MEU جامعة الشرق الأوسط
MIDDLE EAST UNIVERSITY
Amman - Jordan

A Novel Framework to Secure Schema for Data Warehouse in Cloud Computing (Force Encryption Schema Solution)

إطار عمل جديد لتأمين مخطط مستودع البيانات في الحوسبة السحابية (حل مخطط التشفير الإجباري)

Prepared by:

Maad Ibrahim Ahmed

Supervised by:

Prof. Hebah H. O. Nasereddin

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of Master's in Cyber Security and Cloud Computing

Department of Computer Science

Faculty of Information Technology

Middle East University

June, 2021

Authorization

I, **Maad Ibrahim Ahmed**, authorize Middle East University to provide an electronic copy of my thesis to libraries, organizations, and institutions concerned in research and scientific studies upon request.

Name: Maad Ibrahim Ahmed

Date: 07/06/2021

Signature:

Examination Committee Decision

This is to certify that the thesis entitled " A novel framework to secure schema for Data Warehouse in cloud computing (Force Encryption Schema Solution) " Was successfully defended and approved on 07/06/2021.

Examination Committee Members:

Prof. Hebah H . O. Nasereddin (Supervisor) ………………………

Middle East University

Dr. Ahmed Al-Hmouz (Internal Examiner/Chairman)………………………..

Middle East University

Dr. Bassam Al-shargabi (Internal Examiner)………………………......

Middle East University

Dr. Evon Abo-Taieh (External Examiner) ………………………

The University of Jordan

Acknowledgment

Thanks to his almighty God for his blessing and support for giving the opportunity of this achievement.

I would like to express my thanks and gratitude to my supervisor, **Prof. Hebah H. O. Nasereddin** for her continuous scientific support, suggestions, guidance.

also thanks all of my doctors from the Department of Information Technology who were generous in their teaching and academic services, the dean **Dr. Abdelrhman AbuArqoub, Prof. Mohammed Al-Husainy, Dr. Hisham Abu Saaymeh, Dr. Mudhafar Al-Jarrah, Dr. Bassam Al-Shargabi, Dr. Ahmed Tabieh,** and the Middle East University with all its staff.

The Researcher

Dedication

I dedicate this work to dear father, dear mother,my beloved wife ,my sons (**Taha & Ameen**), **my sisters**, **my close friend Dr. Mohammed Abdullah Bahgat**, all the family and friends who supported me during my studies**.**

The Researcher

Table of Contents

Subject **Page**

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning
ANSI	The American National Standards Institute
BI	Business intelligence
CRT	Chinese remainder theorem
CSP	Cloud Service Provider
DBMS	Database management system
DLA	Data lake architecture
DW	Data Warehouses
DWaaS	Data Warehouse as a Service
DWHA	Data warehouse architecture
EDW	Enterprise data warehouse
ER	Entity–relationship
FK	Foreign key
GS	Galaxy Schema
IaaS	Infrastructure-as-a-Service
NoSQL	Not Only Structured Query Language
ODS	Operational Data Store
OLAP	Online Analytical Processing
PaaS	Platform-as-a-Service
PKs	primary keys
SaaS	Software-as-a-Service
SFS	Snowflake’s schema
SQL	Structured Query Language
SS	Star schema

A Novel Framework to Secure Schema for Data Warehouse in Cloud Computing (Force Encryption Schema Solution).

Prepared by:

Maad Ibrahim Ahmed

Supervised by:

Prof. Hebah H. O. Nasereddin

Abstract

Data warehouses for organizations offers the benefit in vital decisions making as well as in standardizing big data which is formed through different sources, also to keep data secured from external and internal threats, both sensitive and non-sensitive data must be protected. In this thesis, a new encryption mechanism is proposed based on the column names of the data warehouse, as it was implemented on the Replit program to test its validity and the time taken for the operations executed on encryption, decryption, and querying (text, cipher).

The proposed algorithm is coded using C++ programming language using online integrated development environment (Replit), that allows users to write code and build applications using a browser, Replit has various collaborative features such as; a capability for real-time programming and code hosting platform, which helps to deal with CSV files as a cloud DW. The experimental work was carried out on a table that contains several names for the columns through which the various operations were performed and the time for their implementation was calculated. An encryption was made based on a password, mainly verifying it twice, the first time is to determine the process that will take place on the text by referring to a table in which all the letters, numbers and symbols that can be entered within the password are installed, and the second time is to use the password in the process of generating a random key and use it in the encryption process. When the algorithm is executed, the actual time for encryption, decryption and querying is calculated.

Keywords: Big Data, Data Warehouse, Encryption, Decryption, Querying, Cloud DW.

إطار عمل جديد لتأمين مخطط مستودع البيانات في الحوسبة السحابية

(حل مخطط التشفير الإجباري)

إعداد:

معد ابراهيم أحمد

إشراف:

الأستاذة الدكتورة هبه ناصر الدين

الملخّص

توفر مستودعات البيانات للمؤسسات ميزة في اتخاذ القرارات الحيوية وكذلك في توحيد البيانات الضخمة التي يتم تشكيلها من خلال مصادر مختلفة، وكذلك للحفاظ على البيانات آمنة من التهديدات الخارجية والداخلية، يجب حماية البيانات الحساسة وغير الحساسة. في هذه الأطروحة، تم اقتراح آلية تشفير جديدة بناءً على أسماء أعمدة مستودع البيانات، حيث تم تنفيذها على برنامج Replit لاختبار صلاحيتها والوقت المستغرق للعمليات المنفذة على التشفير وفك التشفير والاستعلام (نص، المشفر).

يتم ترميز الخوارزمية المقترحة باستخدام لغة البرمجة C ++ باستخدام بيئة التطوير المتكاملة عبر الإنترنت (Replit)، والتي تتيح للمستخدمين كتابة التعليمات البرمجية وإنشاء التطبيقات باستخدام متصفح، ولدى Replit ميزات تعاونية متنوعة مثل؛ القدرة على البرمجة في الوقت الحقيقي ومنصة استضافة التعليمات البرمجية، مما يساعد على التعامل مع ملفات CSV مثل DW السحابية. تم تنفيذ العمل التجريبي على جدول يحتوي على عدة أسماء للأعمدة التي تم من خلالها تنفيذ العمليات المختلفة وحساب وقت تنفيذها. تم إجراء تشفير بناءً على كلمة مرور، والتحقق منها بشكل أساسي مرتين، المرة الأولى يتم تحديد العملية التي ستتم على النص من خلال الرجوع إلى جدول فيه جميع الأحرف والأرقام والرموز التي يمكن إدخالها داخل كلمة المرور والمرة الثانية هي استخدام كلمة المرور في عملية توليد مفتاح عشوائي واستخدامه في عملية التشفير. عند تنفيذ الخوارزمية، يتم حساب الوقت الفعلي للتشفير وفك التشفير والاستعلام.

الكلمات المفتاحية: البيانات الضخمة، مستودع البيانات، التشفير، فك التشفير، الاستعلام، سحابة مستودع البيانات.

CHAPTER ONE

Introduction

CHAPTER ONE
Introduction

1.1 Research Topic

The title of this search thesis proposes a framework to secure data warehouse (DW) in cloud computing, DW is a central station for a collection of information, which originates either from several sources or a single source. DW consists of several forms, including structured, unstructured, and semi-structured data, given that data gets loaded organized, and then processed. Where data scientists and decision-makers use tools intelligently Businesses and Structured Query Language (SQL) clients can access and process the DW.

DW differs from databases in that it works to store historical information during a specific period, and it performs the sorting process based on different and various topics such as products and profits. (Almeida, 2017)

Cloud computing is a service Simmon, (2018) provided by the cloud that depends mainly on pay-as-you-go and provides three main services **Software-as-a-Service (SaaS)**, **Platform-as-a-Service (PaaS)**, **Infrastructure-as-a-Service (IaaS)** as well as providing other services, including Data Warehouse as a Service (DWaaS) where inquiries are made through the web service through analytical inquiries. This technology which is used by top companies like: AWS, Azure and, google faced several security challenges and all were addressed in various ways, (Gandhi & Kumbharana, 2018).

Cloud computing is also published in four models (Public cloud- Private cloud- Hybrid cloud- Community cloud). It provides services to large companies in saving the infrastructure work needed, and the cost to maintain.

cloud computing provides applications and storages that spared the high cost and complexity, in return simply by paying to use and stop services anytime needed.

1.2 Problem Statement

DW may contain massive amounts of organizational data such as financial information, credit card numbers, organization trade secrets, and personal data, thus DW are vulnerable to cyber-attack. DW must ensure that sensitive data is protected and cannot be discovered by unauthorized persons. especially when data are consolidated into one big repository and become an easy target for malicious outside or inside attackers.

The main problem that DW faces in terms of security is the technical changes in the development of DW that is affected by the security control. Moreover, there is huge development in hacking methodologies that is growing constantly and is increasing. Therefore, it is necessary to search for new, unconventional ways to protect data that are suitable to prevent hackers from knowing the names of the columns of the DW.

Another key point is that many of the previous studies focused only on securing DW on premises or on the cloud, but the problem in my view is that the encryption process takes place on the scheme in DW and not on the data, because of its large size in comparison to the column name as it takes time to encrypt and decrypt and to prevent a hacker from knowing what important information it may contain. When the problem was solved by encrypting the names of the columns in the DW in a new way, in which I relied mainly on the password and performed several operations to extract the name of the encrypted column.

1.3 Significance

Data is the main factor in the systems; any breach or leakage of data causes crises for organizations especially if the data is sensitive. So, securing sensitive data is important for an organization.

Lots of studies focuses on securing DW, but my research proposes a different way to secure DW through encrypted column names for the fact and dimension table. The importance of change techniques in securing data is to keep the sensitive data out of competitors' hands and to face any potential threats.

Indications refer to an increase in data breach techniques. Therefore, there is an urgent need to change data prevention techniques to keep our data secure. To do this I proposed an enhanced encryption model the column name for tables while uploading data to the cloud and store it in the cloud.

1.4 Objectives

The main objective is to implement and apply security solutions for DW in cloud computing which are different to ensure that sensitive data be safe from unauthorized persons. Because once data is collected in one, big DW, it becomes an easy target for outside attackers.

A new algorithm was designed to achieve encryption of table names, then implemented the proposed algorithm on the cloud based on the input data, analyzed it, and finally showed the results.

1.5 Research Questions

1. What is the main idea of encryption column names in a DW?

2. How does the key gets generated and used in the encryption and decryption process?

3. What conclusions are based on the results extracted through encryption, decryption, and query time (text and cipher)?

4. What are the limitations of the proposed solution and how can it be improved it in the future?

1.6 Delimitations

There are several obvious big data security issues which needs more concentration to cover these issues and it should be considered in any related study. One of these issues is the size of data storage and location of the data stored, so having the need to consider distributed data, and Not Only Structured Query Language (NoSQL) concepts when being concerned about security, and there are many new tools for DW to increase the performance of processing and could be difficult to secure these tools.

Updating security is one of the challenges that could be faced in any DW environment to mitigate loss and exposure data risk. Even security tools need to be under control and to be monitored to measure the effectiveness of these tools. These days, a new technology is being discovered very often and is being used by the attackers to breach data, so any security solution should be multi-layer and changeable to face the new attacking techniques.

CHAPTER TWO

Theoretical Background and Literature Review

CHAPTER TWO
Theoretical Background and Literature Review

2.1 Data Warehouse

A DW is a "subject-oriented, integrated, nonvolatile, and time-variant collection of data in support of management's decisions" (Konda & More, 2015) it is used to connect and analyze business data from different sources. The DW is the core of the Business intelligence (BI) system which is built for data analysis and reporting.

DW’s platform is different from databases because DW keeps historical information, making it easier for businesses to analyze data over a specific period of time to help them make decisions easier. DW platforms collects, and sorts of data based on different variables, such as customers, products, or business activities.

2.1.1 Concept of DW

The concept of DW is defined by Almeida, (2017)DW acts as a central repository of information that is collected from many sources. Data is collected from transactional systems and relational databases to the DW.

DW consists of structured, semi-structured, and unstructured data. This data is loaded, processed, and consumed by businesses daily. Users for DW such as scientists, business analyst, and decision-makers use (BI) tools, Structured Query Language (SQL), and spreadsheets to access data processed in the DW. The results of DW are the main source of information for report generation, information analysis, and presentation, and to be useful for business through reports, portals, and dashboards. Building DW used to be difficult. Many interested in DW found it to be costly, and time-consuming. Over the years, it has become risky, and this produced a need to be changed.

The DW view includes fact tables and dimension tables. It represents the information that is stored in DW, including some calculations for totals and counts, as well as information regarding the source, date, and time of origin added to provide historical context.

