IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 2, Ver. III (Mar.-Apr. 2017), PP 56-67
www.iosrjournals.org

Security Criteria and Indicators of Rdbmss: a Comparative Study

Awad M. Awadelkarim Mohamed[1], Anass A. Nour[2]
[1]*Faculty of Computers and Information Technology University of Tabuk, Tabuk, Saudi Arabia*
[2]*College of Computer Science and Information Technology Sudan University of Science and Technology, Sudan*

Abstract: *Selection of an appropriate Database Management System (DBMS) to sustain and support a particular database system or project is considered as crucial stage in the allied DB development lifecycle. The selection process supposes undertaken prior physical design stage and based on numerous DBMS evaluation features and criteria, which in line with the given system requirements. Recently, security features raise and become a foremost selection criterion as well as an elementary system requirement. Therefore, this paper contributes to such context by conducting a comparative study intended for the security criteria and indicators of the most three famed and widely used Relational DBMSs, namely Oracle, MS SQL Server, and MySQL. The study proposes and formulates security evaluation features derived from the standard criteria in order to accomplish such appraisal. The result of the study classifies and grades the three chosen RDBMSs consistent with the developed security evaluation criteria, which ranks Oracle on the topmost.*

Keywords: *Database security, DBMSs, RDBMSs, Oracle Security, MS SQL Server Security, MySQL Security*.

I. Introduction

A database represents an essential corporate resource that should be properly secured using appropriate controls. This need for security, while often having been neglected or overlooked in the past, is now increasingly recognized by organizations. The reason for this turnaround is the increasing amounts of crucial corporate data being stored on computer and the acceptance that any loss or unavailability of this data could prove to be disastrous [1]. The Database Management System (DBMS) is a collection of programs that enables users to store, modify, manipulate, and extract data or information from a database [2]. Similarly, database security may be defined as the mechanisms that protect the database against intentional or accidental threats [1]. Moreover, Database security may be defined as the system, processes, and procedures that protect a database from unintended activity such as data lost, authenticated misuse, malicious attacks or inadvertent mistakes made by authorized individuals or processes. Database security provides many layers and types of information security, including access control, auditing, authentication, Encryption, and Integrity controls [3]:

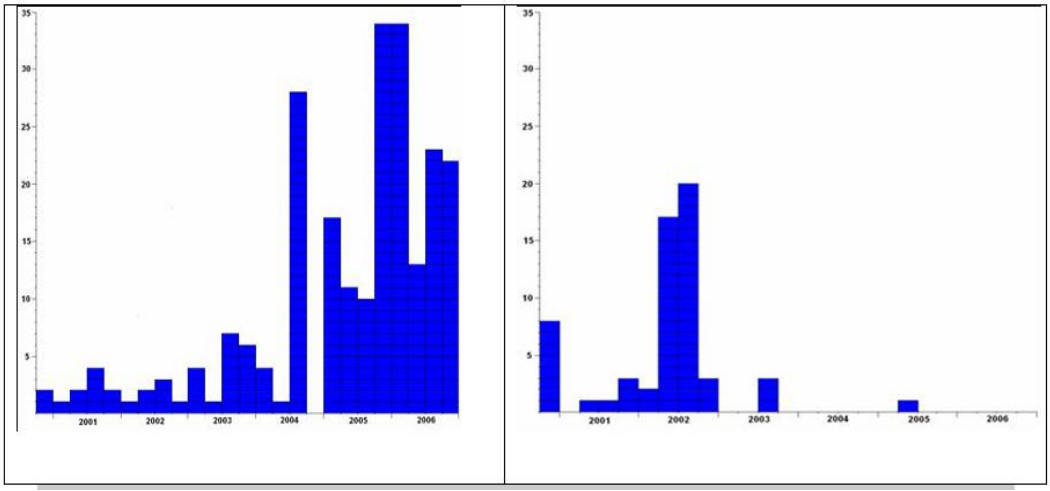
- **Access control** is the ability to permit or deny the use of a particular resource by a particular entity. Access control mechanisms can be used in managing physical resources, logical resources, or digital resources. Access control techniques are sometimes categorized as either discretionary or non-discretionary. The three most widely recognized models are Discretionary Access Control (DAC), Mandatory or Label-Based Access Control (MAC), and Role Based Access Control (RBAC). MAC and RBAC are both non-discretionary [3].
- **Auditing** is ability to trace access to sensitive or important information stored on computer systems (databases), as well as access to the computer systems themselves. Auditing is the analysis of log records to present information about the system in a clear and understandable manner. With respect to computer security, logs provide a mechanism for analyzing the system security state, either to determine if a requested action will put the system in a non-secure state or to determine the sequence of events leading to the system being in a non-secure (compromised) state [4].
- **Authentication** is the act of establishing or confirming something (or someone) as authentic, specifically, that claims made by or about the subject are true. This might involve confirming the identity of a person, the origins of an object/subject, or assuring that a computer program is a trusted one.
- **Encryption** is the process of transforming information using an algorithm to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. The result of the process is encrypted information (in cryptography, referred to as cipher text). In many contexts, the word encryption also implicitly refers to the reverse process, decryption [3].
- **Data integrity** is a term that can mean ensuring data is "whole" or complete, the condition in which data are identically maintained during any operation (such as transfer, storage or retrieval), the preservation of data for their intended use, or, relative to specified operations, the a priori expectation of data quality. Put simply, data integrity is the assurance that data is consistent and corrects [5].

