JOMO KENYATTA UNIVERSITY OF AGRICULTURE AND TECHNOLOGY

MASTERS OF SCIENCE IN INFORMATION TECHNOLOGY

PETER IRUNGU MWANGI

SCT321-C004-2079/2018

MIT 3104 ADVANCED DATABASE SYSTEMS AND MANAGEMENT

ASSIGNMENT

---

1. Do a comparative review of the various approaches for securing DBMS applications that are deployed on the web. [20 Marks]

DBMS (Database Management System)

A database-management system (DBMS) is a collection of interrelated data and a set of programs to access those data. The collection of data, usually referred to as the database, contains information relevant to an enterprise. The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient (Anjard, 1994).

In choosing a DBMS from the variety of concepts and vendors, one of the consideration before making a decision is the security which ranges from data security, data protection, multi access and integration.

DBMS characteristics

A modern DBMS has the following characteristics;-

- **Represent Some Aspects of real world applications -** A database represents some features of real world applications. Any change in the real world is reflected in the database.
- **Self-Describing nature -** A database is of self-describing nature; it always describes and narrates itself. It contains the description of the whole data structure, the constraints and the variables

- **Logical relationship between records and data -** A database gives a logical relationship between its records and data. So a user can access various records depending upon the logical conditions by a single query from the database.
- **Control Data Redundancy -** DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values.
- **Query Language -** DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and as different filtering options as required to retrieve a set of data.
- **Multiuser and Concurrent Access -** DBMS supports multi-user environment and allows them to access and manipulate data in parallel.
- **Multiple views of database -** Basically, a view is a subset of the database. A view is defined and devoted for a particular user of the system. Different users of the system may have different views of the same system.
- **Security -** Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features.

DBMS security threats and approaches for securing them (Amin et al., 2017)

DBMS security Threat	**Threat description**	**Measure to mitigate threat**
Primary threats	Primary threats to the security of a database server involve unauthorized disclosure or modification of sensitive information	To counter these measures, the DBSSO, DBSA, and OSA must ensure that all users of the DBMS are identified and authenticated before they are able to use or ` the software or data.
Privileged activity threat	Improper or unchecked activity by users with privileged roles	• The DBSSO can enable auditing of all DBSA

	(DBSSO, AAO, DBSA, or OSA) can introduce security vulnerabilities and possible threats to the database server.	actions, and the AAO can review DBSA actions in the audit trail. • The countermeasure to a threat from the DBSSO is independent scrutiny of the DBMS audit trail because auditing DBSSO actions are enabled by the AAO • The countermeasure to an OSA threat is independent scrutiny of the activities of the OSA, as recorded in the audit trail. • The countermeasure to this threat is to ensure that an AAO is authorized to view information that might be yielded when the database audit trail is reviewed
Shared memory connection threats	A shared-memory connection provides fast access to a database server if the client and the server	The OSA ensures that the shared-memory connection method is not specified in the

	are on the same computer, but it poses some security risks. False or no trusted applications can delete or view message buffers of their own or of other local users. Shared-memory communication is also vulnerable to programming errors if the client application explicitly addresses memory or over-indexes data arrays.	configuration file for client/server connections. If the client and the server are on the same computer, a client can connect to a server with a stream-pipe connection or a network-loopback connection.
Threats from malicious software	Database users can easily and unknowingly download malicious or unauthorized software. This is a security threat that can come from not only server machines that host the databases, but also computers used to access the databases	To protect the database server from malicious software: • Keep the database server on a different computer from the clients that must connect to it • Restrict access to the computer hosting the database server • Monitor the software installed on the database server computers (for example, by running a checksum process periodically) • Keep a record of all the files and permissions on the database server

		computer • Institute a strict security policy • Make all users aware of the dangers of starting software of unknown or untrusted origin
Remote access threats	When a user is granted database access privileges, the host computer of the user is not specified. Therefore, the user can gain access to the privileged data from any computer that is configured to connect to the host computer. As a result, a user might not be aware of having remote access to privileged data when the user grants another user direct access to that data.	Make sure that all users are aware that access privileges are granted to user names, with no dependencies on the origin of the remote connection.
Obsolete user threats	A user is identified by user name or user ID or both. The data access privileges and individual user audit masks of the software are based on the user name. At the operating-system level, a user account might be removed and this user name might become unassigned. If any of the access privileges of the software or the individual user	To avoid this problem, have the OSA notify the DBSA when a user account is removed from the operating system. The DBSA can then perform the actions necessary to eliminate references to this name in the DBMS. These actions might involve revoking access privileges and removing

	audit mask associated with that user name are not removed before the same user name is allocated to a new user, the new user inadvertently inherits the privileges and audit mask of the previous user.	an individual audit mask
Untrusted software used in a privileged environment	Problems might occur if DBSAs or OSAs run untrusted software. Untrusted software can use the privileges of the DBSA or the OSA to perform actions that bypass or disable the security features of the product or that grant inappropriate access privileges.	The primary countermeasure to this vulnerability is to make sure that DBSAs and OSAs do not run software of unknown or untrusted origin.
Distributed database configuration threats	A distributed database user might gain access to data on a remote system with an incompatible configuration when that data would not be accessible to the same user directly on the remote system. In the worst case, the software might connect two systems that have an account with the same user name but are owned by a different user. Each user is granted the privileges of the other user at access of the database that is located on the host computer of the other user.	When two UNIX workstations are connected, the OSA must ensure that accounts with user names in common are owned by the same user.