2.1.2 Types of Data Warehouse

According to Simões, (2010)

1. Enterprise data warehouse (EDW): It provides decision support services across the enterprise.
2. Operational Data Store (ODS): DW is refreshed in real-time.
3. Data Mart: is a subset of the DW designed for a particular line of business, such as sales, and finance.

2.1.3 Characteristics of Data Warehouse

A DW is subject-oriented as it offers information related to the theme instead of companies' ongoing operations. The data also needs to be stored in the DW in a common, and acceptable way.

A DW is relatively extensive for the time horizon compared with other operational systems, and its non-volatile which means previous data cannot be erased when new information is entered into it.(Almeida, 2017)

2.2 Data Warehouse architecture

DW needs to rely on architectures, data warehouse architecture (DWHA) is developed synchronously along with progress to serve DW(Yang et al., 2019) A variety of different DWHA is proposed to fit different requirements. There is no classification that categorizes all DWHA.

The latest DWHA classification is proposed by (Blažiü et al., n.d.2017), but some DW architecture, such as Virtual DWHA, Big DWHA and, DW with Data Lake architecture (DLA), are still excluded. These DWHA have advantages to solve some problems, which would be more time-consuming and less efficient.

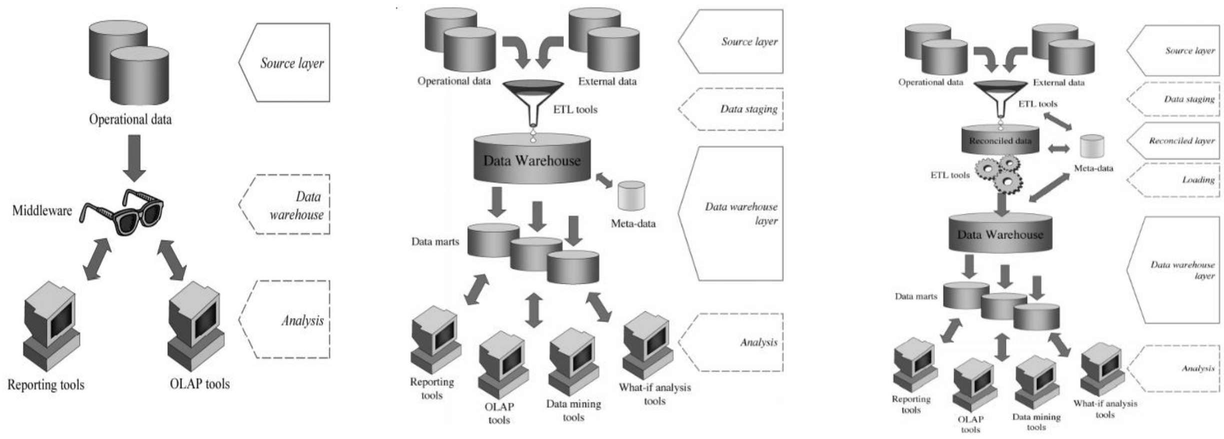
2.2.1 DWHA Classifications

According to Blažiü et al., n.d(2017), As in Figure (1) The first classification includes single-layer, two-layer, and three-layer architectures that depend on the number of layers used by the architecture. The second classification consists of independent architecture, Bus, Hub-and spoke, Centralized and, Federated architectures based on components and relationships.

The mechanism of the single layer (real-time layer) is the data sets that are stored one time only. An operating system and a DW system share identical data that are stored in the same place.

The two-layer architecture adds a derived data layer to enable the operational and analytical requirements. This approach addresses the single-layer architecture problems, but it has limitations, which as low performance in large volumes of data.

The three-layer architecture consists of a real-time data layer, a reconciled data layer, and a derived data layer, which adds a reconciled layer as the new layer compared with the two-layer architecture. This new layer materializes the integrated and cleansed data.



Single-layer architecture for a Two-layer architecture for a Three-layer
architecture for a data warehouse system data warehouse system
data warehouse system

Fig.1: Data Warehouse architecture (Malinowski & Zimányi, 2008)

2.2.2 DWHA Overview

According to Yang et al., (2019), the models of DWHA as the following:

- **The Hub-and-Spoke DWHA**: Bill Inmon proposes the Hub-and-Spoke DWHA, it is suitable for traditional relational database tools. it is used to develop an enterprise wide DW. It is a top-down approach.
- **The Data Mart Bus DWHA**: Ralph Kimball supports the Data Mart Bus using dimensional modeling, it is a bottom-up approach to develop a DW.
- **The Centralized DHWA**: The Centralized DHWA collects heterogeneous data into the functional information system, which enables the sources to be used and stores the presentation format for any requests from users.
- **The Independent DWHA**: the independent DHWA derives from a broader scope instead of Independent Data Marts. The Independent Data Marts consists of physically/logically separated and irrelevant DW. while the Independent DHWA may contain hybrid and less-coupling components including DW.

- **The Federated DWHA**: The Federated DWHA is based on the bottom-up implementation, the federated DW model can be acted as the added data staging layer that enables the easier implementation of analytical solutions to hide the complexities of operational data sources.
- **The Virtual DWHA**: The virtual DWHA manipulates and analyses operational data sources with limited data integration functions and summary views of the DW depending on the complexity of requirements.
- **The Distributed DWHA**: The distributed DWHA includes DW working in parallel or built on multi-node cloud computing platforms, in which data are pre-processed and physically distributed in a Database management system (DBMS) with a predefined schema. This architecture fragments the warehouse schema using partitioning algorithms and allocates the generated fragments over compute nodes using algorithms. To gain the availability of data, and the high performance of the system, the generated fragments may be duplicated and saved in different compute nodes.
- **The Big DWHA**: is built on a big datum platform such as Hadoop, in which the distributed file system and, other mechanisms like MapReduce are applied to store data, respectively. This DW can address big data issues such as petabyte level for reporting and ad-hoc analyses by using SQL-like language (Hive)(Institute of Electrical and Electronics Engineers et al., n.d 2010.), which is easily executed by users with SQL experience. It can manage unstructured data provided by a logical DW (Yang et al., 2019)This logical DW does not need to strictly pre-process data with a predefined schema.
- **The Data Lake Architecture**: The DLA can be leveraged directly as a centralized raw data repository providing information for further analyses. The DLA and DW

can also cooperate to address the big data issues and achieve data analysis requirements, presenting an architecture with a DW and DLA. In their architecture, the DW populates the DLA, which means the DW is built first or a company already has a legacy DW then builds a DLA. Another situation is the DW fed by the DLA. The DW obtains data from the DLA, which is suitable for building the DLA than the DW, optimizing the legacy DW data flows, or extending the DWHA to meet intangible requirements.

2.3 Big Data Warehouse

Nowadays, DW is facing massive growth in data size, which is referred to as big data. Many of these data are unstructured like text, audio, video, traditional DW is not designed to scale and handle exponential growth of unstructured data. To address new unstructured big data challenges, DW is extended and merged with state-of-the-art technologies. The big data platforms or concepts (e.g., Hadoop and Hive) are helped to build DW with high-speed processing capacities(Institute of Electrical and Electronics Engineers et al., n.d.2010).

2.4 Data Modeling Techniques

According to Ballard et al., (2012), there are two main data modeling techniques that are relevant in a DW environment, Entity–relationship (ER) modeling and Multidimensional modeling.

- **ER Modeling**: produces a data model of the specific area of interest, using two basic concepts: entities, and the relationships between those entities. The ER model is an abstraction tool because it can be used to understand and simplify the ambiguous data relationships in the business world and, complex systems environments.

- **Multidimensional modeling**: uses three basic concepts: measures facts, and dimensions. Multidimensional modeling is powerful in representing the requirements of the business user in the context of database tables, Multidimensional model focuses on the behavior of the fact table in any schema. In the fact table whole primary keys (PKs) lie as the foreign key (FK) that shows exactly one record for specifying index this behavior also called the star schema (SS).

The structure of an application may encounter intense change. Measurement change, quality change, level change, and property change in the outline plan of DW, the outline advancement challenges are everlasting, and relevant to each database for all intents and purposes conceivable.

2.4.1 Star Schema

According to Iqbal et al, (2020) Extract, transform, load (ETL) process is followed to design a DW, dimension modeling is used that is done on different models, one of which is SS.

The name, **"star**" is shaped like Star (SS) As in Figure (2) consists of dimensions, which are in dimension tables, and a central fact table that carries facts.

Fact tables carry PKs that in dimension tables are added with the name of FKs, which is the means of connectivity between dimension and fact table. Dimension tables have qualitative data and as discussed above with the FK in fact table contains measures that can be summed to analyze or to carry some process.

In a SS, most of the tables are not normalized, only at first normal form. The execution time of the query of SS is fast for large data. SS database has some tables, clear join paths, and Small single-table queries.

SS design is easy to follow and especially, with a joined query which tends only through the fact table. These joints help to boost the performance of schema. Only simple queries are used because all tables are less complex therefore time to run and output results show less time.

2.4.1.1 Properties of a star schema

1. Dimension tables pass the PK which is added with the name of the FK in the fact table.
2. it contains tabled that are distinguished with, mainly fact tables and dimension tables.
3. SS is not in normalized form except the fact table in the schema.
4. Whenever tables are not normalized it obviously due to redundancy memory needs more, more space is required.
5. SS uses all dimension table’s PKs, so the fact tables are less complex than snowflakes schema (SFS).
6. Denormalized means lesser tables and lesser tables mean fewer complex queries. Queries that are used in SS to access data are, “star join queries.”

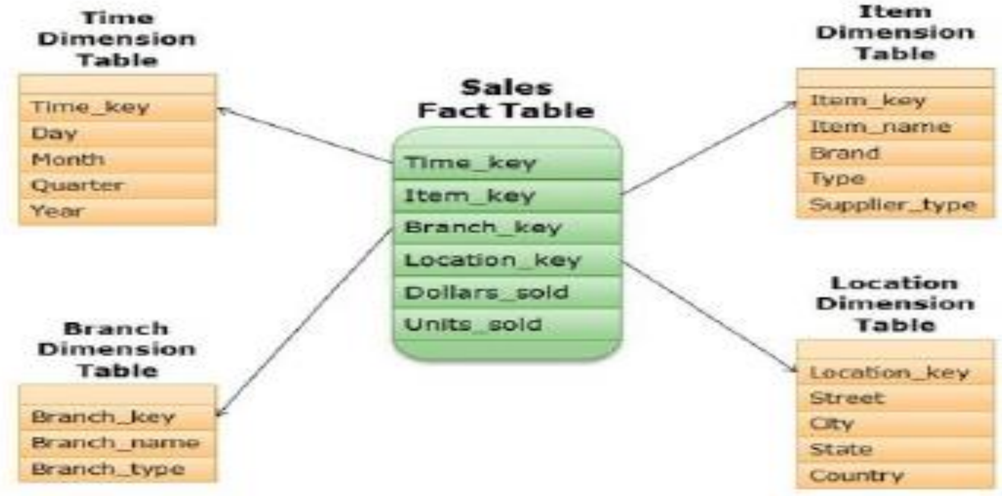


Fig.2: Star Schema example (Iqbal et al., 2020)

2.4.2 Snowflakes Schema

The name (SFS) comes through its resemblance to a snowflake. It is used in the ETL process, and at the same level. Unlike SS, SFS consists of three types of tables, which are, fact tables, dimension tables, and sub-dimension tables As in Figure (3). Working of the fact table is the same as it carries FKs as of dimension tables as FK and contains its other values. The fact or base table connected with dimension tables and dimension tables more time connects to normalize dimension tables through keys.

According to Benjelloun et al., (2018), due to sub-dimensions or the splitting of dimensions into the sub-dimensions concept of hierarchy is introduced in the SFS, and it implements on a simple database system, it works with different devices and administrations, for example, Informatics, Looker or Tableau notwithstanding.

2.4.2.1 Properties of Snowflake Schema

According to Iqbal et al., (2020), the properties of SFS are:

1. It consists of fact tables, dimension tables, and sub-dimension tables.
2. Sub dimension tables are constructed by splitting or normalizing dimension tables, so, it can be called SFS in a controlled form.
3. The schema is normalized, so data are not redundant, repetitions are not present, and values are atomic.
4. When the redundancy removed from the tables then lesser memory is required, which means a decrease in the storage space required.
5. Snowflake structure is more complex because many tables are arranged when normalizing databases till 3rd normal form.
6. The problem there is, due to increase in several tables, querying the data from these tables is a tough job because linking tables from the sub-dimension of one table to a sub-dimension of another table can be complex.

