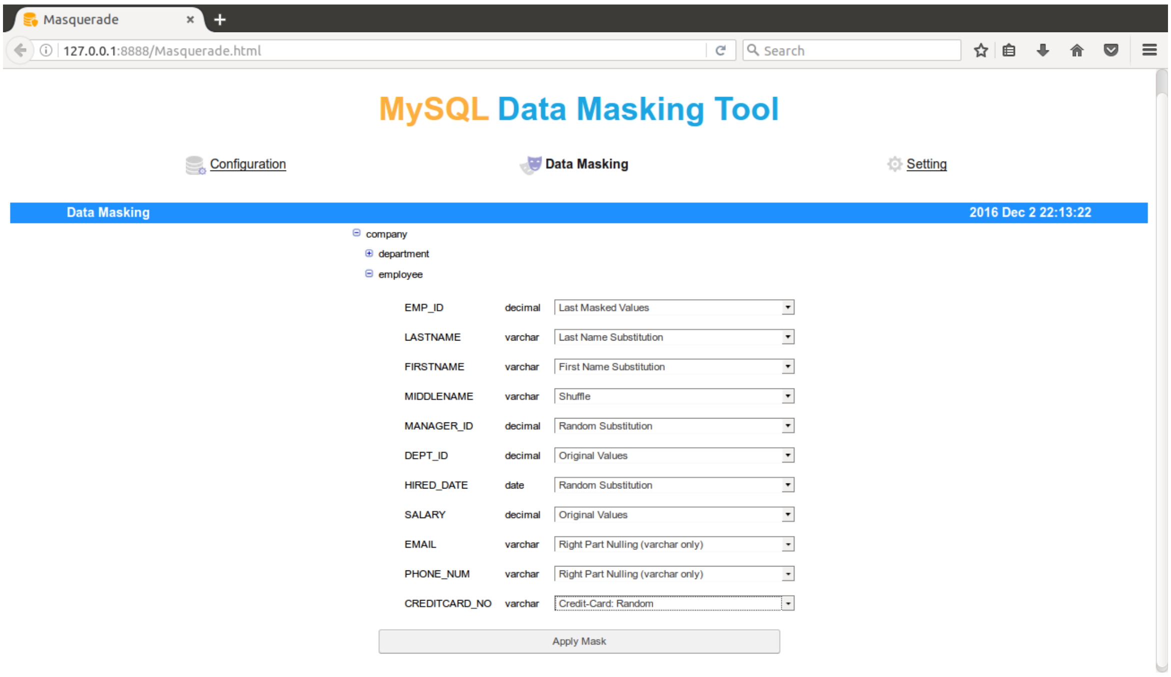
Configuration page as per Configuration module explained in part 4.6.2.



Name	Edit	Remove
company	✎	🗑

Fig.5.3 Configuration Page

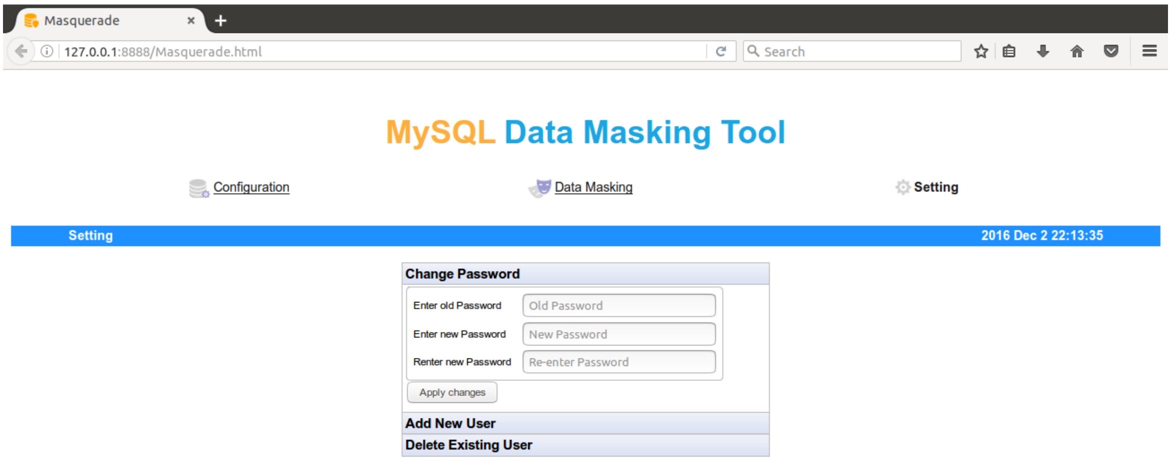
Data Masking page as per Data Masking module explained in part 4.6.2.