Therefore, the offered security features by DBMSs raise and become a foremost selection criterion. However, there is a huge variation and diversity concerning the security features provided by the available RDBMSs now in the market, in addition to the revolution of the schemes employed to enforce such features. Consequently, the wise and prudent selection decision becomes problematical and challenging. Moreover, according to the best of our knowledge, *there is no single impartiality scientific comparative study in such context*, primarily for the leading DBMSs that dominate the market. Accordingly, this paper presents and provides an in-depth comparative assessment intended for the security perception and features of the most three famed and widely used Relational DBMSs, specifically Oracle Database 11g, Microsoft SQL 2008, and MySQL 5.1. The investigation proposes and formulates security evaluation characteristics derived from the standard criteria in order to accomplish such appraisal. Moreover, the indicated versions of the chosen DBMSs are decided based on several factors and influences, such as: the settings availability and licenses in our workbench at our research lab, second their widely usage and dominance in the associated surroundings, as well as they cover and involve all required essential and substantial security features that typically found in the latest versions such Oracle 12c and Microsoft SQL 2016.

The rest of the paper is organized as follows: section 2 presents the related work. Section 3 elucidates the proposed formulated security evaluation criteria. Section 4 demonstrates the conducted planned comparative study. Section 5 reveals the summary and grasp of the obtained findings. Section 6 and 7 provide a conclusion of this paper and the research limitations respectively. Part 8 suggests issues could be done in the future research.

II. Related Work

In actual fact, the previous related work is very infrequent, and as mentioned earlier, according to the best of our knowledge, there is no *single impartiality scientific comparative investigation* in such context, primarily for the leading DBMSs that dominate the market. Therefore, this section reviews the lone-relevant study that identified as a documented report: David Litchfield [6] has examined the differences between the security posture of Microsoft's SQL Server and Oracle's RDBMSs based upon faults reported by external security researchers. Only flaws affecting the database server software itself have been considered in compiling this data so issues that affect, for example, Oracle Application Server have not been included. A general comparison is made covering Oracle 8, 9 and 10 against SQL Server 7, 2000 and 2005.



Oracle Ms **Sql server**

Figure 1: Oracle (8, 9, and 10) faults vs. Microsoft SQL Server (7, 2000, and 2005) faults [6]

The numbers at the x-axis represent the years, while the numbers at the y-axis represent security faults. Thus, **Figure 1** shows the number of security flaws in the Oracle and Microsoft database servers that have been discovered and fixed since December 2000 until November 2006. Also, **Figure 2** illustrates the documented flaws for Oracle 10g contrast to Microsoft SQL server 2005.

Thus, the conclusion of such concerned study is immediately apparent from these four graphs that Microsoft SQL Server has stronger security posture than the Oracle RDBMS in term of security faults. Hence, and as stated by David Litchfield [6] *" if security robustness and a high degree of assurance are concerns when looking to purchase database server software – given these results one should not be looking at Oracle as a serious contender*".

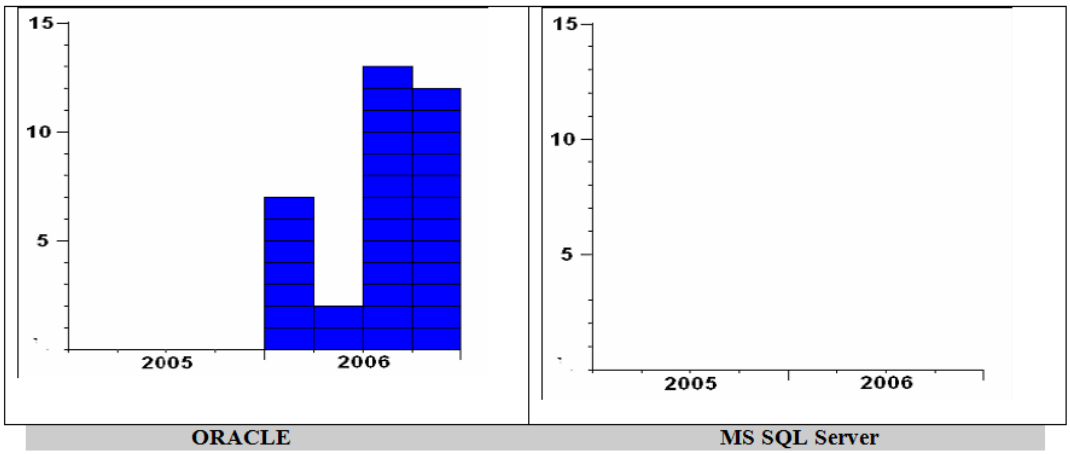


Figure 2: Oracle 10g faults vs. Microsoft SQL server 2005 faults [6]

III. The Proposed Formulated Security Evaluation Criteria:

With the large readily available number of DBMSs; and as each DBMS product has its own security features and characteristics that distinguish it from others, likewise the granularity of each security feature of each DBMS is differ from one to another. Therefore, the following clauses present and discuss the proposed developed *(i.e. well-formulated based on the associated security requirements and standards)* security evaluation criteria and features that will be used and applied to conduct the planned comparative study amidst the selected RDBMSs:

3.1 High Availability

Availability is the degree to which an application, service, or functionality is available upon user demand. High availability is a system design protocol and associated implementation that ensures a certain absolute degree of operational continuity during a given measurement period. One challenge in designing a high availability IT infrastructure is examining and addressing all possible causes of downtime. Downtime can be classified into two primary categories: unplanned and planned [7]. The database high availability responsible from failures with the database and address these failures. There are some types of failures such as: System Failures, Data Failures, Disaster Recovery, Human Errors, System Maintenance, and Data Maintenance [7] [8].

