Enhancing the QoS of a Data Warehouse through an Improved ETL Approach

Zahir Khan
United States International University-Africa
zkhan@usiu.ac.ke

Leah Mutanu
United States International University-Africa
lmutanu@usiu.ac.ke

Abstract
This study aims at developing a data warehouse with enhanced security and Quality of Service (QOS) by enriching the Extract Transform Load (ETL) process. Existing Data warehouse models, which include the Relational core model, the Dimensional core model, and the Data Vault Model fail to either adequately address user requirement, perform ad-hoc queries, or require vast amounts of storage space and computational power. The proposed model addresses these challenges with Remote Sync (RSYNC) Utility to improve the performance of a data warehouse, Secure Shell (SSH) protocols to enhance security, and the nearest neighbor approach for more flexible data extraction and loading process. The research used an experimental design to implement a prototype and data collected by simulating laboratory experiments. When compared to the traditional models, the enhanced model improved Extraction by enhancing the flexibility of ad-hoc queries, introducing host-based Authentication, and reducing the data transmitted between the source and destination.

Keywords. *Data Warehouse, Secure Shell protocols, Remote Sync utility, Extract Transform Load process, Quality of Service*

1. Introduction

The value of big data comes from the intelligence it can provide after analysis (Gouret al., 2010). This analysis starts with the integration of data into a data warehouse. This integration is achieved via a technique known as ETL (Extract, Transform, Load) (Nwokeji et al., 2018). ETL is typically carried out in three steps: (i) *Extract*- data is retrieved from various sources, (ii) *Transform* - data is cleaned, filtered, normalized, and sorted using various transformation techniques; and (iii) *Load*- data is imported/loaded into a centralized data store for processing and analyzed.

While investigating Security Measures for Web ETL Processes, Dammak et al., (2016) stated that the design and security of ETL is a key factor in determining the success of a data warehouse. Despite this, research in ETL security and performance is limited focusing mainly on the extract stage of the ETL process. The aim of this study Design and implement an enhanced data warehouse model with a specific focus on securing the Transform and Load stages of the ETL process where data is most vulnerable i.e. Data in motion (Janacek, 2015) while improving data quality.

The specific objectives that guided the study focused on: *(i)* improving the performance of a data warehouse by reducing data latency through Remote Sync (RSYNC) utility during the ETL Loading phase, *(ii)* leveraging Secure Shell (SSH) protocols for securing data transmitted between the source systems and data warehouse during transmission in the ETL process, and *(iii)* improving data quality within the data warehouse by using data grouping (clustering) within the ETL Extraction phase.

2. Review of related work

Often, business intelligence (BI) systems are implemented as separate entities that interact with transactional systems (Nedelcu, 2012). To realize their full potential, BI systems should work with a data warehouse for more in-depth analytics that can utilize data from integrated systems (Ally, 2016). Similarly, a data warehouse without a BI system remains “underutilized and untapped” (Chen, 2012). Because data warehouses integrate data from disparate systems, they are considered as a proactive approach to information integration as compared to the more traditional passive approaches where the data processing begins when a query arrives (Di Tria F, 2012). However, the process of integrating BI Systems with data warehouses is often constrained by various factors resulting in degraded performance of the entire system. Measuring the performance of a data warehouse, the data quality, query response time, and data latency are key factors in assessing the performance of a data warehouse (Rahman, 2013). ETL Complexity, Robustness, and management also form technical yardsticks for measuring a data warehouse's performance.

To address this issue this study started by developing a data warehouse prototype that would provide the testbed for the enhanced model. Several models have been proposed for developing data warehouses (Mathiews, 2012). Three popular models are, a subject modeling approach referred to as the *Relational Core Model* (Inmon, 2005), a dimensional modeling approach referred to as the *Dimensional Core Model* (Kimball and Ross, 1996), and a scalable approach known as the *Data Vault Model* (Dan Linstedt, 1990).

2.1 Data Warehouse Models

2.1.1 Relational Core Model

The Relational Core approach, being a top-down approach (Aljawarneh, 2016), is a “data-driven” approach that designs the warehouse based on the entity-relationship diagrams of the transactional systems and does not consider the user requirements duringdesign (Aljawarneh, 2016; Martino, 2014; Yessad, 2016; Oketunji and Omodara, 2011). In this model, the entire dataset from the source is extracted onto the Data Warehouse, without focusing on the user requirements at design time (Yessad, 2016). The data is transformed into structured relationships ensuring that all the tables adhere to relational integrity rules as illustrated in the example in Figure 1.