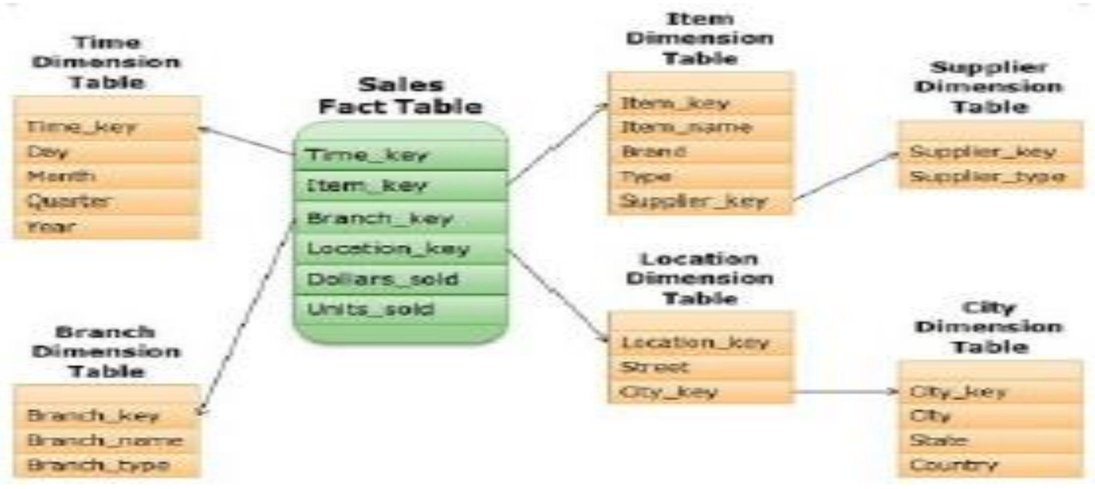


Fig3: Snowflake Schema example(Iqbal et al., 2020)

2.4.2.2 Advantages of snowflakes:

- **Relational**.: it supports all transactions of major platforms like SQL, The American National Standards Institute (ANSI), and ACID compliance. The big benefit is moving from one platform to another platform without a big change.
- **Semi-Structured**: The query performance increases with the built-in functions that help for navigating, destruction, and organizing semi-structured data that are used in a controlled form. Also, with the help of JSON and Avro. Fewer operations performed due to the auto-discovery of schema and storage operations less perform with respect to efficiency and save the efforts.
- **Elastic**: Usually called the portable schema because of scale and resource independent schema.
- **Highly Available**: SFS takes less time to recover and rare chances of failure there for tolerance nodes and cluster less affect the performance. Without down can change the hardware.
- **Durable**: SFS has the properties of cross-region backup. Another kind of backup performed this schema with the help of cloning undrop.in this way, any damage cannot highly affect the schema.
- **Cost-efficient**: SFS is highly cost-effective, due to fewer resources required and compressed data stored.
- **Secure**: The data that is stored in the schema is encrypted end to end, including all traffic over the network and temp files. Additionally, access control also helps the user to protect the data.
- **Performance**: SFS with bigger fact tables which means containing a greater amount of data or the FK in the fact tables can reduce the performance of SS or

can be said the lesser the FK in fact table and more the partitioned is the table the faster and more efficient it works (Benjelloun et al., 2018)When using SS and SFS in a similar environment:

1. When dimension tables are bigger it is good to use SFS because it is in normalized form the dimension tables are partitioned and hence it reduces the size leading to less memory consumption.
2. Query processing time increases when using a SS with large data.

2.4.3 Star Vs Snowflakes Schema

According to Iqbal et al., (2020)

- **Query Complexity**: It is evident that SS and SFS both perform differently due to their nature or properties. When it comes to query complexity, the SFS is more complex for querying. SS has a lesser number of tables because it is not in normalized form, which makes it easier together data from different tables, and in an SFS tables are normalized resulting in a larger number of tables and making it much more complex to query data. Complex joins are sometimes very difficult to handle and execute because several tables sometimes are far greater.
- **Execution time:** There is a difference between their execution time. SS carries in it also the redundant data, so traversing through those causes a bit of time delay too, table size is also bigger because tables not split so traversal through the bigger table is a bit expensive. SFS is normalized tables are split carrying mostly relative data, and no redundancy is there thus making the queries execute faster.
- **Effect on the size of DW:** Normalization is again a key role in discussing the size of DW size. Star being the one having Denormalized dimensions always occupy larger space because it has a lot of redundant data hence, requiring space and

memory and SFS being completely normalized require less space and memory than SS.

- **Results:** When we apply SS, execute the query, and have been noted results of SFS have some difference from SS.

Query Optimization: SS's performance figures gathered from the above-carried example are not much prompting to use them in the construction of our DW. The performance of SS can be improved by adding bitmap indexing. Performance of SFS can also be enhanced but comparing both SFS already have good figures, but the example can be carried for both the schemas.

2.4.4 Galaxy Schema

According to Al-rammahi, (2016), A Galaxy Schema (GS) contains two fact tables that share dimension tables between them As in Figure (4). It is also called Fact Constellation Schema. The schema is viewed as a collection of stars hence the name GS.

2.4.4.1 Characteristics of Galaxy Schema:

1. The dimensions in this schema are separated into separate dimensions based on various levels of hierarchy.
2. For example, if geography has four levels of hierarchy like region, country, state, and city then GS should have four dimensions.
3. Moreover, it is possible to build this type of schema by splitting the one-SS into more SSs.
4. The dimensions are large in this schema which is needed to build based on the levels of hierarchy.
5. This schema is helpful for aggregating fact tables for a better understanding.

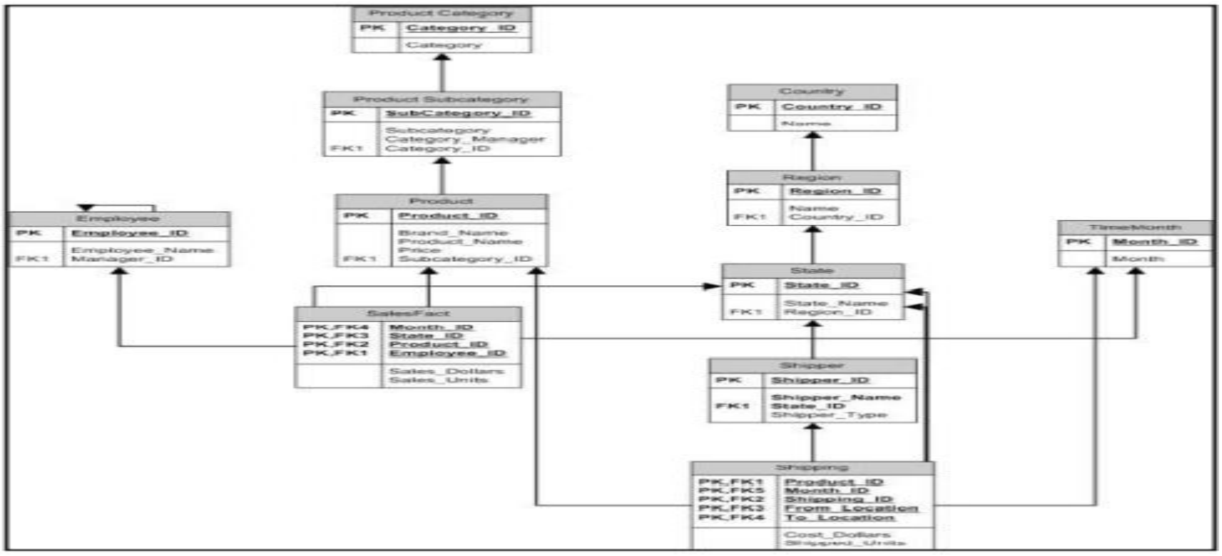


Fig.4: Galaxy Schema example (Al-rammahi, 2016)

2.5 Cloud Computing

According to Simmon, (2018), NIST Cloud Computing Definition, "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be provided with minimal effort or interaction by the service provider. It also contains several main characteristics include on-demand self-service, broad network access, resource pooling, rapid elasticity, measured service. As well, it was divided into three main services and four deployment models.

2.5.1 Cloud Computing Service Model

According to Simmon, (2018).

- **Software as a Service (SaaS):** It is consumer use of applications provided by customers without controlling the cloud infrastructure, for example (email, office365), Except for user-specific settings.
- **Platform as a Service (PaaS):** The user's ability to publish applications that have been created using different programming languages, as he cannot manage or

control the cloud infrastructure such as operating systems or servers as it can be an environment hosted for the application. Where this service makes sense if you are a developer, for example (Google App Engine).

- **Infrastructure as a Service (IaaS):** Providing storage, networks, and basic resources to the consumer, as it can install various operating systems and applications, which gives the user more flexibility using the cloud.

2.5.2 Cloud Computing Deployment Models

According to Simmon, (2018).

- **Private cloud:** This cloud is used by one organization that includes several consumers, it may be owned by the company or managed by a third party, usually the costs are high for this type of cloud and the problems of protecting it from new threats.
- **Public cloud:** This cloud provides the possibility of use by the consumers and may be managed by a governmental or commercial institution or a combination, the safety standards in this cloud differ from a private cloud, as it can be accessed by anyone.
- **Community cloud**: It is provided by a specific community of organizations with similar orientations and may be owned by one or more organizations or a third party.
- **Hybrid cloud:** This cloud consists of several cloud infrastructures (private, community, or public), but they are linked to a specific standard that allows data transfer between them.

2.5.3 Comparison of traditional and cloud DW

Cloud DW provides SaaS. A traditional DW requires lots of time to configure the infrastructure. It also takes a lot of time to optimize and manage the system. A cloud DW be designed to take the advantage of a larger number of users and applications. The benefit of cloud DW over traditional DW is that it can be used easily. Using the cloud, data can be scaled up or down instantly without any issue, traditional DW data cannot be scaled up and down instantly. It is quick and easy to get a DW up and running in the cloud whereas, deploying a traditional DW takes a long time.

Cloud DW provides cost benefits. Using cloud DW there is no hardware, server rooms, IT-related staffing issues, or operational expenses to maintain your DW. Cloud DW reduces the cost, and complexity of managing on-premises systems.

2.6 Literature Review

Existing research provides solutions for different security issues related to DW, According to study in Arora & Gosain, (2020), authors have proposed an enhanced encryption model for DW, where column names has been encrypted with help of keys from a secure host, their proposal has majored components which are: user application, secure application layer, secure host, DBMS and encrypted columned- encrypted databases. As for my thesis, it depends mainly on the password to choose the type of operation that takes place on the name of the column, as well as using the CSV file, entering data for the names of columns, and performing the encryption and decryption process on the online (Replit) program.

In this study algorithm mechanism is once initiated table through the application interface, the user wrote table name with column names, then certain column names are encrypted after removing pretext and post text which distinguishes between column names must be encrypted and other column names and done through a key generated by a secure application host. In case of adding a new column to existing tables, the existing ‘key’ is required to be fetched from the secure host by a secure application layer to encrypt desired column name, to avoid re-encryption of column names during data retrieval.

According to a study in Pacheco & Mar, (2018), it is based on simple privacy homomorphism, their Proposition for securing DW in the Cloud is to encrypt data stocked in the DW, Encryption function $\phi(x) = [x \bmod p, x \bmod q]$, the ciphertext $xp = x \bmod p$ and the ciphertext $xq = x \bmod q$ Sent to the cloud provider with modulo m.

p and q are the prime numbers that should be kept secret at the owner. The data stored in the cloud has been processed uniformly. So, the cloud provider cannot decrypt the data with the modulo m because it is hard to factor. So, the data is stored securely in the cloud.

The homomorphic characteristic of the modular arithmetic query, using arithmetic operations such that $\{+, -, x\}$ made in the cloud in a ciphertext without decryption. After processing the query in the cloud, the provider sends the result to the owner in a ciphertext. The owner of data decrypts the data with the two secret numbers p and q and the two chunks of encrypted data received from the cloud using the Chinese remainder theorem (CRT).

This solution is good regarding storage overhead and time complexity. Besides, the advantage of querying addition, subtraction, and multiplication in a ciphertext is very important in the case of a DW because the nature of its Online Analytical Processing (OLAP) query requires a massive volume of data, but it can be broken by the cloud provider because it has the two chunks of data, and the secret modulo m. It can infer the two chunks of data and get the two secret parameters p and q.

Various techniques to optimize query processing time over encrypted data have been proposed, according to Al-Saraireh, (2017)authors Develop appropriate solutions for different types of attributes, where the choice of symmetric or asymmetric encryption is determined based on the classification of information based on the attributes.

Attributes supporting aggregation functions may be encrypted using Homomorphic Encryption. Homomorphic encryption makes operations such as querying and searching on encrypted data possible without decryption of data. Attributes were the order of data needed to be maintained.