3.2 Access Control

Access control mechanisms are necessary and crucial design element to any secure application. Mostly, applications should protect front-end and back-end data and system resources by implementing access control restrictions on what users can do, which resources they have access to, and what functions they are allowed to perform on the data. Ideally, an access control scheme should protect against the unauthorized viewing, modification, or copying of data. Additionally, access control mechanisms can also help limit malicious code execution, or unauthorized actions through an attacker [7] [9] [10] [11]. Examples of most common mechanisms/criteria utilized in DBMSs that support access control:

- **Virtual Private Databases** (VPD) is the combination of fine-grained access control and application context. VPD combines these two features, enabling user to enforce security policies to control access at the row level, based on application or session attributes [11] [12].
- **A View** is a presentation of data selected from one or more tables (possibly including other views). In addition to showing the selected data, a view also shows the structure of the underlying tables, and can be thought of as the result of a stored query [11] [12].
- **Authentication** means verifying the identity of someone (a user, device, or other entity) who wants to use data, resources, or applications. Validating that identity establishes a trust relationship for further interactions [11] [12].

3.3 Auditing

Auditing is the monitoring and recording of selected user database actions. It can be based on individual actions, such as the type of SQL statement run, or on combinations of factors that can include name, application, time, and so on. Auditing helps you to track unauthorized user behavior on your systems and stop it. Auditing is especially useful to protect against rogue administrators or users with elevated privileges [11] [12]. Actually, various gradations of auditing are provided by DBMSs.

3.4 Encryption

Encryption is the process of making data unrecognizable to people who do not have the proper keys to read it. You have two key factors to worry about when dealing with data from a database: sending data over a network and storing data in the database [11] [12]. However, protecting databases using encryption is ruled by many constraints and conditions, there is a numerous cryptographic algorithms are equipped by DBMSs, for instances: Data Encryption Standard (DES) Algorithm, Triple-DES, Advanced Encryption Standard (AES), RSA, and RC4. Moreover, Encryption of network data provides data privacy so that unauthorized parties are not able to view plaintext data as it passes over the network.

3.5 Data Integrity

Data Integrity is the quality of correctness, completeness, wholeness, soundness and compliance with the intention of the creators of the data. Thus, it generally refers to the validity of data, algorithms examples such as MD5 and SHA-1 [13]. Consequently, **Table 1** summarizes the formulated/selected security evaluation criteria/features, along with the associated key mechanisms and indicators. Consistent with the entire number of the subedit criteria and indicators, and to artlessly compute and appraise the findings, the recommended utmost point/weight is 6, where a unit weight is 0.5 point.

Table 1: The Proposed Formulated Security Evaluation Criteria

Criteria	Indicators/Mechanisms	Max Points
High Availability	System Failures	6
	Data Failures	6
	Disaster Recovery	6
	Human Errors	6
	System Maintenance	6
	Data Maintenance	6
Access Control	Virtual Private Databases (VPD)	6
	Views	6
	Roles	6
	Privileges	6
	Authentication	6
Data Encryption	Advanced Encryption Standard (AES)	6
	Data Encryption Standard (DES)/ DES40	6
	Triple DES	6
	RC4	6
	SHA-1 or Cryptographic Hash/(MAC)	6
Data Integrity	MD 5	6
	SHA-1	6

IV. The Accomplished Comparative Study

This section demonstrates the conducted planned comparative study for the indicated versions of the chosen DBMSs namely: Oracle Database 11g, Microsoft SQL 2008, and MySQL 5.1., and based on the designed key indicators of the formulated security evaluation criteria:

4.1. High Availability Criterion

Any organization evaluating a database solution for enterprise data must also evaluate the High Availability (HA) capabilities of the database. Data is one of the most critical business assets of an organization. If this data is not available and/or not protected, companies may stand to lose millions of dollars in business downtime plus negative publicity [14].

4.1.1 Addressing System Failures: *(An Indicator)*

System failures are the result of hardware failures, power failures, and operating system or server crashes. The challenges with system failures lie in ensuring fast recovery, or better still, a higher level of fault tolerance.

As shown in **Table 2**, Oracle provides an array of features that clearly differentiate Oracle from SQL Server and MySQL in terms of how effectively it addresses system failures.

Table 2: System Failures Indicator

System Failures benchmark	Oracle	SQL Server	MYSQL
Active-active clustering	Supported√	Not Supported×	Not Supported×
Transparent application scalability	Supported√	Not Supported×	Not Supported×
Dynamic addition/removal of nodes with no effects on data distribution	Supported√	Not Supported×	Not Supported×
Integrated cluster-ware that supports all major OS	Supported√	Not Supported×	Not Supported×

The Findings Discussion: with reference to [12] [14] [15], we identified and specified the followings:
The cornerstone of Oracle's high availability solutions that protects from system failures is Oracle Real Application Clusters (RAC). Oracle RAC is a cluster database with a shared cache architecture that overcomes the limitations of traditional shared-nothing and shared-disk approaches, to provide a highly scalable and available database solution for all business applications. In a RAC configuration, all nodes are active and serve production workload. If a node in the cluster fails, the Oracle Database continues running on the remaining nodes. Individual nodes can also be shutdown for maintenance while application users continue to work.
SQL Server has no solution equivalent to RAC. The SQL Server architecture is based on a Federated Database model, which is a collection of independent servers connected over a network. Data is horizontally partitioned across each participating server, and applications see a logical view of the data through UNION ALL views and distributed SQL, using a technology called Distributed Partitioned Views (DPVs). This model leads to complexities in the areas of data partitioning (to avoid "hot nodes"), adding/removing nodes, and dealing with node failures. It may be noted that to protect from server failures, Microsoft suggests using SQL Server with Microsoft Cluster Service (MSCS), in a Failover Clustering model. However, in this model, a particular SQL Server instance runs in only one node, while the other "backup" node remains in a passive state, waiting for the failover to occur.