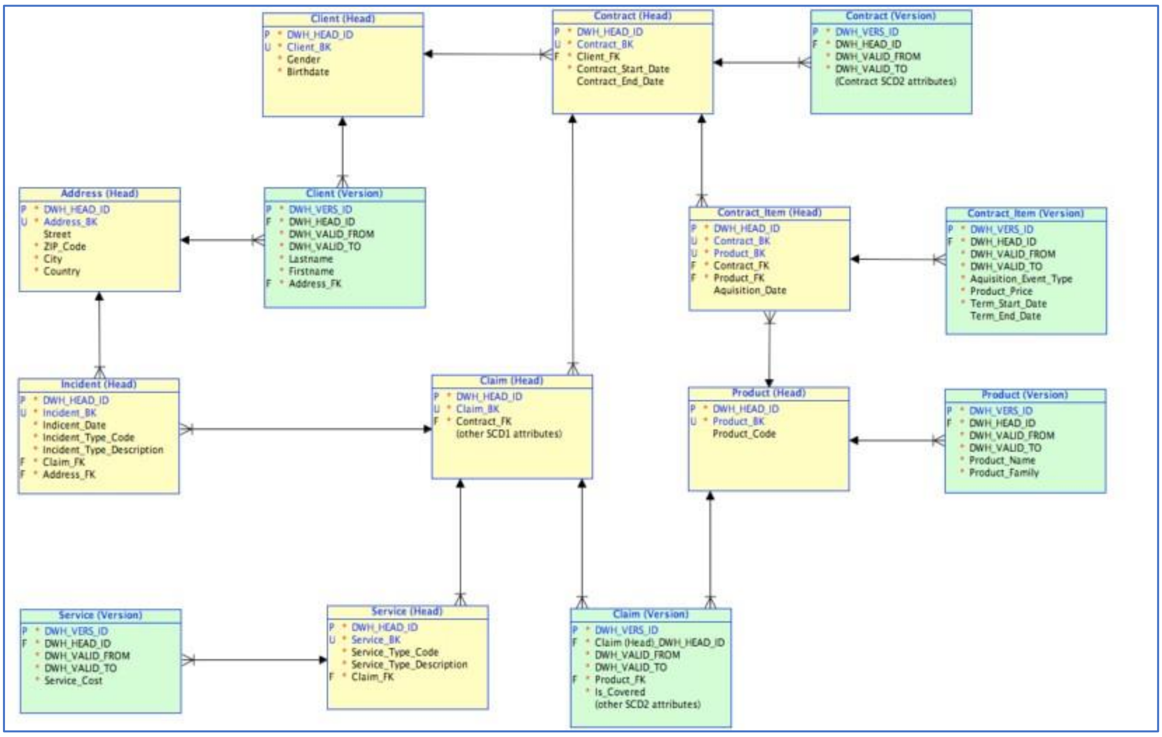


Figure 1: Relational Core Schema, Source: (Martino, 2014)

It is worth noting that this approach ignores user requirements hence requiring data from all the source systems extracted and loaded into the warehouse, at the time of data collection. This makes the approach flexible enough to handle any ad hoc queries. However, the Business requirements definition involves collaboration with business users to understand their requirements and ensure that there is buy-in to the data warehouse/BI project (Kimball, 2013).

2.1.2 Dimensional Core Model

This approach involves users in the early stages of the project, hence dubbed as "User requirements-driven" or bottom-up approach (Aljawarneh, 2016, Rorimpandey et al., 2018). The Dimensional Core Model improves the performance of query execution by extracting relevant data only, rather than the entire dataset. However, it cannot perform ad hoc queries on dimensions that were not included in the iil design. The example in Figure 2 illustrates how the dimensional core model works by retrieving fact tables (in orange) and the dimensions tables only (in green) during the ETL process.

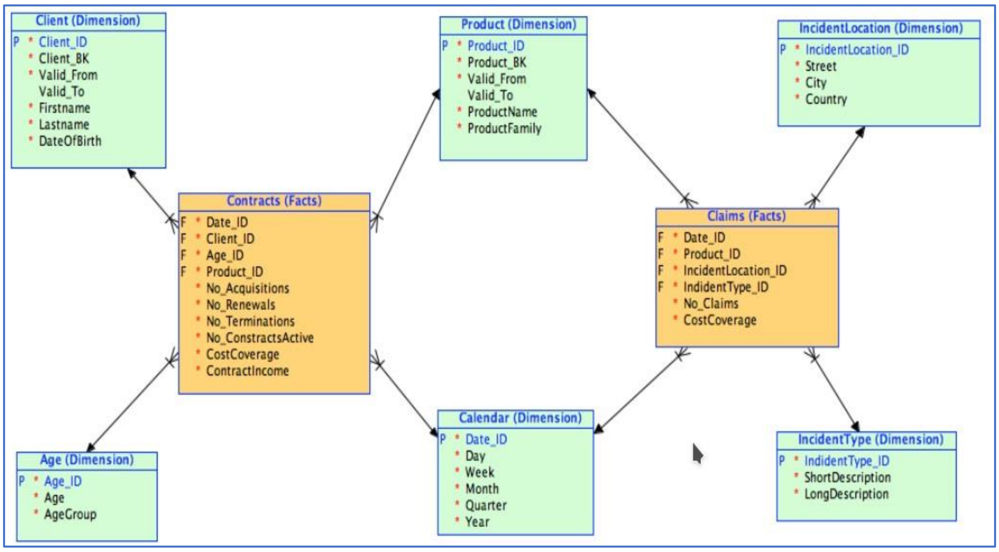


Figure 2: Dimensional Core Schema: Source: (Martino, 2014)

2.1.3 Data Vault Model

The approach makes use of a data vault, which is the actual data warehouse. The data warehouse contains the data fromthe various source systems without any transformation. It is suitable for environments where the data is coming from several sources, thus calling for rapid system adaptation to different environments (Martino, 2014). Data within the vault is never changed, enabling reuse without the need to query fresh data. In this regard, they respond well to ad hoc queries. However, the absence of a transformation process calls for a vast amount of storage space and high computational resources to host and run queries due to the lack of uniform structures. The example in Figure 3 shows data that has redundant relationships (e.g. HUB_CONTRACT) or no relationships (e.g. REF_DATE).

From these reviews, we note that each of the three popular models has strengths and weaknesses. The comparative analysis of these models reveals that the Data vault Model has the fewest merits compared to the other models as seen in Table 1. The other models both have merits that significantly enhance the quality of service of data warehouses. In light of these findings, the study opted to design a model adapted from both the Relational and Dimensional core models.