They proposed a mechanism that lets the clients write their query by using an application program or SQL editor; then the DBMS is responsible to validate and manipulate the user requests to access data. In the proposed framework a new layer is used to determine whether the query has sensitive data or not based on the metadata. If a query has sensitive data two new functions are used; encryption/decryption function and hash map function to retrieve the sensitive data from the database. This research has presented an enhancement to the database encryption approach to achieve better response time for the execution of SQL queries. Therefore, the speed of encryption, decryption and querying in my thesis is much faster because I rely on encrypting column names in the DW only and not encrypting fragmented data according to importance.

According to Ali & Afzal, (2017), Authors have proposed a security solution by dividing the process into three stages, first determine critical data, then determine the importance of data, and divide it into categories which include very critical, inactive, and duplicate data. Very critical data has the highest priority and duplicate data the lowest priority.

Once the data is categorized and separated it is important to ensure that the end-users have access to the data. The end-users should be able to access the very critical and critical data as well as the inactive data. In this case, the time used for the encryption and decryption process will be reduced, and the query performance will be more.

Authors in Moghadam et al., (2017) propose S4 as a new schema based on secret sharing for enforcing privacy in Cloud DW. The idea is to store secrets at one single Cloud Service Provider (CSP) instead of sharing secrets with n CSP's. The privacy in S4 relies on the fact that k-1 splits are stocked in the CSP and the Kth splits necessary for reconstructing the secret are stocked in the owner. They can avoid the problem of collusion, but the processing of the query cannot be done totally in the cloud.

Authors in Divya Shaly & Anbuselvi, (2016), provided a multi-cloud schema that guarantees the availability of data with minimum cost, authors in Attasena et al., (2015)have proposed a new model for sharing DW by secret sharing. by splitting the data into blocks before encrypting it with a random linear equation. However, this approach suffers from the high time complexity of decryption steps, another problem that arises when using this approach is that it cannot resist collusion attacks.

Authors in Khan et al., (2015) , Encryption algorithms have been developed on a large scale in a number of fields to process text documents, video and images, as this hybrid technology provides encryption of data using the Fibonacci series, XOR logic and PN sequences, but the method was developed by using fewer bits during encryption than in the Fibonacci series technology. Analyze it and compare its performance. The researcher used the encryption key and the hash key to be equal on both sides for further authentication. For his research on the other side, the system was complicated by the involvement of more than one type of encryption device, which led to an increase in cost and complexity.

CHAPTER THREE

Proposed Methodology

CHAPTER THREE
Proposed Methodology

3.1 Methodology

This research helps to verify the security of DW over the cloud and modify the last proposal for securing DW which protects DW from analysis of table column name, this reduced any possibility by attackers to focus on interesting columns.

Encryption of column names of dimension and fact tables restrict column access to protect sensitive data. This restriction makes security more reliable and robust by reducing the surface area of the overall security system. In addition, Encryption of column names of tables also eliminates the need for introducing views to filter out columns for imposing access restrictions on the users.

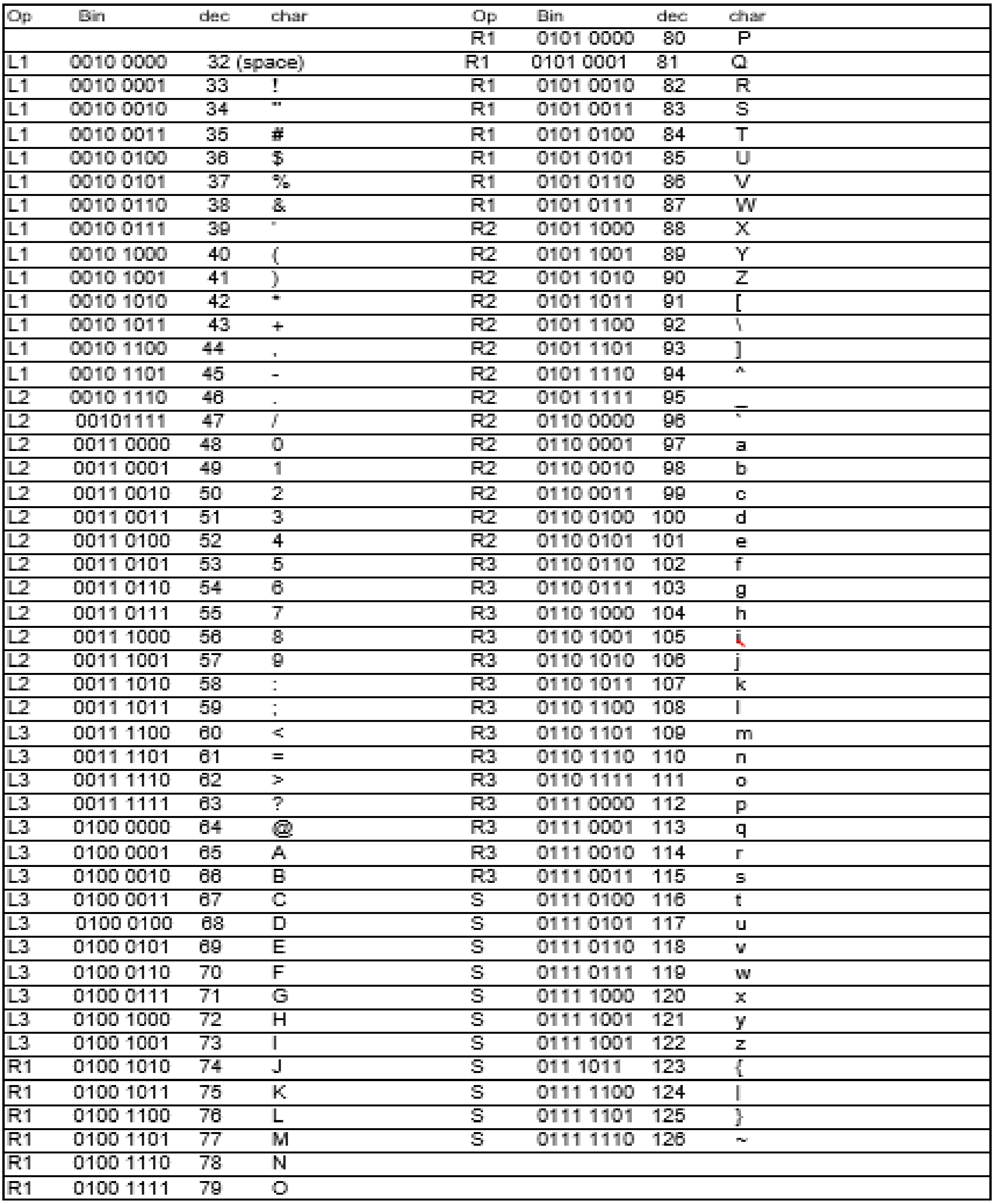
3.1.1 Methodology Steps

1. Designing a solution that provides an encryption method that is used in the DW schema encryption.
2. Implement the algorithm using C++ programming language and DW system.
3. Test the proposed solution.

3.2 Proposed Models

Suggested a column-based encryption solution is a new idea that needs an additional security function for encryption, have been named **FESS (Force Encryption Schema Solution)**. that assumes the database server on the cloud is trusted and propose a lightweight encryptions solution, and our solution will force the security to data during transferring processes to the cloud as shown table 1.

Table 1: Operation for Password



Op	Bin	dec	char	Op	Bin	dec	char
				R1	0101 0000	80	P
L1	0010 0000	32 (space)		R1	0101 0001	81	Q
L1	0010 0001	33	!	R1	0101 0010	82	R
L1	0010 0010	34	"	R1	0101 0011	83	S
L1	0010 0011	35	#	R1	0101 0100	84	T
L1	0010 0100	36	$	R1	0101 0101	85	U
L1	0010 0101	37	%	R1	0101 0110	86	V
L1	0010 0110	38	&	R1	0101 0111	87	W
L1	0010 0111	39	'	R2	0101 1000	88	X
L1	0010 1000	40	(	R2	0101 1001	89	Y
L1	0010 1001	41	)	R2	0101 1010	90	Z
L1	0010 1010	42	*	R2	0101 1011	91	[
L1	0010 1011	43	+	R2	0101 1100	92	\
L1	0010 1100	44	,	R2	0101 1101	93	]
L1	0010 1101	45	-	R2	0101 1110	94	^
L2	0010 1110	46	.	R2	0101 1111	95	_
L2	00101111	47	/	R2	0110 0000	96	`
L2	0011 0000	48	0	R2	0110 0001	97	a
L2	0011 0001	49	1	R2	0110 0010	98	b
L2	0011 0010	50	2	R2	0110 0011	99	c
L2	0011 0011	51	3	R2	0110 0100	100	d
L2	0011 0100	52	4	R2	0110 0101	101	e
L2	0011 0101	53	5	R3	0110 0110	102	f
L2	0011 0110	54	6	R3	0110 0111	103	g
L2	0011 0111	55	7	R3	0110 1000	104	h
L2	0011 1000	56	8	R3	0110 1001	105	i
L2	0011 1001	57	9	R3	0110 1010	106	j
L2	0011 1010	58	:	R3	0110 1011	107	k
L2	0011 1011	59	;	R3	0110 1100	108	l
L3	0011 1100	60	<	R3	0110 1101	109	m
L3	0011 1101	61	=	R3	0110 1110	110	n
L3	0011 1110	62	>	R3	0110 1111	111	o
L3	0011 1111	63	?	R3	0111 0000	112	p
L3	0100 0000	64	@	R3	0111 0001	113	q
L3	0100 0001	65	A	R3	0111 0010	114	r
L3	0100 0010	66	B	R3	0111 0011	115	s
L3	0100 0011	67	C	S	0111 0100	116	t
L3	0100 0100	68	D	S	0111 0101	117	u
L3	0100 0101	69	E	S	0111 0110	118	v
L3	0100 0110	70	F	S	0111 0111	119	w
L3	0100 0111	71	G	S	0111 1000	120	x
L3	0100 1000	72	H	S	0111 1001	121	y
L3	0100 1001	73	I	S	0111 1001	122	z
R1	0100 1010	74	J	S	011 1011	123	{
R1	0100 1011	75	K	S	0111 1100	124	|
R1	0100 1100	76	L	S	0111 1101	125	}
R1	0100 1101	77	M	S	0111 1110	126	~
R1	0100 1110	78	N				
R1	0100 1111	79	O				