MySQL Cluster uses the normal MySQL Server technology paired with a new storage engine NDB Cluster. Data within MySQL Cluster is synchronously replicated among the data nodes in the cluster. MySQL Cluster uses the shared-nothing architecture, data nodes in the cluster handle their own storage and the only means of communication between the nodes is through messages.

4.1.2. Addressing Data Failures: ***(An Indicator)***

A system or network fault may prevent users from accessing data, but media failures without proper backups can lead to lost data that cannot be recovered. Thus, **Table 3** illustrates mechanisms that evidently compare and assess the Data Failures Indicator amongst three chosen RDBMSs.

Table 3: Data Failures Indicator

Data Failures benchmark	Oracle	SQL Server	MYSQL
Built-in database failure detection, analysis, and repair	Supported√	Not Supported×	Not Supported×
Automated disk backup management	Supported√	Not Supported×	Not Supported×
Incrementally updated backup strategy	Supported√	Not Supported×	Not Supported×
Parallelize backup within a single file	Supported√	Not Supported×	Not Supported×
Unused block compression during full backup	Supported√	Not Supported×	Not Supported×
Automatic data file creation during recovery	Supported√	Not Supported×	Not Supported×
Automatic restore failover to next available backup during recovery	Supported√	Not Supported×	Not Supported×

The Findings Discussion: with reference to [12] [14] [15], we identified and specified the followings:

- ***Built-in Database Failure Detection, Analysis, and Repair:***
- ***Automated disk backup management:***

When faced with data failures, a DBA first invests time to diagnose the issues and plan an appropriate recovery strategy. Depending on the nature of the failure, this investigation and planning time can often comprise a large percentage of the total recovery time. Available with Oracle Database 11g and above, the Data Recovery Advisor (DRA) dramatically reduces this time by automatically detecting failures in real-time (e.g. block corruptions, missing files), reporting failure analysis results, and generating a feasible recovery strategy (e.g. RMAN recovery script) that can be run as-is or customized for running at a later time. In addition, regularly scheduled Data Integrity Checks allow proactive monitoring of database integrity, thereby catching and repairing data issues before users even come across them. SQL Server uses (SQL Server File Group Restore and SQL Server Fast Recovery) to allow for easily restoring just the objects that have been corrupted, and improves data availability by allowing users to reconnect to a recovering database as soon as the transaction log has been rolled forward. The most popular method used to backup and recovery a MySQL database is the (mysqldump and mysqlhotcopy), which ships with every version of MySQL. The mysqldump utility creates a backup file for one or more MySQL databases that consists of DDL/DML statements needed to recreate the specified databases along with their data. To restore a particular database, the backup file is simply read back into the MySQL utility command prompt as an input file. Thus, SQL Server and MySQL have no such intelligent, database-aware diagnosis and recovery tools and continue to rely on manual restore, recovery, and data verification procedures.

- ***Incrementally updated backup strategy:***

With fast incrementally updated backups, RMAN rolls forward an image copy by applying incremental backups. The image copy is updated with block changes up through the SCN at which the latest incremental backup was taken. Incrementally updated backups eliminate the need and overhead of performing a full database backup every day. SQL Server and MYSQL do not offer and support such facility.

- ***Parallelize backup within a single file:***

RMAN can back up or restore a single file in parallel by dividing the work among multiple channels. Each channel backs up one file section, which is a contiguous range of blocks. SQL Server and MySQL do not offer a comparable capability.

- ***Automatic data file creation during recovery:***

During a restore, when RMAN finds corruption in a backup, or finds that a backup cannot be accessed, RMAN tries to restore the file from all known backups before returning an error. SQL Server and MYSQL lack this capability too.

4.1.3. Addressing Disaster Recovery: *(An Indicator)*

In order to support the disaster recovery, Oracle provides Oracle data guard, SQL Server provides both Microsoft Database Mirroring and Microsoft Log Shipping, while MySQL offers MySQL DRBD (Distributed Replicated Block Device). Nevertheless, **Table 4** summarizes and appraises the three chosen RDBMSs based on the designated Disaster Recovery Benchmarks.

Table 4: Disaster Recovery Indicator

Disaster Recovery Benchmark	Oracle	SQL Server	MYSQL
Multiple standbys for non-stop protection after failover	Supported√	Not Supported×	Not Supported×
No performance impact while creating standby databases	Supported√	Not Supported×	Not Supported×
Standby apply process failure does not impact primary database or transmission of changes	Supported√	Not Supported×	Not Supported×
Pausing data transmission does not cause the primary database to stall	Supported√	Not Supported×	Not Supported×
Support for a number of mixed primary/standby configurations	Supported√	Not Supported×	Not Supported×

4.1.4 Addressing Human Errors: *(An Indicator)*

Another leading cause of data failure and application downtime is human error, which may be due to accidents (e.g. deleting important data) or even sabotage. As shown in **Table 5**, Oracle provides clearly differentiating capabilities compared to SQL Server and MySQL in terms of how effectively it addresses human error circumstances.