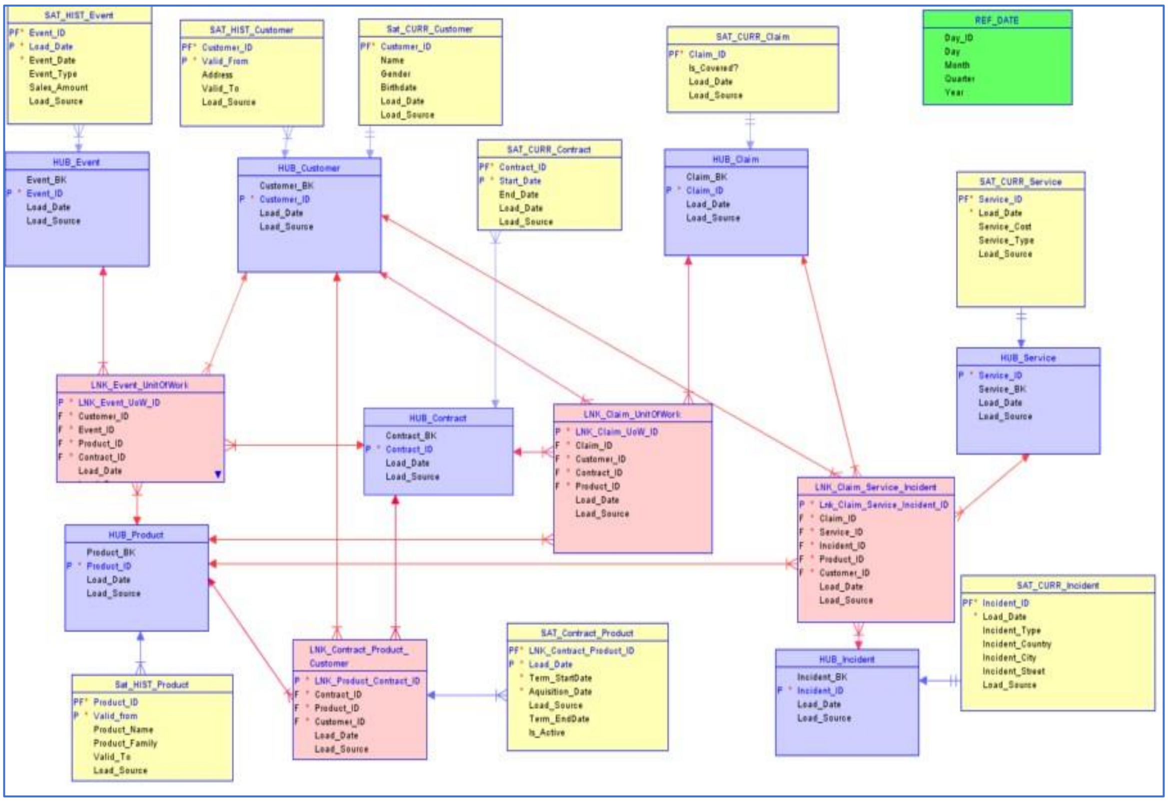


Figure 3: Data Vault Model, Source: (Yessad, 2016)

Table 1: Comparison of Warehouse models

	Relational Core	**Dimensional Core**	**DataVault**	**Referen ces**
Data Integration	Enterprise-Wide	Individual Business requirements	Enterprise-Wide	(George,2012) (Yessad, 2016)
Loading of fact Tables	Easy because facts have no versiontable	Very Easy due to similar structure in data marts and core	Incremental Loading of the fact tables impacts hugely on performance	(Martino,2014) (Yessad, 2016)
Complexity of ETL	Relatively Simple	Very Complex (Transformation ofdata from OLTP to Dimensional is complex)	Simple Data loading no transformation required	(Martino,2014) (Yessad, 2016)

Query Performance	Slow due to the normalization in the relational core	Very fast due to the Dimensional core	Very Slow due to the high data standardization, Data martsare required for reporting and analysis.	(Yessad, 2016)

Table Key: Merit, demerit

2.2 Performance Enhancing Approaches

2.2.1 Remote Sync utility

RSYNC is a powerful system tool that excels at efficient file synchronization (Tan-pure et al., 2015). It efficiently synchronizes files by identifying the changed regions between the source and destination files and only moving those blocks (Tanpure et al., 2015). By using a remote-update protocol, which allows the transfer of differences between two sets of files, the execution time of the system is enhanced. Rsync algorithm consists of three basic steps: (i) Signature generation: A signature block describing an existing file is generated (ii) Signature Search: Finds the difference between the file data and new data, and (iii) Reconstruction: The differences are applied to the old data to generate the new data (Tanpure et al., 2015). This process is useful in synchronizing data warehouses when extracting, transforming, and loading data.

2.2.2 Secure shell Protocol

This is an open-source software package that provides a command shell, data tunneling services for TCP/IP-based applications, and secure file transfer (Michael and Karthikeyan, 2017). SSH connections give extremely secure encryption, authentication, and data integrity (Michael and Karthikeyan, 2017). There are several ways to use SSH, one way is to use 'Host Authentication' (Michaeland Karthikeyan, 2017) where the authentication is based on a manually generated public-private key pair, this form of authentication can allow users to log onto remote systems without having to provide a password (Garimella and Kumar, 2015). The other way is using the public-private key pair for the tunnel encryption and the user then has to provide a password for authentication (Garimella and Kumar, 2015). Secure shell provides all the security facets of confidentiality, integrity, and authentication. This research will focus on achieving both confidentiality and integrity through host authentication and data encryption respectively. Integrity will also be maintained by the SSH tunnel for data in motion using the data padding that is added to each SSH package.