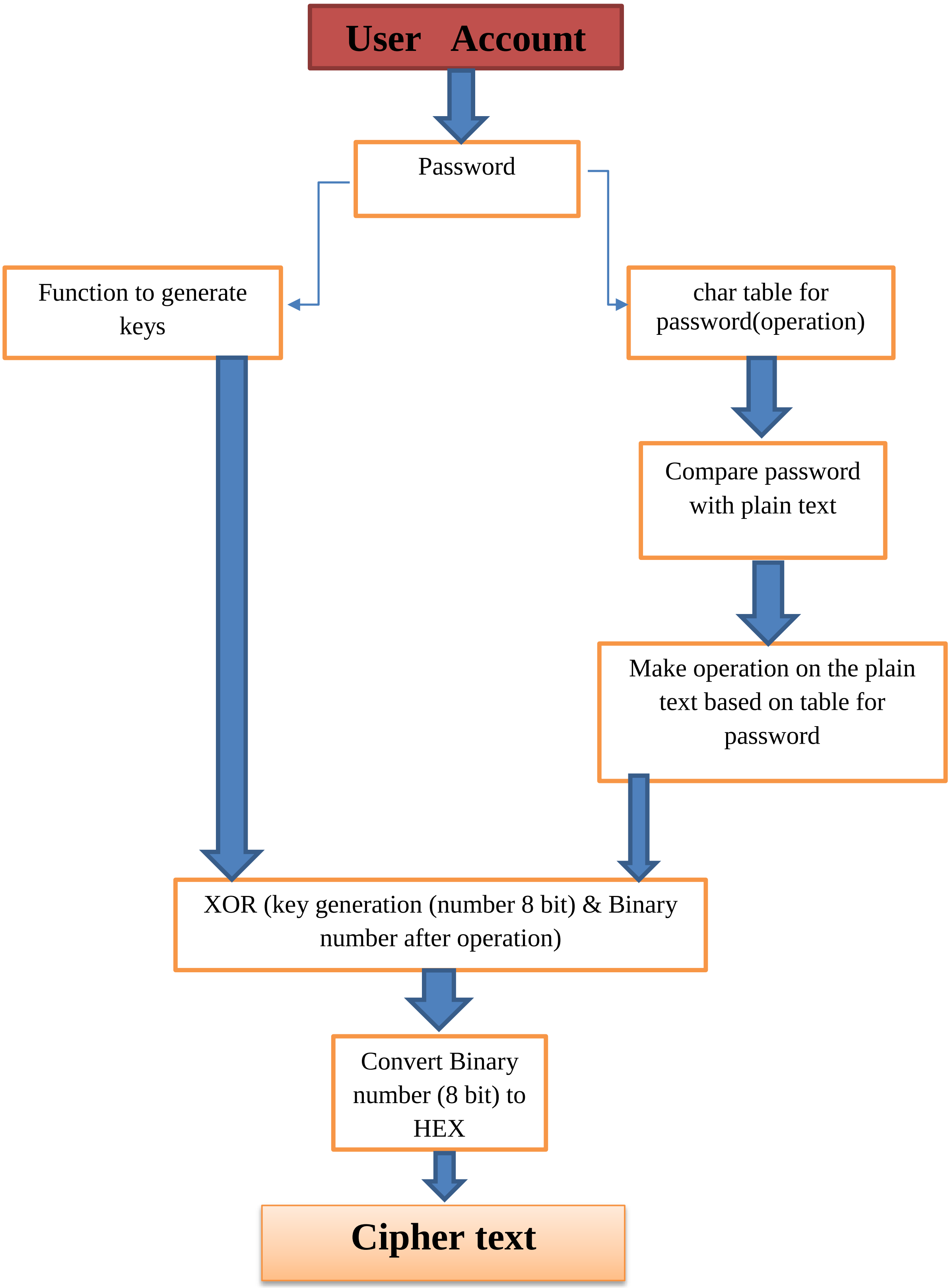
3.3 Description of the Encryption Model

My proposal has multistage to secure DW, In the beginning, the cloud user registers his/her account data from a username and password, The password is used and makes the primary key of the algorithm, and the minimum number of characters for the password is 8 varied characters between upper and lowercase letters, numbers, and symbols, thus the lowest key can be worked So, it is 8 bytes = 64 bits, which means = 2 ^ 64 possible selection of the key size and more probability if the password exceeds 8 bytes, and this gives a strong point to the algorithm, in addition, to use the password as a seed through which one can get random keys by taking a bit of Each letter is 1 byte. Use it to randomly encrypt letters. Each column name encrypts a different key. It will add a little extra time to the encryption process, but it will give strength to the algorithm.

An operations table has been created in which that define a different encryption process for each character that can be entered into the password, as the column name is encrypted with the same character sequence in the password. If it exceeds the characters of the password, the Password is repeated to cover the number of characters desired Encrypting, this method gives a point of strength as the single character is encrypted in several ways each time, in addition to another strong point, which is every change of the password. The encryption method is changed by various operations that are created through the operations table.

After performing the above-mentioned operations, the process of XOR is performed using the random key consisting of 1 byte and finally the process of converting the file from the binary system to the hexadecimal system and the ciphertext is output.

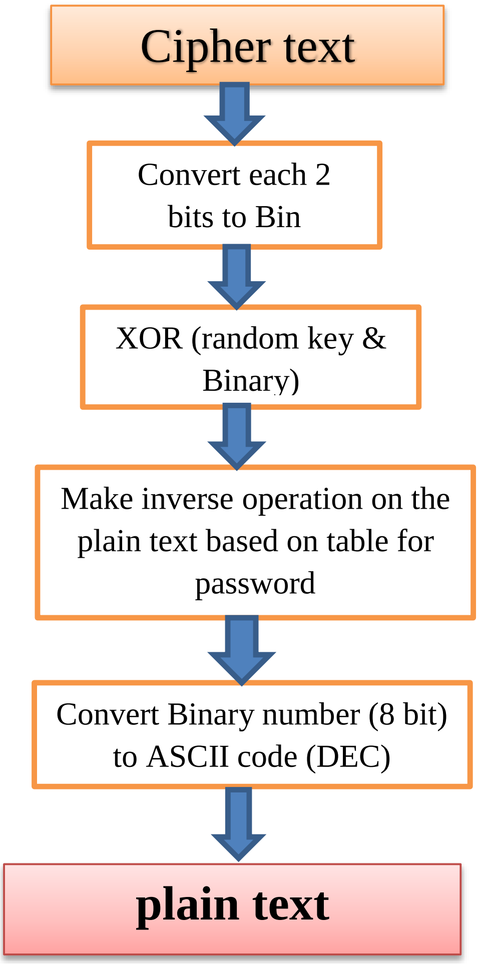
3.3.1 Encryption process flow


User Account
Password
Function to generate keys
char table for password(operation)
Compare password with plain text
Make operation on the plain text based on table for password
XOR (key generation (number 8 bit) & Binary number after operation)
Convert Binary number (8 bit) to HEX
Cipher text

3.4 Description of the Decryption Model

The proposed algorithm for decryption Column name includes converting its formula from hexadecimal to binary so that each two letters are taken from hexadecimal and transformed into 8-bit binary then run XOR operation with the same random key that was used in the encryption process, and then the process is opposite to what was done in the encryption step, for example, if shift right is done, shift left is performed, and so on. Then the binary formula is converted to decimal, and at the end the original text of column name appears after combining the letters.

3.4.1 Decryption process flow


Cipher text
Convert each 2 bits to Bin
XOR (random key & Binary)
Make inverse operation on the plain text based on table for password
Convert Binary number (8 bit) to ASCII code (DEC)
plain text

3.5 Encryption and Decryption Algorithm

```
START PROGRAM
CALL FUNCTION ENCRYPTION(STRING)
CALL FUNCTION DECRYPTION(STRING)
DEFINE FUNCTION ENCRYPTION (CIPHER_TEXT)
    DISPLAY (ENTER THE PASSWORD OF LENGTH 8 CHARACTERS)
    PASSWORD ← TAKE USER INPUT
    WHILE(LENGTH(PASSWORD) != 8)
        DISPLAY (WRONG PASSWORD: ENTER THE PASSWORD OF LENGTH 8 CHARACTERS)
        PASSWORD ← TAKE USER INPUT
    END WHILE
    PASSWORD_BINARY = CONVERT_TO_BINARY(PASSWORD)
    SEED = PASSWORD_BINARY
    SEED = CONVERT_TO_INTEGER (SEED)
    SEED = CONVERT_TO_DECIMAL (SEED)
    KEY = RANDOM_NUM_GENERATOR(SEED)
    READ COLUMN FROM FILE(FILE_NAME)
    WHILE (COLUMN_NAME != NULL)
        DTC = CONVERT_TO_DECIMAL(CHAR(COLUMN_NAME))
      BTC = CONVERT_TO_BINARY (CT)
        DTP = CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))
      BTP = CONVERT_TO_BINARY (CT)
    IF(32<=CONVERT_TO_DECIMAL(CHAR(BTP))=<45)
      CIRCULAR_SHIFT_LEFT (CHAR(BTC))
    IF(46<=CONVERT_TO_DECIMAL(CHAR(BTP))=<59)
      CIRCULAR_SHIFT_LEFT (CHAR(BTC))
        CIRCULAR_SHIFT_LEFT (CHAR(BTC))
    IF(60<=CONVERT_TO_DECIMAL(CHAR(BTP))=<73)
      CIRCULAR_SHIFT_LEFT (CHAR(BTC))
        CIRCULAR_SHIFT_LEFT (CHAR(BTC))
        CIRCULAR_SHIFT_LEFT (CHAR(BTC))
    IF(74<=CONVERT_TO_DECIMAL(CHAR(BTP))=<87)
      CIRCULAR_SHIFT_RIGHT (CHAR(BTC))
    IF(88<=CONVERT_TO_DECIMAL(CHAR(BTP))=<101)
```

```
        CIRCULAR_SHIFT_RIGHT (CHAR(BTC))
            CIRCULAR_SHIFT_RIGHT (CHAR(BTC))
      IF(102<=CONVERT_TO_DECIMAL(CHAR(BTP))=<115)
        CIRCULAR_SHIFT_RIGHT (CHAR(BTC))
            CIRCULAR_SHIFT_RIGHT (CHAR(BTC))
      IF(116<=CONVERT_TO_DECIMAL(CHAR(BTP))=<127)
        SWAP (CHAR(BTC))
  RESULT = BTC
      XOR(RESULT & KEY)
      INCREMENT CHAR(BTC)
      INCREMENT CHAR (BTP)
      END WHILE
      RESULT = CONVER_TO_HEXADECIMAL(RESULT)
      DISPLAY (RESULT)
END FUNCTION
DEFINE FUNCTION DECRYPTION (CIPHER_TEXT)
      WHILE (CIPHER_TEXT != NULL)
            DTC = CONVERT_TO_DECIMAL(CHAR(CIPHER_TEXT))
        BTC = CONVERT_TO_BINARY (CT)
            DTP = CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))
        BTP = CONVERT_TO_BINARY (CT)
            XOR(CHAR(CIPHER) & KEY)
      IF(32<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<45)
        CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
      IF(46<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<59)
        CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
            CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
      IF(60<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<73)
        CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
            CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
            CIRCULAR_SHIFT_RIGHT (CHAR(CIPHER_TEXT))
      IF(74<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<87)
        CIRCULAR_SHIFT_LEFT (CHAR(CIPHER_TEXT))
      IF(88<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<101)
        CIRCULAR_SHIFT_LEFT (CHAR(CIPHER_TEXT))
            CIRCULAR_SHIFT_LEFT (CHAR(CIPHER_TEXT))
      IF(102<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<115)
        CIRCULAR_SHIFT_LEFT (CHAR(CIPHER_TEXT))
            CIRCULAR_SHIFT_LEFT (CHAR(CIPHER_TEXT))
      IF(116<=CONVERT_TO_DECIMAL(CHAR(PASSWORD_TEXT))=<127)
        SWAP (CHAR(CIPHER_TEXT))
  RESULT = CIPHER_TEXT
      INCREMENT CHAR(CIPHER_TEXT)
      INCREMENT CHAR (PASSWORD_TEXT)
      END WHILE
      RESULT = CONVER_TO_STRING(RESULT)
      DISPLAY (RESULT)
END FUNCTION
END PROGRAM
```

3.6 The Generation of Key

The password was relied upon to create the key by making it as a seed by choosing the first bit of the first character and the second bit of the second character up to the eighth bit. Also, the password is used to generate random keys by entering them into a function used to encrypt column names with different keys.

3.7 Summary

In this chapter, a new method is proposed as a solution for Encryption of the column names of the DW to be stored in the cloud, contains multiple stages of securing DW, The password is used through a table that specifies the type of operation that takes place on each character that can be included in the password, which is applied to the name of the column to be encrypted, provided that it is not less than 8 characters in the password used.

CHAPTER FOUR

Implementation and Experimental Results

CHAPTER FOUR
Implementation and Experimental Results

4.1 Introduction

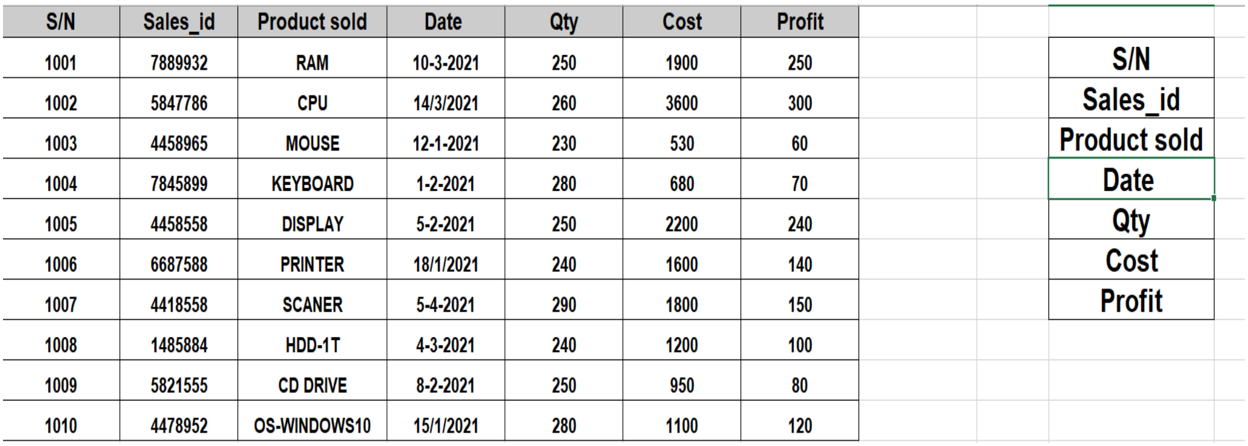
This chapter presents the implementation of the proposed model and the experimental results of the proposed algorithm which encrypts the column name of DW table as the column name that has encrypted with the same sequence of characters in the password, the key will be generated according to the password which entered by the user.