Table 5: Human Errors Indicator

Human Errors Benchmark	Oracle	SQL Server	MYSQL
Retrieve data from the past using SQL queries	Supported√	Not Supported×	Not Supported×
Support Recycle Bin	Supported√	Not Supported×	Not Supported×
Examine and back-out changes to the database at the transaction level	Supported√	Not Supported×	Not Supported×
View changes across row versions	Supported√	Not Supported×	Not Supported×
Flashback a table to a point in time in the past	Supported√	Not Supported×	Not Supported×
Flashback the database to a prior point in time without restoring a backup	Supported√	Not Supported×	Not Supported×

The Findings Discussion: with reference to [12] [14] [15], we identified and specified the followings:
Oracle Flashback technologies provide point-in-time viewing and quick recovery at the row, transaction, table, and database level. Additionally, Oracle flashback technologies contains: Oracle Flashback Query, Oracle Flashback Version Query, Oracle Flashback Transaction Query, Oracle Flashback Table, and Oracle Flashback Drop. SQL Server has no capability similar to Oracle Flashback technologies, but it use SQL Server Recovery, SQL Server File Group Restore which allow for easily restoring just the objects that have been corrupted. SQL Server Database Snapshots SQL Server includes database snapshots that allow quick and easy restoration of damaged data. While, MySQL uses Mysqlhotcopy, MySqldump utility creates a backup file for one or more MySQL databases that consists of DDL/DML statements needed to recreate the specified databases along with their data.

4.1.5 Addressing System Maintenance/Maintainability: *(An Indicator)*

As business needs change, system changes may also be required. For example, business growth often entails growth in data processing volume. This may translate into a requirement for additional processing power through hardware upgrades of disks, memory, CPUs, nodes in a cluster, or entire systems. Oracle is unique in the ability to change any system resource dynamically, as proven in **Table 6**.

Table 6: System Maintenance Indicator

System Maintenance Benchmark	Oracle	SQL Server	MYSQL
Add a node to a cluster online	Supported√	Not Supported×	Not Supported×
Add or drop disks online	Supported√	Not Supported×	Not Supported×

The Findings Discussion: with reference to [12] [14] [15], we identified and specified the followings:

- ***Adding a Cluster Node online:***

Data partitioning in a shared-nothing environment makes adding new servers to a cluster time consuming and costly, because redistribution of partitioned data according to the new partitioning map is required. Here's what a DBA or System Administrator has to do to add a node to a MySQL or SQL Server database that operates in a Federated model to support scale-out:

- Add hardware
- Configure a new partition (set partition-specific parameters, etc.)
- Restart the database (i.e. shut down and restart all nodes)
- Re-distribute the data to spread it across a larger number of partitions

On the other hand, just the following management tasks are needed when a node is added to Oracle-RAC:

- Add hardware
- Configure new instance (set instance-specific parameters, etc.)

Thus, that's it, no data re-partitioning, no offline maintenance, and no database restart; just a seamless scale-out. The RAC allows nodes to be added without interrupting database access.

- ***Adding or Dropping Disks Online:***

With Oracle ASM (Automatic Storage Management), it is possible to add disks to, or drop disks from the disk group that the Oracle database is actively using, without causing any downtime to the database. ASM automatically rebalances a disk group whenever disks are added or dropped, ensuring that database files are spread evenly across all disks in a disk group. This means that administrators do not need to search for hot-spots in a disk group and manually move data around to restore a balanced I/O load. SQL Server does not have any such integrated capability – for example, it has to rely on the underlying platform support (e.g. Microsoft Windows Server –based hardware) for hot swapping of storage drives.

4.1.6 Addressing Data Maintenance: ***(An Indicator)***

Maintaining, re-defining and transforming the data that supports a business is a critical activity for any DBA – this may be required unexpectedly with new business conditions, or this may even be a regularly scheduled activity. **Table 7** demonstrates the differentiation amongst three chosen RDBMSs with regards to the Data Maintenance Benchmark.

Table 7: Data Maintenance Indicator

Data Maintenance Benchmark	Oracle	SQL Server	MYSQL
Online add, drop, exchange, move partitions	Supported√	Not Supported×	Not Supported×
Online reorganization of individual tables, including relocating table to a different tablespace	Supported√	Not Supported×	Not Supported×
Online reorganization of individual table partitions	Supported√	Not Supported×	Not Supported×
Extensive online table redefinition capabilities, including data transformations	Supported√	Not Supported×	Not Supported×
Fast online add column, with default value	Supported√	Not Supported×	Not Supported×
Online rename and merge columns	Supported√	Not Supported×	Not Supported×
Invisible indexes	Supported√	Not Supported×	Not Supported×
Online add/modify constraint, add column, index create/rebuild do not require exclusive lock	Supported√	Not Supported×	Not Supported×

The Findings Discussion: with reference to [12] [14] [15], we identified and specified the followings:

Oracle offers a wide range of online index and table reorganization operations, from the ALTER INDEX and ALTER TABLE commands to management of more complex reorganization tasks via Online Redefinition. In particular, Oracle's unique Online Redefinition capability allows one to:

- Modify the storage parameters of a table
- Move a table to a different tablespace
- Add, modify, or drop one or more columns in a table
- Add or drop partitioning support
- Change partition structure
- Change physical properties of a single table partition, including moving it to a different tablespace in the same schema
- Add support for parallel queries
- Re-create a table to reduce fragmentation

SQL Server and MySQL cannot perform online table and partition redefinition, including even simple changes to tables, nor online add/drop/exchange/move partition operations.

- ***Fast Online Add Column With default value:***

Oracle, adding new columns with DEFAULT value and NOT NULL constraint does not require the default value to be stored in all existing records. Instead, default values of columns are simply maintained in the data dictionary. This not only enables a schema modification in sub-seconds and independent of the existing data volume, it also consumes virtually no space. SQL Server and MYSQL did not offer online add column.