2.2.3 Nearest Neighbor Query Approach

While the dimensional core model performs better because it only hosts the required user data required, it is not flexible where ad hoc queries are concerned. In their work, Mehdi and Beheshti (Mehdi and Beheshti, 2013) propose a solution that groups related entities, by performing some analysis and then using those groups to respond to Ad-hoc queries. In this way, they can perform an analysis on just a given group of neighboring tables instead of the full data set hence improving on ad hoc queries while maintaining good data warehouse performance. This study adopted the same immediate neighbor query approach to bridge the gap between the issues in the relational core and dimensional core warehouses. In this study, we shall refer to it as the nearest neighbor query approach.

`

3. Solution Design

3.1 Use Case

To design and test the solution this research used a university Enterprise Resource Planning (ERP) system as a use case. The ERP that hosts the Main organization's processes has over 4000 tables. Most of these tables contain system data that is not required for making business decisions. This study made use of entities relating to student decisions that the management needs to make and any neighboring entities.

3.2 Refined Data warehouse model

The study designed a Relational - Dimensional Core hybrid model, which enforced the relational core integrity rules while extracting dimensional tables for each fact table identified. For example in Figure 4, the fact table “Faculty” has a dimensional table “Departments” which is pulled based on the Dimensional core model and relational integrity rules applied based on the relational core model. This model was implemented tests conducted to establish its ability to enhance data quality (flexible ad hoc queries) while improving performance by reducing data latency (Time taken to transmit data from the source system to the warehouse).

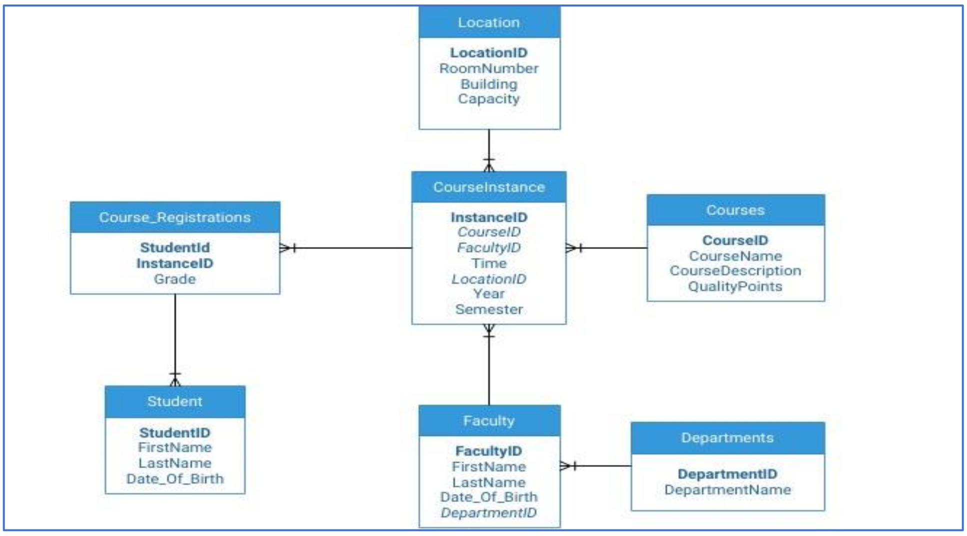


Figure 4: Relational - Dimensional Core hybrid model

3.3 Prototype Implementation

Research (Plechawska-W’jcik and Rykowski, 2016) on databases shows that Postgresql performs better than most database tools and its open source. Therefore, to develop the prototype this research used Postgresql, running on an Ubuntu Operating System. The study created a shell script to filter and extract relevant tables. The generated list of tables is cleaned to remove duplicates, delimiters, and blank lines. The refined table list is converted into an SQL script by adding SQL statements used to load the files in the data warehouse via SSH and RSYNC utilities. The algorithm in Listing I is derived from this process and used to implement the shell script used to implement the ETL process.

For both host authentication and data transfer encryption, SSH was used. In host authentication, the public key of the client machine is generated and placed in the authorized keys file of the SSH server. This allows the client machine to authenticate against the server without the need to provide a password. This in addition allows for automation of SSH-based processes because there is no need for the user to put in a password because the authentication is key-based (Garimella and Kumar, 2015).

```
read: keywords
while tablescount ≤ source tables
    if keyword ϵ tablename then
        tablelist ← tablename
    end if
    if relation ϵ tablesname then
        tablelist ← relation tablename
    end if
end while
read: tablelist
while ~end of file
    if duplicate tables → tablelist then
        drop() - delete table
 end while
 read: tablelist
while ~end of file
    tablename← tablelist
    Extraction(tablename) -  read data from tablelist.
    Transform(data) - clean and format data.ϵɜ
    Load(data) - Initiate RSYNC over SSH and copy data
end while
```

Listing I: Shell Script Algorithm for the ETL process

4. Experimental Results and Discussions

This section describes the experiments conducted to investigate the prototype's ability to enhance the quality of data warehouses. Specifically, the experiments wanted to find out the model's ability to improve performance, security, and the quality of data in a data warehouse by using RSYNC, SSH, and Nearest Neighbor Query approach data grouping as described by the objectives.

4.1 Performance Results using RSYNC

The study conducted several instances of RSYNC file transfer to evaluate the average performance of RSYNC. The results show that the speeds improved significantly. The initial file transfer always took a large amount of time; however, subsequent transfers were significantly faster. This is expected given that RSYNC functions initially transfer all the data and subsequently only the modified bits of data. A screenshot of one of the RSYNC runs is shown in Figure 5.