The proposed algorithm is coded using C++ programming language using online integrated development environment (Replit), that allows users to write code and build apps using a browser, Replit has various collaborative features such as a capability for real-time programming, code hosting platform, this helps us to deal with CSV files as a cloud DW.

4.2 Implementation

The encryption, decryption, and query (text & cipher) method, which are used in this study and the technique of generating cipher column names to maintain the security and integrity of the data that be stored on the cloud storage, are implemented using the C++ programming language.

As shown in Figure 5, the table containing the names of several columns and the details of each column were used from data on which the encryption and decryption process was performed in addition to the query, print screen were taken from the execution process and the time taken to carry out the operations was calculated.



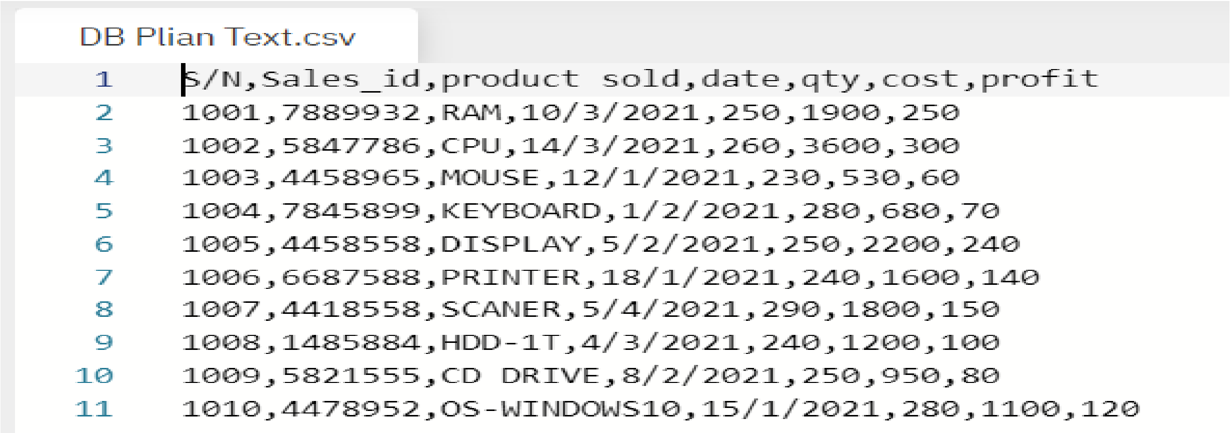
S/N	Sales_id	Product sold	Date	Qty	Cost	Profit
1001	7889932	RAM	10-3-2021	250	1900	250
1002	5847786	CPU	14/3/2021	260	3600	300
1003	4458965	MOUSE	12-1-2021	230	530	60
1004	7845899	KEYBOARD	1-2-2021	280	680	70
1005	4458558	DISPLAY	5-2-2021	250	2200	240
1006	6687588	PRINTER	18/1/2021	240	1600	140
1007	4418558	SCANER	5-4-2021	290	1800	150
1008	1485884	HDD-1T	4-3-2021	240	1200	100
1009	5821555	CD DRIVE	8-2-2021	250	950	80
1010	4478952	OS-WINDOWS10	15/1/2021	280	1100	120

S/N
Sales_id
Product sold
Date
Qty
Cost
Profit

Fig 5: Table detail example

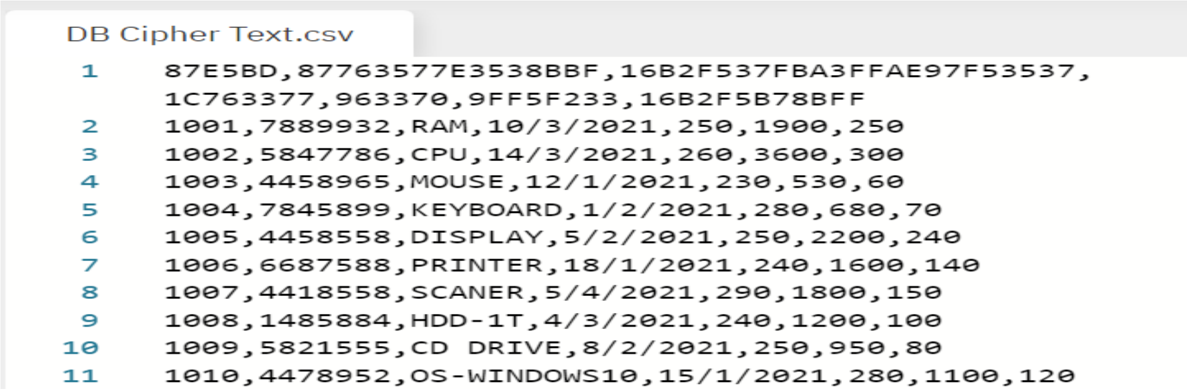
Basically, the operation of the program depends on the name of the column that is stored in the CSV file and then encrypted and stored in another CSV file in the cloud, shown in fig 6, 7.

Initially, when the program is run, it opens a data file to read the name of the entered column, and then the program asks the user to enter a password using a minimum of 8 characters, and based on it, the column name is encrypted.



```
S/N,Sales_id,product sold,date,qty,cost,profit
1001,7889932,RAM,10/3/2021,250,1900,250
1002,5847786,CPU,14/3/2021,260,3600,300
1003,4458965,MOUSE,12/1/2021,230,530,60
1004,7845899,KEYBOARD,1/2/2021,280,680,70
1005,4458558,DISPLAY,5/2/2021,250,2200,240
1006,6687588,PRINTER,18/1/2021,240,1600,140
1007,4418558,SCANER,5/4/2021,290,1800,150
1008,1485884,HDD-1T,4/3/2021,240,1200,100
1009,5821555,CD DRIVE,8/2/2021,250,950,80
1010,4478952,OS-WINDOWS10,15/1/2021,280,1100,120
```

Fig 6: DB Plain Text

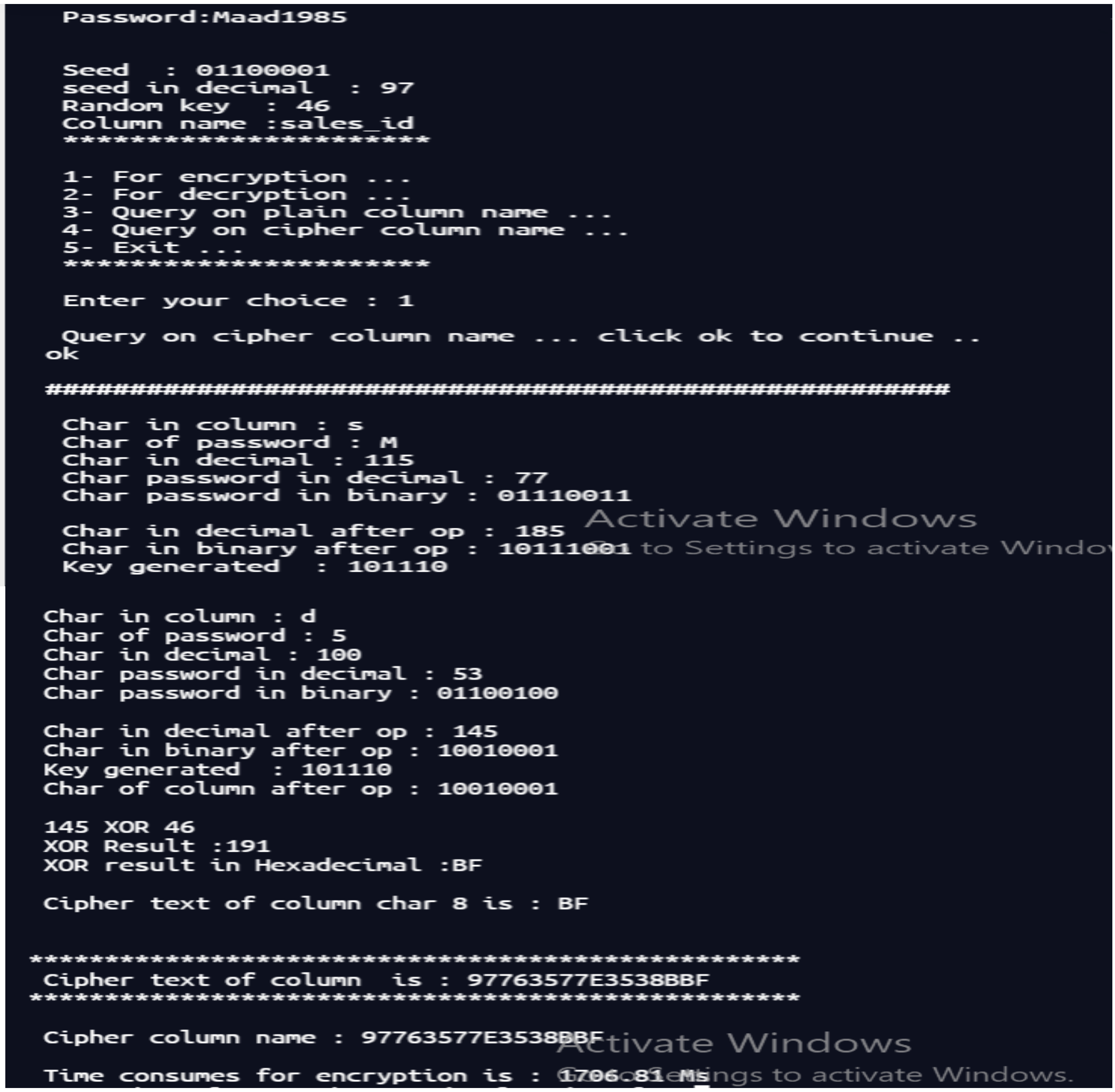


```
87E5BD,87763577E3538BBF,16B2F537FBA3FFAE97F53537,
1C763377,963370,9FF5F233,16B2F5B78BFF
1001,7889932,RAM,10/3/2021,250,1900,250
1002,5847786,CPU,14/3/2021,260,3600,300
1003,4458965,MOUSE,12/1/2021,230,530,60
1004,7845899,KEYBOARD,1/2/2021,280,680,70
1005,4458558,DISPLAY,5/2/2021,250,2200,240
1006,6687588,PRINTER,18/1/2021,240,1600,140
1007,4418558,SCANER,5/4/2021,290,1800,150
1008,1485884,HDD-1T,4/3/2021,240,1200,100
1009,5821555,CD DRIVE,8/2/2021,250,950,80
1010,4478952,OS-WINDOWS10,15/1/2021,280,1100,120
```

Fig7: DB Cipher Text

4.3 Encryption Module

In this module the encryption process of the plain text (column name) of the cloud DW table is performed, to generate the ciphertext column name that will be stored in the cloud as the data of the organization that needs to be protected. The encryption module allows the secure storage of important data of sales for customers like **sales_id**. As shown in figure 8.