- ***Online No-Lock Index:***

Creation and Rebuild Oracle's online index and rebuild operations do not use exclusive locks at any time during the operation. This means that ongoing DML (i.e. update, insert, delete) operations on the table work transparently and do not wait for the index operations to finish, thereby minimizing the drops and spikes in system usage that can occur with locks/waits. SQL Server's 'online' index creation and rebuild, in fact, requires exclusive locks during the preparation and finish stages of the index operation, so there are two periods of time where concurrent user activity can halt. Thus, SQL Server's use of the term 'online' is inaccurate.

- ***Invisible Indexes:***

An Oracle invisible index is an alternative to making an index unusable or dropping it. An invisible index is maintained for any DML operation, but is not used by the optimizer unless the index is explicitly specified with a hint. Invisible indexes have great uses in application development and testing. Applications often have to be modified without being able to bring the complete application offline. Invisible indexes enable you to leverage temporary index structures for certain operations or modules of an application without affecting the overall application. Furthermore, invisible indexes can be used to test the removal of an index without dropping it right away, thus enabling a grace period for testing in production environments. SQL Server has no such equivalent index capabilities. In summary, and as above clauses verified and evidenced, Oracle provides an integrated set of High Availability (HA) capabilities. These capabilities take care of most scenarios that might lead to data unavailability, such as system failures, data failures, disasters, human errors, system maintenance operations and data maintenance operations. Microsoft SQL Server and MySQL database provides rudimentary functionality for Data High Availability. The summary demonstrated in **Figure 3** below based on the designed weight/point system.

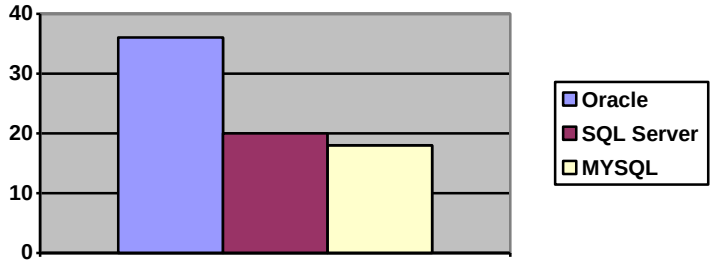


Figure 3: High Availability Criterion: The Findings Summary

4.2. Access Control Criterion

In general, access control refers to mechanisms and policies that restrict access to computer resources. DBMSs principally and chiefly concern about *(i.e. manage and control)* logical access control and the associated mechanisms and polices. Thus, **Table 8** shows the comparison amongst three chosen RDBMSs with regards to the Access Control Benchmark, while **Figure 4** exhibits the allied findings summary based on the designed weight/point system.

Table 8: Access Control Criterion

Access Control Indicators/Benchmark	Oracle	SQL Server	MYSQL
Virtual Private Database (VPD)	Supported√	Not Supported×	Not Supported×
Privilege	Supported√	Supported√	Supported√
Views	Supported√	Supported√	Supported√
Roles	Supported√	Supported√	Supported√
Authentication	Supported√	Supported√	Supported√

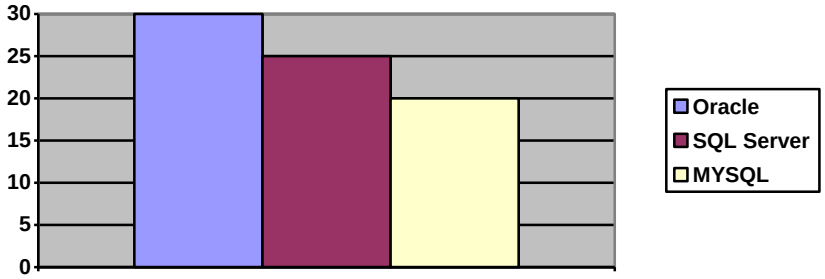


Figure 4: Access Control Criterion: The Findings Summary

In short, and as verified and evidenced above, there are no big distinctions amid Oracle, SQL Server, and MySQL regarding the data access control criterion/domain, since the three DBMSs relatively use the same techniques and elements to prevents unauthorized access to data, although, Oracle surpasses by providing the VPD apparatus.

4.3 Auditing Criterion

As elucidated in section 3.3 above, Auditing is the monitoring and recording of selected user database actions. Thus, **Table 9** articulate**s** the techniques offered by the three specified RDBMSs in order to support and provide such auditing assignments. Furthermore, the evaluation of the coverage and compliance of such provided techniques with the targeted auditing assignments is shown in **Figure 5** based on the designed weight/point system.

Table 9: Auditing Criterion: Auditing Techniques offered by the three Specified RDBMSs

Oracle	SQL Server	MYSQL
Statement auditing	Windows Security Event Log	Trigger
Privilege auditing	SQL Profiler	
Schema Object Auditing	SQL Trace	
Fine-Grained Auditing	data definition language (DDL) trigger	

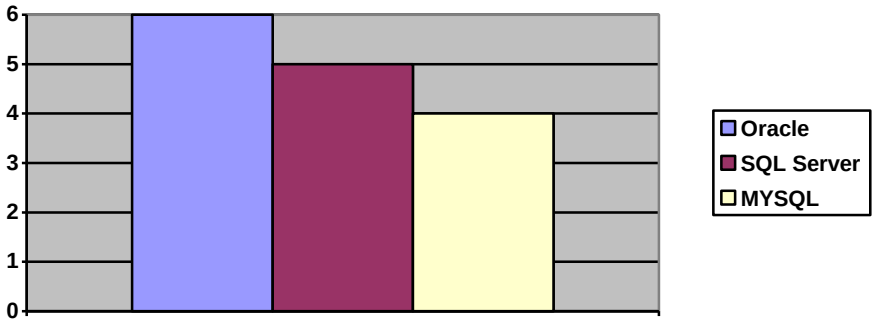


Figure 5: Auditing Criterion: The Findings Summary

In short, and with reference to [12] [14] [15] and the findings summary above (*i.e. in Table 9 and Figure 5*), we identified and specified that there is no big differences between the three specified RDBMSs concerning the compliance with the assignments of the auditing criterion/domain. However, in practical terms, Oracle excels by providing Fine-Grained Auditing (FGA) mechanism.