To test if RSYNC improved data latency, tests were conducted by transferring small (488 MB) and large datasets (2339MB) via SCP (secure copy: i.e. A direct copy of files via SSH). The average transfer time for 488 MB via SCP resulted in 11.667 Seconds while for 2339MB of data yielded 81.667 Seconds. The tests using 488MB of data were repeated using RSYNC over SSH resulted in significantly slower speed (20.667Seconds), however similar tests using 2339MB of data significantly improved speeds (21 seconds) as tabulated in Tables 2 and 3. The experiments excluded the initial transfer time because it only executes once in the lifetime of the ETL, during its initial run, RSYNC will just copy over the changed bits of the data thus giving a massive improvement in ETL performance. Thus, for small data, SCP is significantly faster than SSH + RSYNC while for large datasets the latter works best. These findings concurred with the study by Wilman Banditvilai et al (2014) and Tanpure et al (2015) who noted that a reduction in data latency is seen in subsequent data transfers where the data transfers are large.

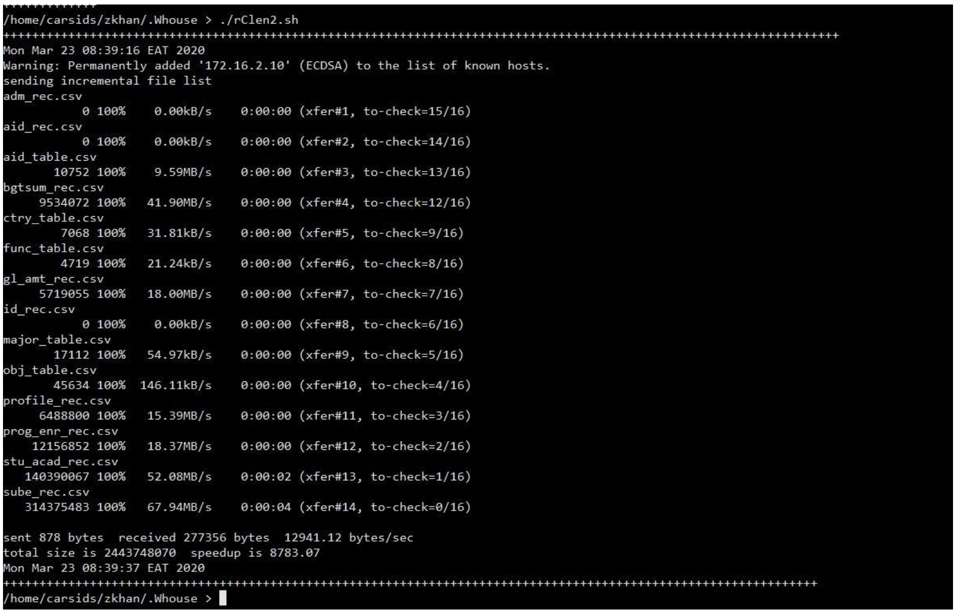


Figure 5: RSYNC incremental data transfer

Table 2: Comparative Results of 488MB Data Transfer

	Time in	Seconds
	SSH + RSYNC	SCP
Initial Transfer	21	10
Transfer 2	20	12
Transfer 3	22	11
Transfer 4	20	12
Average Excluding Initial	20.667	11.667

Table 3: Comparative Results of 2339MB Data

	Time in	Seconds
	SSH + RSYNC	SCP
Initial Transfer	81	82
Transfer 2	21	82
Transfer 3	20	81
Transfer 4	22	82
Average Excluding Initial	21	81.667

4.2 ETL Security using SSH for Authentication and Encryption

SSH was used for both host authentication and data transfer encryption, in host authentication the public key of the client machine is generated and placed in the authorized keys file of the SSH server. This allows the client machine to authenticate against the server without the need to provide a password. Additionally, this allows for the automation of SSH-based processes because there is no need for the user to put in a password. The SSH tunnel also provides encryption of all data that is being transferred between the ERP and the warehouse. By running a packet capture on the data transferred between two machines, the data was observed to be encrypted on the packet capture tool (Wireshark) as shown in Figure 6.