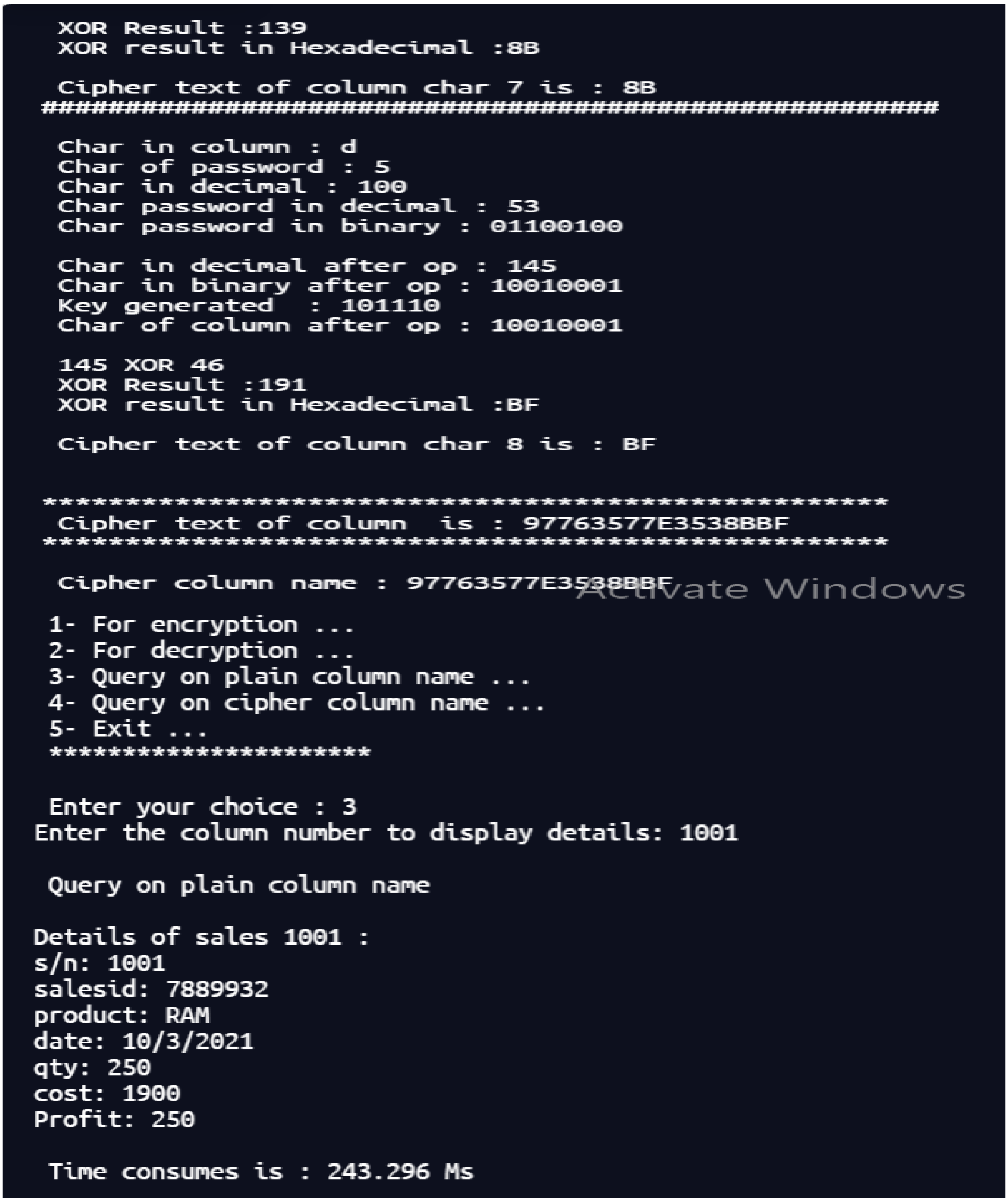


```
   Password:Maad1985


   Seed   : 01100001
   seed in decimal   : 97
   Random key   : 46
   Column name :sales_id
   **********************

   1- For encryption ...
   2- For decryption ...
   3- Query on plain column name ...
   4- Query on cipher column name ...
   5- Exit ...
   **********************

   Enter your choice : 1

   Query on cipher column name ... click ok to continue ..
  ok

  ######################################################

   Char in column : s
   Char of password : M
   Char in decimal : 115
   Char password in decimal : 77
   Char password in binary : 01110011

   Char in decimal after op : 185
   Char in binary after op : 10111001
   Key generated  : 101110

  Char in column : d
  Char of password : 5
  Char in decimal : 100
  Char password in decimal : 53
  Char password in binary : 01100100

  Char in decimal after op : 145
  Char in binary after op : 10010001
  Key generated  : 101110
  Char of column after op : 10010001

  145 XOR 46
  XOR Result :191
  XOR result in Hexadecimal :BF

  Cipher text of column char 8 is : BF


 ****************************************************
  Cipher text of column  is : 97763577E3538BBF
 ****************************************************

  Cipher column name : 97763577E3538BBF

  Time consumes for encryption is : 1706.81 Ms
```

Fig8: Implementation of encryption process

4.4 Query on plain column name Module

In this module the query process is performed on the plain column name, this function is done before encrypting the column name or after encrypting the column name and in this situation, decryption is needed before the query is performed, as shown in figure 9.

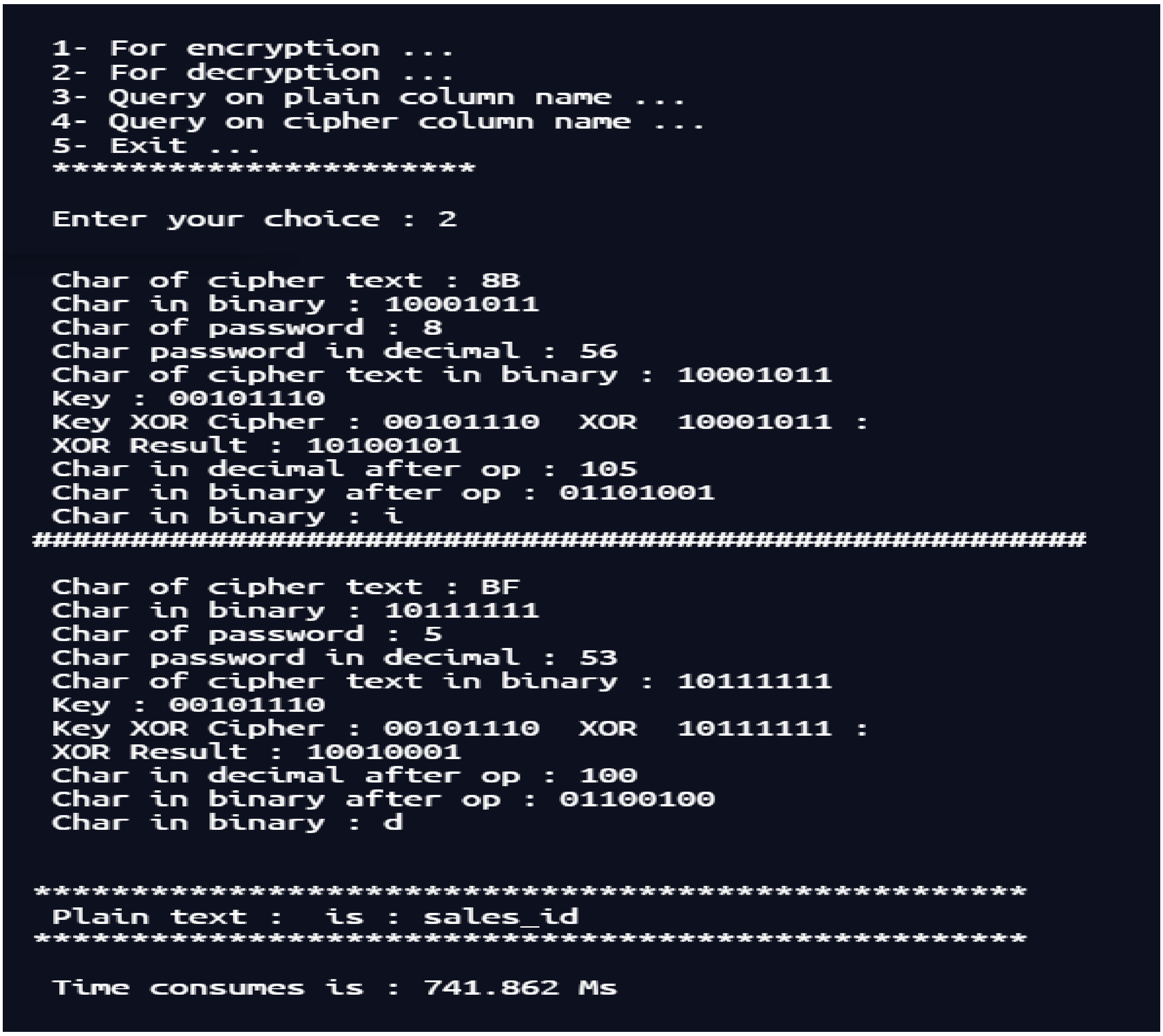


```
   XOR Result :139
   XOR result in Hexadecimal :8B

   Cipher text of column char 7 is : 8B
  ##################################################################

   Char in column : d
   Char of password : 5
   Char in decimal : 100
   Char password in decimal : 53
   Char password in binary : 01100100

   Char in decimal after op : 145
   Char in binary after op : 10010001
   Key generated  : 101110
   Char of column after op : 10010001

   145 XOR 46
   XOR Result :191
   XOR result in Hexadecimal :BF

   Cipher text of column char 8 is : BF


  **************************************************************
   Cipher text of column  is : 97763577E3538BBF
  **************************************************************

   Cipher column name : 97763577E3538BBF

  1- For encryption ...
  2- For decryption ...
  3- Query on plain column name ...
  4- Query on cipher column name ...
  5- Exit ...
  **********************

  Enter your choice : 3
 Enter the column number to display details: 1001

  Query on plain column name

 Details of sales 1001 :
 s/n: 1001
 salesid: 7889932
 product: RAM
 date: 10/3/2021
 qty: 250
 cost: 1900
 Profit: 250

  Time consumes is : 243.296 Ms
```

Fig9: Implementation of Query on plain column name process

4.5 Decryption Module

In this module the cipher column name will be returned into its original plain column name when the user makes a query for data, using a cipher column name query, downloads the cipher column name from the cloud table, and performs the decryption process. In this way, the stored table and the query will be in cipher column name, which prevents unauthorized data access to the stored table and the query, as shown in figure 10.

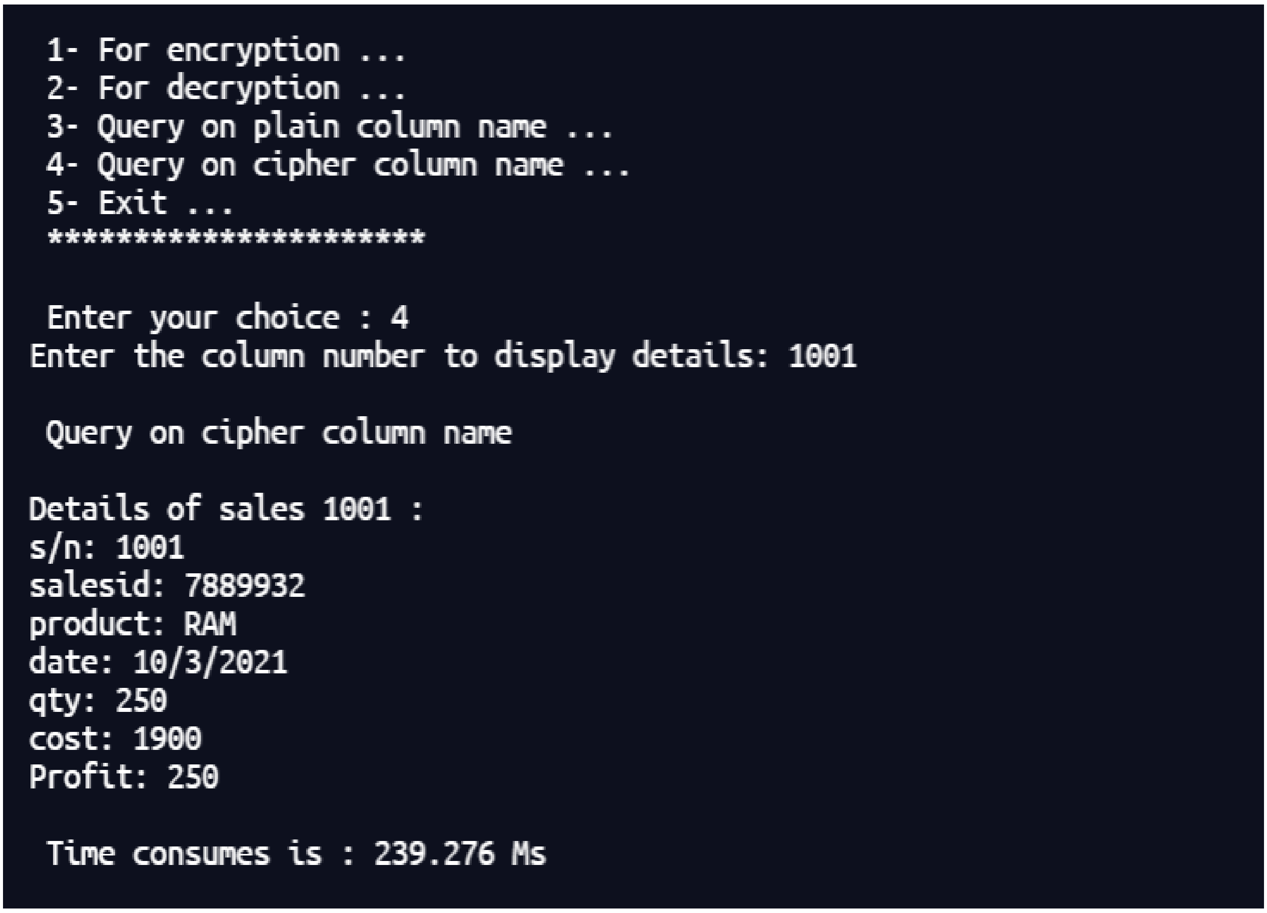


```
  1- For encryption ...
  2- For decryption ...
  3- Query on plain column name ...
  4- Query on cipher column name ...
  5- Exit ...
  **********************

  Enter your choice : 2

  Char of cipher text : 8B
  Char in binary : 10001011
  Char of password : 8
  Char password in decimal : 56
  Char of cipher text in binary : 10001011
  Key : 00101110
  Key XOR Cipher : 00101110  XOR  10001011 :
  XOR Result : 10100101
  Char in decimal after op : 105
  Char in binary after op : 01101001
  Char in binary : i
 #######################################################

  Char of cipher text : BF
  Char in binary : 10111111
  Char of password : 5
  Char password in decimal : 53
  Char of cipher text in binary : 10111111
  Key : 00101110
  Key XOR Cipher : 00101110  XOR  10111111 :
  XOR Result : 10010001
  Char in decimal after op : 100
  Char in binary after op : 01100100
  Char in binary : d


 ***************************************************
  Plain text :  is : sales_id
 ***************************************************

  Time consumes is : 741.862 Ms
```

Fig10: Implementation of decryption process

4.6 Query on Cipher Column Name Module

In this module the query process is performed on the cipher column name, this function is done after encrypting the column name or after decrypting the column name but in this situation, encryption is needed before the query is performed, as shown in figure 11.



```
 1- For encryption ...
 2- For decryption ...
 3- Query on plain column name ...
 4- Query on cipher column name ...
 5- Exit ...
 **********************

 Enter your choice : 4
Enter the column number to display details: 1001

 Query on cipher column name

Details of sales 1001 :
s/n: 1001
salesid: 7889932
product: RAM
date: 10/3/2021
qty: 250
cost: 1900
Profit: 250

 Time consumes is : 239.276 Ms
```

Fig11: Implementation of Query on cipher column name process

4.7 Experimental Results

In the table below, the password **(Maad1985)** was used in the implementation of the algorithm to encrypt, decrypt and query (text, cipher) and show the time taken to complete the process as table 2.