4.4. Encryption Criterion

As elucidated in section 3.4 above, Encryption is the process of transforming information *(referred to as plaintext)* using an algorithm *(called cipher)* to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. Thus, **Table 10** exhibits the associated algorithms tendered by the three specified RDBMSs**.**

Table 10: Encryption Criterion

Encryption Algorithms Indicators/Benchmark	Oracle	SQL Server	MYSQL
Advanced Encryption Standard (AES)	Supported√	Supported√	Supported√
Data Encryption Standard (DES)	Supported√	Supported√	Supported√
RC4	Supported√	Supported√	Not Supported×
SHA-1 Cryptographic Hash	Supported√	Not Supported×	Supported√

4.5. Data Integrity Criterion

As clarified in section 3.5 above, Data Integrity is the quality of correctness, completeness, wholeness, soundness and compliance with the intention of the creators of the data, also refers to the validity of data. Thus, **Table 11** reveals the associated algorithms offered by the three specified RDBMSs**.**

Table 11: Data Integrity Criterion

Data Integrity Algorithms Indicators/Benchmark	Oracle	SQL Server	MYSQL
Message Digest 5 (MD5)	Supported√	Supported√	Supported√
Hash Algorithm (SHA-1)	Supported√	Not Supported×	Supported√

V. The Findings: Summary And Grasp

With regards to the detailed assessment and evaluation conferred in section 4 above, **Table 12** summarizes the mechanisms, algorithms, and indicators offered by the three specified RDBMSs in order to compliance and fulfill the designed criteria. Moreover, the partial and overall totals based on the designed weight/point system are handed in the same table.

Table 12: Database security features Comparison

Criteria	Oracle	SQL Server	MYSQL
High Availability	Real application Clusters	N-Way Clustering	My SQL Cluster
	Data Guard	Database Mirroring Log Shipping	DRBD
	Oracle Flashback	Fast Recovery	MySqldump
	Flashback Query	Database Snapshots File Group Restore	Mysqlhotcopy/OS Backup
	Flashback Version Query	Database Replication	**None**
	Flashback Transaction Query	**None**	**None**
	Flashback Drop	**None**	**None**
Total	52	36	30
Access Control	Virtual Private Databases(VPD)	Not Support	Not Support
	Privileges	Privileges	Privileges
	Views	Views	Views
	Roles	Roles	Roles
Total	30	25	20
Auditing	Statement Auditing	Windows Security Event Log	Trigger
	Privilege Auditing	SQL Profiler	**None**
	Schema Object Auditing	SQL Trace	**None**
	Fine-Grained Auditing	Data Definition Language (DDL) Triggers	**None**
Total	6	5	4
Authentication	Authentication by the Operating System	Windows Authentication	**None**
	Authentication by the Network	SQL Server Authentication	**None**
	Authentication of Database Administrators	Mixed-Mode Authentication	**None**
	Authentication by Oracle DBMS	**None**	**None**
Encryption	Advanced Encryption Standard (AES)	Advanced Encryption Standard (AES)	Advanced Encryption Standard (AES)
	Data Encryption Standard (DES)	Data Encryption Standard (DES)	Data Encryption Standard (DES)
	Triple DES	Triple DES	Triple DES
	DES40	DES40	DES40
	RC4	RC4	**None**
	SHA-1 Cryptographic Hash	**None**	**None**
	SHA-1 Message Authentication Code (MAC)	**None**	SHA-1 (MAC)
Total	24	18	18
Data Integrity	-MD5	-MD5	-MD5
	-SHA-1		-SHA-1
Total	12	6	12
The Overall Total	124	90	84

Table 13: The Findings Summary: by Percentage

Criterion	Oracle	SQL Server	MYSQL
High Availability	100%↑	85% →	75% ↓
Access Control	100%↑	83%→	55% ↓
Auditing	100%↑	83%→	66% ↓
Encryption	100%↑	75%→	75% →
Data Integrity	100%↑	50%↓	100% ↑
Total	100%↑	53%↓	52%↓

Accordingly, and consistent with the designed weight/point system, Oracle accomplished 124 Points, SQL Server obtained 90 Points, while MySQL gained 84 Points, this is illustrated in **Figure 6**.