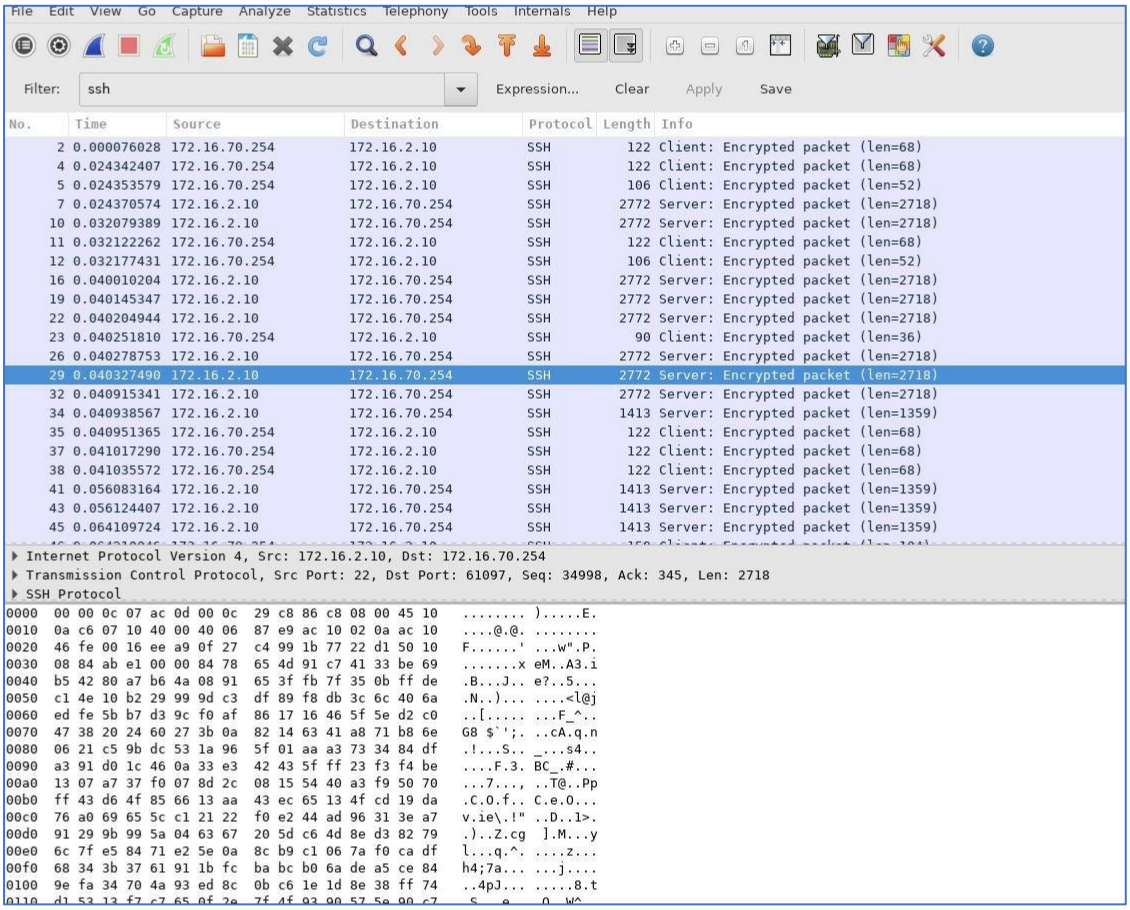


Figure 6: SSH Encrypted Packets

A similar experiment conducted over an unencrypted channel shows that the data transferred is in plain text. The captured data also contained the username and password used in the FTP captured in plain text (Figure 7). This shows that data is transferred securely in the developed data warehouse prototype where SSH protocols are used.

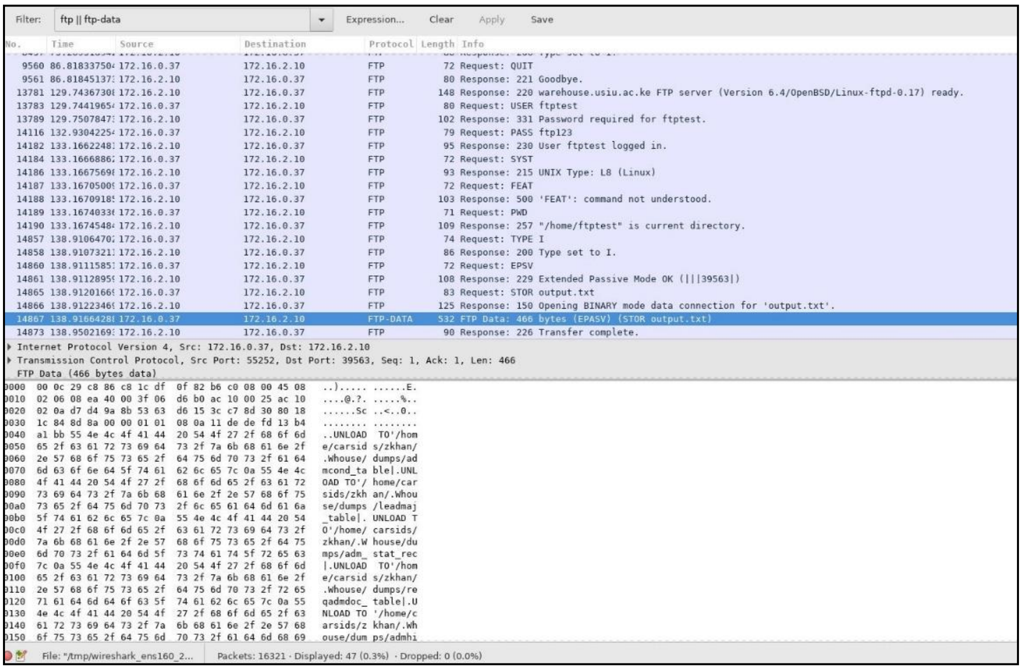
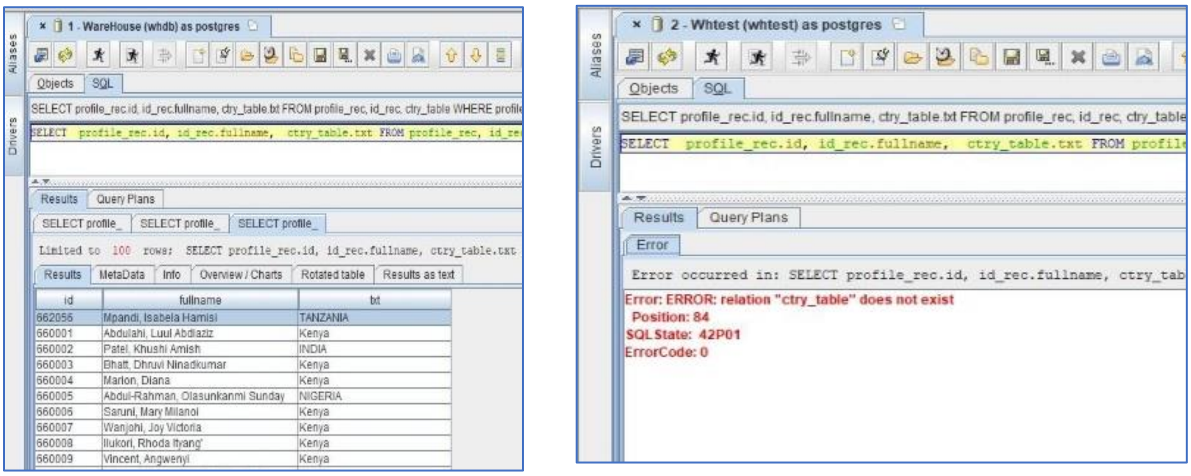