EMP_ID	decimal	Last Masked Values
LASTNAME	varchar	Last Name Substitution
FIRSTNAME	varchar	First Name Substitution
MIDDLENAME	varchar	Shuffle
MANAGER_ID	decimal	Random Substitution
DEPT_ID	decimal	Original Values
HIRED_DATE	date	Random Substitution
SALARY	decimal	Original Values
EMAIL	varchar	Right Part Nulling (varchar only)
PHONE_NUM	varchar	Right Part Nulling (varchar only)
CREDITCARD_NO	varchar	Credit-Card: Random

Fig.5.4 Data Masking Page

Setting page as per Setting module explained in part 4.6.2.



MySQL Data Masking Tool

Setting 2016 Dec 2 22:13:35

Change Password

Enter old Password: Old Password

Enter new Password: New Password

Renter new Password: Re-enter Password

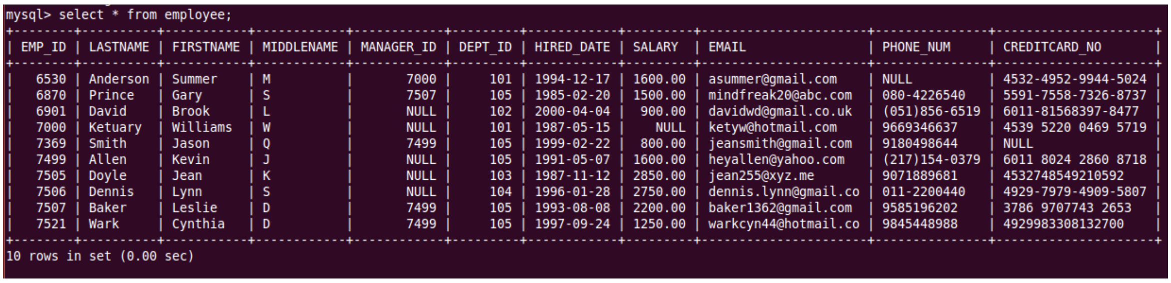
Add New User

Delete Existing User

Fig.5.5 Setting Page

5.2. Querying By Admin Credentials

Admin will see the original data as shown below

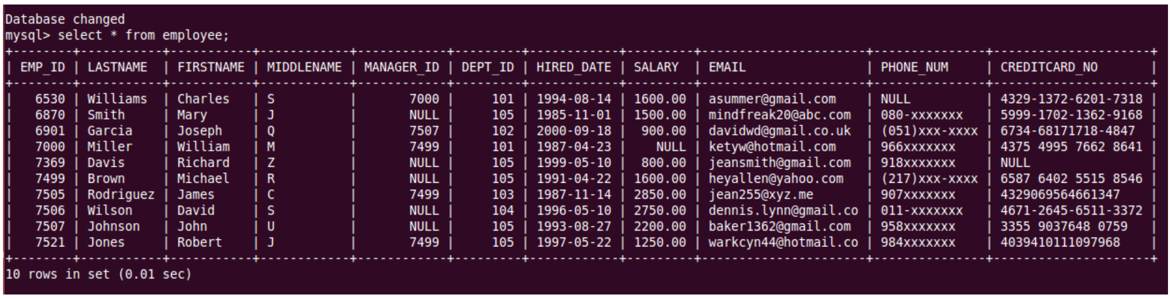


```
mysql> select * from employee;
+--------+----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
| EMP_ID | LASTNAME | FIRSTNAME | MIDDLENAME | MANAGER_ID | DEPT_ID | HIRED_DATE | SALARY  | EMAIL                | PHONE_NUM     | CREDITCARD_NO       |
+--------+----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
|   6530 | Anderson | Summer    | M          |       7000 |     101 | 1994-12-17 | 1600.00 | asummer@gmail.com    | NULL          | 4532-4952-9944-5024 |
|   6870 | Prince   | Gary      | S          |       7507 |     105 | 1985-02-20 | 1500.00 | mindfreak20@abc.com  | 080-4226540   | 5591-7558-7326-8737 |
|   6901 | David    | Brook     | L          |       NULL |     102 | 2000-04-04 |  900.00 | davidwd@gmail.co.uk  | (051)856-6519 | 6011-81568397-8477  |
|   7000 | Ketuary  | Williams  | W          |       NULL |     101 | 1987-05-15 |    NULL | ketyw@hotmail.com    | 9669346637    | 4539 5220 0469 5719 |
|   7369 | Smith    | Jason     | Q          |       7499 |     105 | 1999-02-22 |  800.00 | jeansmith@gmail.com  | 9180498644    | NULL                |
|   7499 | Allen    | Kevin     | J          |       NULL |     105 | 1991-05-07 | 1600.00 | heyallen@yahoo.com   | (217)154-0379 | 6011 8024 2860 8718 |
|   7505 | Doyle    | Jean      | K          |       NULL |     103 | 1987-11-12 | 2850.00 | jean255@xyz.me       | 9071889681    | 4532748549210592    |
|   7506 | Dennis   | Lynn      | S          |       NULL |     104 | 1996-01-28 | 2750.00 | dennis.lynn@gmail.co | 011-2200440   | 4929-7979-4909-5807 |
|   7507 | Baker    | Leslie    | D          |       7499 |     105 | 1993-08-08 | 2200.00 | baker1362@gmail.com  | 9585196202    | 3786 9707743 2653   |
|   7521 | Wark     | Cynthia   | D          |       7499 |     105 | 1997-09-24 | 1250.00 | warkcyn44@hotmail.co | 9845448988    | 4929983308132700    |
+--------+----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
10 rows in set (0.00 sec)
```