Table 2: Experimental Result

column name Plain text	column name Cipher text	Encryption Time	Query on cipher column name Time	Decryption Time	Query on plain column name Time
S/N	87E5BD	562.486 Ms	198.867 Ms	315.236 Ms	185.882 Ms
Sales_id	87763577E 3538BBF	1706.81 Ms	243.296 Ms	741.862 Ms	239.276 Ms
Product sold	16B2F537FBA3 FFAE97F53537	1770.33 Ms	266.637 Ms	815.364 Ms	183.106 Ms
Date	1C763377	733.994 Ms	217.699 Ms	436.966 Ms	140.875 Ms
Qty	963370	721.673 Ms	163.473 Ms	335.568 Ms	169.533 Ms
Cost	9FF5F233	477.788 Ms	223.331 Ms	477.992 Ms	165.708 Ms
Profit	16B2F5B78BFF	917.135 Ms	216.788 Ms	529.61 Ms	290.24 Ms
Average	----	984.316 Ms	218.584 Ms	521.799 Ms	196.377 Ms
Total	----	6890.216 Ms	1530.091 Ms	3652.598Ms	1374.64Ms

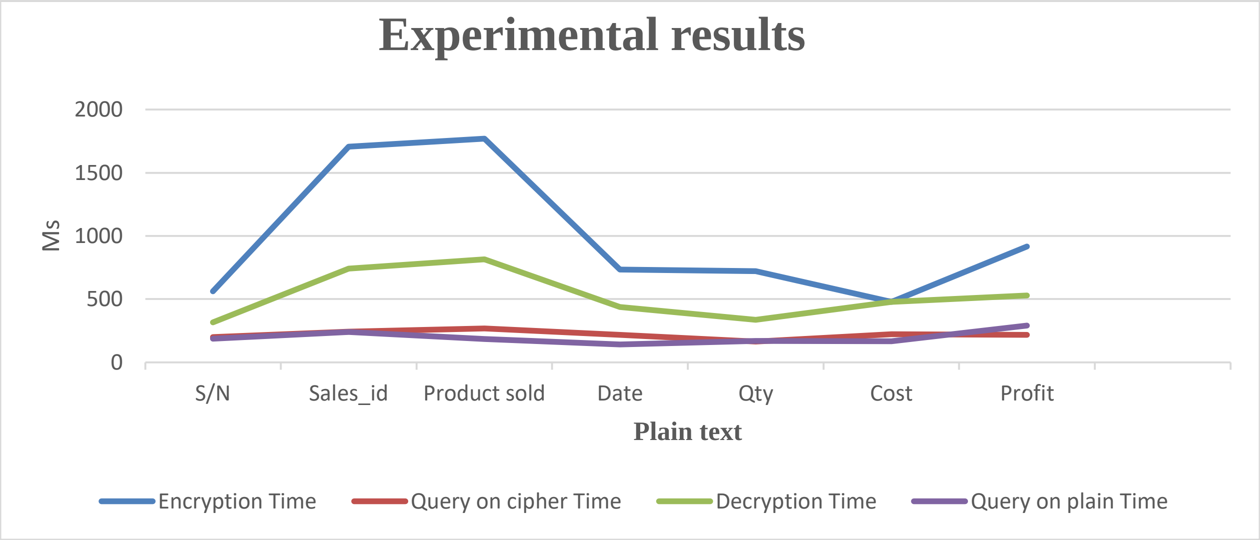


Fig12: Line chart for comparison among of time spent of data retrieval to encryption and query on cipher and decryption and query on plain.

Through the graph, the difference in the execution time in the above example depends on the size of the column name. In the case of encryption and decryption, the word consisting of more letters takes more time, but in the case of querying the column name in the case of the query and the normal and encrypted it turns out that it does not take much time as shown in figure 12.

In addition, the total time for encrypting the entire table takes almost twice the time in the decryption process. In the query process, the time difference is small between the query (text, cipher).

4.8 Summary

This chapter presented the implementation of the proposed model using the C ++ language using a hypothetical table of column names and applying several operations of encryption and decryption and querying (text, cipher).

After the user enters the password, the text encryption method is adopted with the same sequence of characters in the password in a special operations table, and then an XOR operation is performed with the random key that was generated to extract the cipher text. This is the answer to the first question of the study questions.

For the process of creating the key by making the password as a seed by choosing the first bit of the first character and the second bit of the second character up to at least the eighth bit and it can be more than 1 byte, which gives the number of possibilities for the key at least 2^64 and the password was used Generate random keys by entering them into a function used to encrypt column names with different keys. This is the answer to the second question in the study questions.

as the test results showed that the time spent in the encryption and decryption process is more than in the query process, in addition to the size of the column name. It is the answer to the third question of the study.

The encryption, decryption and query operations were carried out on a single table that contains the names of the columns, including the data, and can be developed and worked on a complete DW that contains a link between the tables. This is the answer to the fourth question of the study questions.

CHAPTER FIVE

Conclusion and Future Work

CHAPTER FIVE
Conclusion and Future Work

5.1 Conclusion

In this thesis, a new encryption method is used to encode the column name in the DW aims to preserve the integrity of this important information before uploading it to the cloud, as the results were shown by encrypting a table from the DW and decryption and query(text, cipher) and calculating the time required for execution, as most studies the previous one worked on encrypting the column name in a simple way, and therefore the time for executing encryption, decryption and query(text, cipher) is relatively acceptable compared to the new idea that was implemented in terms of the size of the key used and the use of a different password depending on the user to determine the operations that take place on the name of the column to be encrypted where once it changed The password changes the entire encryption method.

The contributions of this study are as follows:

1. Introducing a new method of Encrypting data that ensures data integrity and secrecy.
2. The proposed Encryption methods use a key generated according to the seed created according to the password entered by the user.
3. The key is subject to change from one user to another according to the entered password.
4. Information retrieval method for the data without compromising the data for unauthorized data access.

5.2 Future Work

Based on the current research on encryption, this encryption can be applied after building a complete DW and showing the results to achieve the integrity and Enhancement of time efficiency of the proposed model, and Enhancement of encryption and query model to deal with NoSQL DB, in addition it can be used in encryption small text documents.

References

Ali, A., & Afzal, M. M. (2017). Database Security: Threats and Solutions. *International Journal of Engineering Inventions, 6*(2), 25–27. www.ijeijournal.com

Almeida, F. (2017). Concepts and Fundaments of Data Warehousing and OLAP. In *INESC TEC and University of Porto* (Vol. 1, Issue January).

Al-rammahi, A. H. I. (2016). *DESIGNING A VARIETY OF DATA WAREHOUSE SCHEMAS SUITABLE FOR META-. May*.

Al-Saraireh, J. (2017). An efficient approach for query processing over encrypted database. *Journal of Computer Science*, *13*(10), 548–557. https://doi.org/10.3844/jcssp.2017.548.557

Arora, A., & Gosain, A. (2020). Mechanism for securing cloud-based data warehouse schema. *International Journal of Information Technology (Singapore)*. https://doi.org/10.1007/s41870-020-00546-1

Attasena, V., Harbi, N., & Darmont, J. (2015). A novel multi-secret sharing approach for secure data warehousing and on-line analysis processing in the cloud. *International Journal of Data Warehousing and Mining*, *11*(2), 22–43. https://doi.org/10.4018/ijdwm.2015040102

Ballard, C., Herreman, D., Schau, D., Bell, R., Kim, E., & Valencic, A. (2012). Data Modeling Techniques for Data Warehouse. *Zenithresearch.Org.In*, *2*(2), 195–196.

Benjelloun, M., El, M., & Amin, E. (2018). Impact of using Snowflake Schema and Bitmap Index on Data Warehouse Querying. *International Journal of Computer Applications*, *180*(15), 33–35. https://doi.org/10.5120/ijca2018916286

Blažiü, G., Pošþiü, P., & Jakšiü, D. (n.d.). *Data Warehouse Architecture Classification*.

Divya Shaly, C., & Anbuselvi, R. (2016). CHARM: A cost-efficient multi-cloud data hosting scheme with high availability. *International Journal of Control Theory and Applications*, *9*(27), 461–468. https://doi.org/10.18535/ijecs/v6i6.44

Institute of Electrical and Electronics Engineers, IEEE International Conference on Data Engineering 26 2010.03.01-06 Long Beach, Calif., & ICDE 26 2010.03.01-06 Long Beach, Calif. (n.d.). *IEEE 26th International Conference on Data Engineering (ICDE), 2010 Long Beach, California, USA, 1 - 6 March 2010*.

Iqbal, M. Z., Mustafa, G., Sarwar, N., Wajid, S. H., Nasir, J., & Siddque, S. (2020). A Review of Star Schema and Snowflakes Schema. *Communications in Computer and Information Science*, *1198*(May), 129–140. https://doi.org/10.1007/978-981-15-5232-8_12

Konda, S., & More, R. (2015). Augmenting Data Warehouse Security Techniques-A Selective Survey. *International Research Journal of Engineering and Technology*. www.irjet.net

Malinowski, E., & Zimányi, E. (2008). Advanced Data Warehouse Design from Conventional to Spatial and Temporal Applications. *Data Vault 2.0*, 1–15.

Moghadam, S. S., Darmont, J., & Gavin, G. (2017). S4: A New Secure Scheme for Enforcing Privacy in Cloud Data Warehouses. *ArXiv*.

Pacheco, A., & Mar, G. (2018). *Research and Practical Issues of Enterprise Information Systems* (Vol. 310). Springer International Publishing. https://doi.org/10.1007/978-3-319-99040-8

Simmon, E. (2018). Evaluation of Cloud Computing Services Based on NIST 800-145. *National Institute of Standards and Technology Special Publication NIST*, *NIST Speci*(February), 1–24. https://www.nist.gov/sites/default/files/documents/2017/05/31/evaluation_of_cloud_computing_services_based_on_nist_800-145_20170427clean.pdf

Tole, A. A. (2015). Cloud Computing and Business Intelligence. *Database Systems Journal*, *V*(4), 49–58.

Yang, Q., Ge, M., & Helfert, M. (2019). Analysis of data warehouse architectures: Modeling and classification. *ICEIS 2019 - Proceedings of the 21st International Conference on Enterprise Information Systems*, *2*, 604–611. https://doi.org/10.5220/0007728006040611

Gandhi, V., & Kumbharana, C. K. (2018). *Comparative study of Amazon EC2 and Microsoft Azure cloud architecture*. *September*.

Khan, M. A., Mishra, K. K., Santhi, N., & Jayakumari, J. (2015). A new hybrid technique for data encryption. *Global Conference on Communication Technologies, GCCT 2015*, *Gcct*, 925–929. https://doi.org/10.1109/GCCT.2015.7342801

Simões, D. (2010). Enterprise Data Warehouses: A conceptual framework for a successful implementation. *Canadian Council for Small Business & Entrepreneurship Annual Conference (CCSBE 2010), January 2010*.