Figure 7: Unencrypted FTP Password and Data

SSH tunnel encrypts all data transferred as indicated by (Burande et al., 2014) and in this way provides Confidentiality, integrity, and authenticity of the data (Garimella and Kumar, 2015).

4.3 Data Warehouse Quality Evaluation.

To test the quality of data queries, we conducted experiments running queries that cut across several tables. For example, a query to check what country, the students were residents of during the initial creation of their profiles or a query to check the student's financial aid. To evaluate the queries, similar queries were run with the Nearest Neighbor filter removed. The results in Figure 8 show that the queries succeeded with the nearest neighbor filter and failed without it. The results indicate the high data quality of the developed model that uses the Nearest Neighbor approach. Data extracted with "data groups" imported enhanced ad-hoc queries as indicated by (Mehdi and Beheshti, 2013).



(a) With Nearest Neighbor Filter (b) Without Nearest Neighbor Filter

Figure 8: SQL Query to check the country of Residence during Admissions

5. Conclusion

This research developed a prototype to show how the quality of data warehouses performance, security, and queries can be enhanced using RSYNC, SSH, and a hybrid relational-dimensional core data warehouse model. The hybrid model was shown to combine the best of the previous models while also getting rid of the shortcomings of these traditional models. Findings from the study revealed that performance improved only where large data transfers occur when using RSYNC over SCP. The results of this study provide a valuable contribution to designers of data warehouses, which are increasingly becoming pivotal in organizations' decision-making processes. Future work in this area will explore the scalability of the solution by testing the approach on different use cases.

References

Aljawarneh, I. M. (2016). Design of a data warehouse model for decision support at higher education: A case study. *Information Development*, 32(5):1691–1706.

Ally, S. S. (2016). Data Warehouse and BI to Catalize Information Use in Health Sector for Decision Making : A Case Study. pages 92–97.

Banditvilai, W. and Boonkrong, S. (2014). A Comparison of Efficiency of Data Transfer by Using Rsync, Rsync+SSH and Dropbox. *International Journal of Computer Theory and Engineering*, 6(3):196–199.

Beig, B. M. (2011). Information availability: Components,. 2(3).

Burande, A., Pise, A., Desai, S., and Martin, Y. (2014). Wireless Network Securityby SSH Tunneling. *International Journal of Scientific and Research Publications*, 4(1):2250–3153.

Castellina, N. and Prouty, K. (2012). ERP in manufacturing 2012. The evolving ERP strategy. *Aberdeen Group*, (July):29.

Castro, D. and Mcquinn, A. (2016). Unlocking Encryption : Information Security andthe Rule of Law. *Information Technology and Innovation Foundation*, (March):1– 50.

Chaudhary, S. (2011). A Critical Review of Data Warehouse. *Global Journal of . . .* , 1(2):95–103.

Chen, E. (2012). Implementation Issues of Enterprise data Warehousing and Business Intelligence in the Healthcare Industry. *Communications of the IIMA*, 12(2):39–50.

Dammak, S., Jedidi, F. G., and Gargouri, F. (2016). Security Measures for Web ETL Processes. *Computer and Information Science*, 614.

Debbarma, N., Nath, G., and Das, H. (2013). Analysis of Data Quality and Per- formance Issues in Data Warehousing and Business Intelligence. *International Journal of Computer Applications*, 79(15):20–26.

Di Tria F, Lefons E, T. F. (2012). Research data mart in an academic system. In *In Engineering and Technology (S-CET), 2012 Spring Congress.*

El-Sappagh, S. H. A., Hendawi, A. M. A., and El Bastawissy, A. H. (2011). A proposed model for data warehouse ETL processes. *Journal of King Saud University - Computer and Information Sciences*, 23(2):91–104.

Flora, H. and Chande, S. (2014). A Systematic Study on Agile Software Development Methodologies and Practices. *International Journal of Computer Science and Information Technologies*, 5(3):3626–3637.

Flory, A., Soupirot, P., and Tchounikine, A. (2014). A Design and implementation ofa data warehouse for research administration universities . A Design and imple- mentation of a data warehouse for research administration universities. (January 2001).

Garimella, A. and Kumar, D. R. (2015). Secure shell (ssh) - its significance in network-ing. 4(3):187–196.

George, S. (2012). Inmon vs. kimball: Which approach is suitable for your data ware- house? (accessed from http://www.computerweekly.com/tip/inmon-vs-kimball- which-approach-is-suitable-for-your-data-warehouse). *Computer Weekly*.

Goldstein, P. J. and Katz, R. N. (2005). Academic Analytics : The Uses of ManagementInformation and Technology in Higher Education. *Educase*, 8:1–113.