Fig.5.6 Querying By Admin Credentials

5.3. Querying By Client/User Credentials

Client will see the masked data as per rules specified by admin in our data masking application.



```
Database changed
mysql> select * from employee;
+--------+-----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
| EMP_ID | LASTNAME  | FIRSTNAME | MIDDLENAME | MANAGER_ID | DEPT_ID | HIRED_DATE | SALARY  | EMAIL                | PHONE_NUM     | CREDITCARD_NO       |
+--------+-----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
|   6530 | Williams  | Charles   | S          |       7000 |     101 | 1994-08-14 | 1600.00 | asummer@gmail.com    | NULL          | 4329-1372-6201-7318 |
|   6870 | Smith     | Mary      | J          |       NULL |     105 | 1985-11-01 | 1500.00 | mindfreak20@abc.com  | 080-xxxxxxx   | 5999-1702-1362-9168 |
|   6901 | Garcia    | Joseph    | Q          |       7507 |     102 | 2000-09-18 |  900.00 | davidwd@gmail.co.uk  | (051)xxx-xxxx | 6734-68171718-4847  |
|   7000 | Miller    | William   | M          |       7499 |     101 | 1987-04-23 |    NULL | ketyw@hotmail.com    | 966xxxxxxx    | 4375 4995 7662 8641 |
|   7369 | Davis     | Richard   | Z          |       NULL |     105 | 1999-05-10 |  800.00 | jeansmith@gmail.com  | 918xxxxxxx    | NULL                |
|   7499 | Brown     | Michael   | R          |       NULL |     105 | 1991-04-22 | 1600.00 | heyallen@yahoo.com   | (217)xxx-xxxx | 6587 6402 5515 8546 |
|   7505 | Rodriguez | James     | C          |       7499 |     103 | 1987-11-14 | 2850.00 | jean255@xyz.me       | 907xxxxxxx    | 4329069564661347    |
|   7506 | Wilson    | David     | S          |       NULL |     104 | 1996-05-10 | 2750.00 | dennis.lynn@gmail.co | 011-xxxxxxx   | 4671-2645-6511-3372 |
|   7507 | Johnson   | John      | U          |       NULL |     105 | 1993-08-27 | 2200.00 | baker1362@gmail.com  | 958xxxxxxx    | 3355 9037648 0759   |
|   7521 | Jones     | Robert    | J          |       7499 |     105 | 1997-05-22 | 1250.00 | warkcyn44@hotmail.co | 984xxxxxxx    | 4039410111097968    |
+--------+-----------+-----------+------------+------------+---------+------------+---------+----------------------+---------------+---------------------+
10 rows in set (0.01 sec)
```

Fig.5.7 Querying By Client/User Credentials

5.4. Test Cases

5.4.1. Unit Test Cases

Table 5.1: Unit Test Cases

Test Case ID	Test case Description	Expected Result	Actual Result
1	Logging in (authentication)	User successfully logged in	PASS
2	Dashboard Module	User can select options to be performed	PASS
3	Configuration Module	User can create, edit, remove database connection	PASS
4	Masking Module	Setting and Applying masking rules should appear	PASS
5	Setting Module	Changing password, adding new user and removing user is performed	PASS