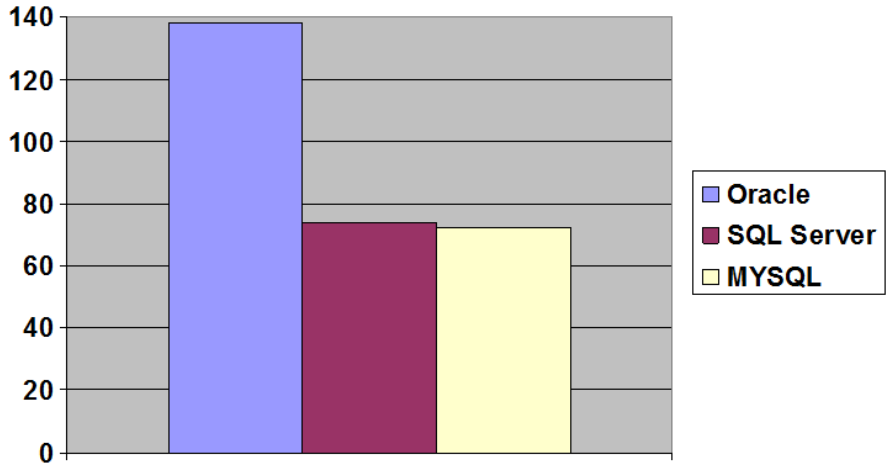


Figure 6: The Findings Summary by Points: (Based on the Designed Weight/Point System)

VI. Conclusion

When taken security into consideration, picking of the right DBMS to conserve a given system is a strategic decision in the allied development lifecycle. This study have contributed to such context by conducting the proposed comparative study for the most three famed and widely used Relational DBMSs, namely Oracle, MS SQL Server, and MySQL, and based on the designed security criteria and indicators. The research have proposed and formulated security evaluation features derived from the standard criteria in order to accomplish the intended assessment. The result of the study has classified and graded the three chosen RDBMSs according to the developed security evaluation criteria, which ranks Oracle on the topmost. The comparative study have confirmed that Oracle provides comprehensive, unique, powerful, and simple-to-use capabilities that protect businesses against unauthorized users, system faults, data corruption, disasters, human errors and so forth. Oracle offers a well-integrated database security and high availability solution stack that comprised of components such as virtual private database, fine grained auditing, RAC, Data Guard, Streams, RMAN, Flashback. In contrast, SQL Server and MySQL offers a basic set of database security features and lacks the completeness and depth of database security functionality required by most businesses today.

VII. Limitations

However, the proposed comparative study have conducted based on the standard security evaluation criteria, there are additional decisive factors have not taken into account. For instance, the reported security breaches, vulnerability incidents, and survey findings or upshot for the chosen RDBMSs. However, such factors are strategic; their influence is trivial to the overall evaluation due to the autonomous implementation.

VIII. Future Work

Although, this paper is derived and deduced initially from our prolonged research in [16], still there are two dimensions open for future research, first: considering the additional strategic security factors, and lastly: accomplishing the other *(i.e. non-security based)* evaluation criteria such as transaction handling, scalability, cost, vender support and stability. Furthermore, an advanced empirical trail for such descriptive study and statistics can be carried out as well.

References

[1]. Connolly, T. and Begg, C. **2015**. Database Systems: A Practical Approach to Design, Implementation and Management. 6th ed. Harlow, UK: © ***Pearson Education Limited***.

[2]. DISA, 2004. Database Security Technical Implementation Guide. Version 7, Release 1. DISA *"Defense Information System Agency"*, 29 October **2004**.

[3]. "Database Security". **2016**. *en.wikipedia.org* Retrieved 2016-07-07.

[4]. Matt Bishop. "What is Computer Security", ***Pages 67-69 IEEE Security & Privacy*** Vol. 99(1) 2003.

[5]. Paul Beynon-Davies, 2004. Database Systems. 3rd ed. London, UK, © ***Palgrave Macmillan***.

[6]. David Litchfield. **2006**. *"Which database is more secure? Oracle vs. Microsoft"*, NGS Software Insight Security Research (NISR) Publication. © ***Next Generation Security Software Ltd***.

[7]. Information Technology Security Evaluation Criteria (ITSEC), Luxembourg: Office for Official Publications of the European Communities, 1991 ISBN 92-826-3004-8, Printed in Germany.

[8]. Michael Gehrke, Andreas Pfitzmann, and Kai Rannenberg. 1992. "*Information Technology Security Evaluation Criteria (ITSEC) –a Contribution to Vulnerability?"* Education and Society, R. Aiken (ed.), Proc. 12th IFIP World Computer Congress, Information Processing 92, ***Vol. II, Elsevier Science Publishers*** B.V. (North-Holland).

[9]. Surajit Chaudhuri, Raghav Kaushik, and Ravi Ramamurthy. **2011.***"Database Access Control & Privacy: Is There A Common Ground?* 5th Biennial Conference on Innovative Data Systems Research (CIDR '11) January 9-12, 2011, ***Asilomar, California, USA***.

[10]. Access control and Authorization.**2017.***cgisecurity.com* Retrieved 27-1-2017 at: http://www.cgisecurity. com/owasp/ html/ch08.html
[11]. Sohail IMRAN and Irfan Hyder. **2009** *"Security Issues in Databases"*. 2009 Second International Conference on Future Information Technology and Management Engineering. **© *IEEE Computer Society***.
[12]. Josh Shaul and Aaron Ingram. **2011**. *"Practical Oracle Security: Your Unauthorized Guide to Relational Database Security"*, SYNGRESS© Syngress Publishing, Inc.
[13]. Stallings, W., **2016**. *Cryptography and Network Security: Principles and Practice. 7th ed.* ***Prentice Hall***.
[14]. Oracle. **2013**. *"Technical Comparison of Oracle Database 12c vs. Microsoft SQL Server 2012: Focus on High Availability*". Oracle White Paper November **2013**.
[15]. "Mikael Ronström, Jonas Oreland", Recovery Principles of MySQL Cluster 5.1. MYSQL-1:30 pm-15/7/2009 http://www. mysql. com/products/backup
[16]. Awad M. Awadelkarim and Anass A. Nour.**2017** *"Security Perception and Features of RDBMSs"* ***Masters Dissertations at repository.sustech.edu/handle/123456789/5149*** **Retrieved 2107-01-07.**