Gour, V., Sarangdevot, S. S., Tanwar, G. S., and Sharma, A. (2010). Improve Perfor- mance of Extract, Transform and Load ({ETL)} in Data Warehouse. *International Journal on Computer Science & Engineering*, 1(3):786–789.

Ha, L. Q., Xie, J., Millington, D., and Waniss, A. (2015). Comparative Performance Analysis of PostgreSQL High Availability Database Clusters through Containment.*Ijarcce*, 4(12):526–533.

Han, J. and Micheline, K. (2006). *Data Mining: Concepts and Techniques*, volume 2.

Hultgren, H. (2012). *Modeling the Agile Data Warehouse with Data Vault (Volume 1)*. Brighton Hamilton.

IBM (2019). Ibm knowledgecenter. Retrieved on 2nd December 2019. Inmon, W. (2005). *Building the data warehouse*.

Jalil, M. (2013). Practical Guidelines for conducting research. 2013(February).

Janacek, B. (2015). Best practices: Securing data at rest, in use, and in motion. *Datamotion*, pages https://www.datamotion.com/2015/12/best–practices– securing–data–at–rest–in–use–and–in–motion/.

Kaladi, A. and Ponnusamy, P. (2012). Performance evaluation of database management systems by the analysis of dbms time and capacity. *International Journal of Modern Engineering Research (IJMER)*, 2(2):67–72.

Kaur, V. (2016). Business Intelligence And E-banking : A Study Of Bi Importance In Banking Sector. *Biz and Bytes*, 7(1, 2016):11–19.

Kavitha, P. (2013). A Survey of Data Warehouse and ETL processes. 3(1):387–390.

Kimball, R. and M.Ross (2013). *The data warehouse toolkit :The definitive guide to dimensional modeling*. John Wiley & Sons.

Kimball, R. and Ross, M. (1996). *The data warehousing toolkit*. New York: John Wiley & Sons.

Kimball, R., Ross, M., Thornthwaite, W., Mundy, J., and Becker, B. (200). The datawarehouse lifecycle toolkit.

Kvavik, R. B. and Katz, R. N. (2002). The Promise and Performance of Enterprise Systems for Higher Education. *ECAR Research Study*, 4.

Martino, A. (2014). Trivadis White Paper Comparison of Data Modeling Methods for a Core Data Warehouse. *Trivadis White Paper*, (June):21.

Mathiews., D. (2012).Data vault et bi. *–URL http://fr.slideshare.net/dlinstedt/prsentationdata-vault-et-bi- v20120508*.

Mehdi, S. and Beheshti, R. (2013). Organizing, Querying and Analyzing Ad-Hoc Processes. (September):245.

Michael, G. and Karthikeyan, R. (2017). A Research on Secure Shell (SSH) Protocol. *International Journal of Pure and Applied Mathematics*, 116(16):559–564.

Nedelcu, B. (2012). Business Intelligence Systems. *Database Systems Journal*, IV(4/2013):12–

20.

Nwokeji, J. C., Aqlan, F., Apoorva, A., and Olagunju, A. (2018). Big data ETL implementation approaches: A systematic literature review. In *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE*, volume 2018-July, pages 714–715.

Oketunji, T. and Omodara, O. (2011). Design of Data Warehouse and Business Intelligence System. Master Thesis, (June).

Peersman, G. (2014). Overview: Data Collection and Analysis Methods in Impact Evaluation. *Methodological BriefsImpact Evaluation No. 10*, 10(10).

Plechawska-W’jcik, M. and Rykowski, D. (2016). Comparison of relational, documentand graph databases in the context of the web application development. *Advances in Intelligent Systems and Computing*, 430:3–13.

Popescul, D., Alexandru, U., and Cuza, I. (2018). The Confidentiality – Integrity – Accessibility Triad into the Knowledge Security . A Reassessment from the Point of View of the Knowledge Contribution to Innovation. (May 2014).

Rahman, N. (2013). Measuring Performance for Data Warehouses-A Balanced Score-card Approach. *International Journal of Computer and Information* . . . , 04(01).

Rodzi, N. A. H. M., Othman, M. S., and Yusuf, L. M. (2016). Significance of data integration and ETL in business intelligence framework for higher education. *Proceedings - 2015 International Conference on Science in Information Technology: Big Data Spectrum for Future Information Economy, ICSITech 2015*, pages 181– 186.

Rolly Constable, Marla Cowell, S. Z. C. D. G. (2012). Ethnography, observational research, and narrative inquiry. *Colorado State University*.

Rorimpandey, G., Sangkop, F., Rantung, V., Zwart, J., Liando, O., and Mewengkang, A. (2018). Data Model Performance in Data Warehousing. *IOP Conference Series:Materials Science and Engineering*, 306(1).

Serra, F. and Marotta, A. (2016). Data Quality in Data Warehouse Systems.

Suri, P. and Sharma, M. (2011). A Comparative Study Between the Performance of Relational & Object Oriented Database in Data Warehousing. *International Journal of Database Management Systems*, 3(2):116–127.

Tanpure, A., Patil, A., Bansod, A., and Kulkarni, A. (2015). RSYNC over HTTPS for Linux and Windows with Seamless data transfer. *Citeseer*, 5(11):582–585.

Winick, S. and Bartis, P. (2016). Folklife and fieldwork: An introduction to cultural documentation. *American Folklife Center, Library of Congress*.

Yessad, L. (2016). Comparative Study of Data Warehouses Modeling Approaches: Inmon , Kimball and Data Vault. pages 95–99.