5.4.2. System Test Cases

Table 5.2: System Test Cases

Test Case ID	Test Case Description	Expected Result	Actual Result
1	Logging in as admin Dashboard view	User successfully logged in. User has a dashboard from which he can choose to configure app, mask schema or manipulate settings	SUCCESSFUL
2	Login as admin Dashboard Module Configuration - Add database	User successfully logged in. User chose Configuration module from Dashboard. A new database added.	SUCCESSFUL
3	Login as admin Dashboard Module Configuration– Remove database	User successfully logged in. User chose Configuration module from Dashboard. An existing database removed.	SUCCESSFUL
4	Login as admin Dashboard Module Configuration - Edit database	User successfully logged in. User chose Configuration module from Dashboard. Details of an existing database changed.	SUCCESSFUL

5	Login as admin Dashboard Module Configuration - Add database Masking - Mask the recently added database.	User successfully logged in. User chose Configuration module from Dashboard. A new database added. The tables present in the database with all columns and details appear in Masking tab. The columns are masked with user specified techniques.	SUCCESSFUL
6	Login as admin Dashboard Module Configuration - Add database Masking - mask the recently added database Settings - Change password for admin Login as admin with new password	User successfully logged in. User chose Configuration module from Dashboard. A new database added. The tables present in the database with all columns and details appear in Masking tab. The columns are masked with user specified techniques. Change password in Settings tab. Login as admin with the new password. All previous changes made were effective.	SUCCESSFUL
7	Login as admin Dashboard Module Settings – Add new user Login as new user	User successfully logged in. Settings tab selected. Added a new user and password set for this user. Logged in as new user.	SUCCESSFUL

8	Login as admin Dashboard Module Settings – Add new user Login as new user Login as admin Delete new user	User successfully logged in. Settings tab selected. Added a new user and password set for this user. Logged in as new user. Logged in again as admin. From the Settings tab deleted the new user. Login as new user (created earlier) fails.	SUCCESSFUL

5.4.3 Experimental Result Analysis

Comparison study of various techniques with the replacement method with respect to response time and our results shown random replacement is strongest method of data masking used across all domain which gives maximum confidence for all the customers and data masking will enable to accomplish the following:

- Increases protection against data theft.
- Enforces ‘need to know access’
- Researchers in 2009 found that almost 80 to 90 percent of Fortune 500 companies and government agencies have experienced data theft.
- Reduces restrictions on data use.
- Provides realistic data for testing, development, training, outsourcing, data mining/research, etc. Enables off-site and cross border software development and resource sharing.
- Supports compliance with privacy legislation & policies.
- Data masking demonstrates corporate due diligence regarding compliance with data privacy legislation.
- Improves client confidence.
- Provides a heightened sense of security to clients, employees, and suppliers. This paper will help in analyzing the level of security needed for real.

CHAPTER 6

CONCLUSION

Our application addresses the necessity of data masking in present information age. Comparison study of various techniques of data masking used across all domains which will give maximum protection to all the customers.

Data masking will enable to accomplish the following:

- Increases protection against data theft.
- Provides realistic data for testing, development, training, outsourcing, data mining/research, software development and resource sharing.
- Compliance with privacy legislation & policies.

6.1. Future Scope

- Security, privacy, and identity management will remain at the top of information security spending priorities the incidence of data breaches will continue to rise unless organizations enforce additional measures to protect sensitive data, both in production and non-production environments.
- We can enhance existing rules and make a comparison study to add more features.

REFERENCES

[1] An Oracle White Paper (2013) - Data Masking Best Practices

[2] Informatica (2011) - Best Practices for Dynamic Data Masking and Securing Production Applications and Databases in Real-Time.

[3] Securosis (2012) - Understanding and Selecting Data Masking Solutions: Creating Secure and Useful Data

[4] A Net 2000 Ltd. White Paper (2010) - Data Masking: What You Really Need To Know Before You Begin

[5] Muralidhar, K. and R. Sarathy, and R. Parsa (1996) – describes how to maintaining the Relationship between confidential and Non-Confidential Attributes in Statistical Databases for data masking

[6] Parsa, R.A., K. Muralidhar, and R. Sarathy (1997) -Discuss the general method for Data Perturbation.

[7] Muralidhar, K. and R. Sarathy (2002) - Presented "The Two Step Data Shuffle: A New Masking Procedure," Invited seminar presented to the Census Bureau and the Washington Statistical Society.

[8] Ravi kumar GK, Dr. Justus Rabi, Manjunath TN (2011) - Proposed a uniform architecture for data masking using random replacement.

[9] Lalit Mittal, NIIT White Paper - Data Masking Techniques for Insurance