2. Discuss the theoretical and technological challenges and opportunities presented by NoSQL database systems. [20 Marks]

Introduction and definition

NoSQL, stands for ―"Not Only SQL," refers to an eclectic and increasingly familiar group of non-relational data management systems; where databases are not built primarily on tables, and generally do not use a standard query language (SQL) for data manipulation.

NoSQL database management systems are useful when working with a huge quantity of data when the data's nature does not require a relational model (Imam, Basri, Ahmad, Watada, & González-Aparicio, 2018).

NoSQL systems are distributed, non-relational databases designed for large-scale data storage and for massively-parallel data processing across a large number of commodity servers. They also use non-SQL languages and mechanisms to interact with data though new feature APIs that convert SQL queries to the system‘s native query language or tool.

Theoretical challenges

- In NoSQL migration, the major challenge theoretical (human challenge) is the deviation from the relational-only mind set as NoSQL does not permit relationships. This is because, in a relational database is a data structure that allows you to link information from different 'tables' which NoSQL does not implement due to static nature of the schema, large feature set, non-linear query execution time and works well on a few dataset.
- Other challenges includes; leveraging on non-integer keys – that is keys or unique identifiers in a NoSQL database are often strings rather than integers or GUIDs, maintaining identity within non-roots – this is a hurdle of moving to NoSQL since not all objects are roots, that is, some objects exist as a child inside the parent object thus how does one maintain identity within these non-roots

Technological challenges

- Essentially NoSQL are implemented to achieve high performance and high scalability, this results to going without one or more staples of SQL databases. Therefore atomic, consistent, isolated, durable (ACID) guarantees are forfeited to gain performance and scalability.
- Instead of offering ACID (atomic, consistent, isolated, durable) properties, NoSQL databases basically implements BASE (basically available, soft state, eventually consistent) properties. The end result is that your data remains consistent with atomic, multi-document writes, and your queries remain fast, running against pre-computed BASE indexes.
- NoSQL databases, you need to plan for data that may break business rules. While SQL databases would have foreign keys with consistency to ensure the database is always in a valid state, most NoSQL databases have no such guarantee.
- Many NoSQL databases don’t have the concept of stored procedures and triggers.

Opportunities of NoSQL

NoSQL databases were created in response to the limitations of traditional relational database technology. When compared against relational databases, NoSQL databases are more scalable and provide superior performance, and their data model addresses several shortcomings of the relational model.

The advantages of NoSQL include being able to handle:

- Large volumes of structured, semi-structured, and unstructured data
- Agile sprints, quick iteration, and frequent code pushes
- Object-oriented programming that is easy to use and flexible
- Efficient, scale-out architecture instead of expensive, monolithic architecture

3. Discuss the Functionality and Issues associated with mobile DBMS. [20 Marks]

Mobile Database Management System

A mobile database is a database that can be connected to by a mobile computing device over a mobile network, the database is portable and physically separate from the corporate database server.

Mobile database also allows the development and deployment of database applications for handheld devices, thus, enabling relational database based applications in the hands of mobile workers. The database technology allows employees using handheld to link to their corporate networks, download data, work offline, and then connect to the network again to synchronise with the corporate database (Technologies, 2013).

For example, with a mobile database embedded in a handheld device, a package delivery worker can collect signatures after each delivery and send the information to a corporate database at day's end.

The current database systems do not provide special facilities for specific update operations in a mobile computing environment. Some of the commercially available Mobile relational Database systems are:

- IBM's DB2 Everywhere 1.0
- Oracle Lite
- Sybase's SQL

Requirements of Mobile DBMSs

Mobile DBMSs should satisfy the following requirements:

Small memory footprint - Memory footprint is amount of main memory that an application uses while running. Mobile devices have limited memory, so the mobile database application should have a small footprint. The size of mobile database affects the overall application footprint. Mobile DBMSs should be customizable to include only the required database functionalities

Flash optimized storage system – Flash memories are dominant storage devices for portable devices. They have feature such as: Small size, Better shock resistance, Low power consumption, Fast access time, and No mechanical seek and rotational latency. Mobile DBMSs need to be optimized to exploit the advantages of the new storage devices.

Data synchronization - Portable devices cannot stay connected all the time. Users can access and manipulate data on their devices. They are also unable to store a large amount of data due to lack of storage capacity. Mobile DBMSs should have the synchronize functionality to integrate different versions of data into a consistent version.

Security - Security is very important for data-centric mobile applications. It is more important when the application works with critical data that its disclosure results in potential loss or damage. Data that are transmitted over a wireless network are more prone to security issues. Mobile DBMSs should implement a complete end-to-end security to ensure the secure transfer of data.

Low power consumption - Portable devices have limited power supplies. Battery life of mobile phones is expected to increase only 20% over the next 10 years. Processor, display and network connectivity are the main power consumers in a mobile device. Mobile DBMSs need to be optimized for efficient power consumption

Self-management - In traditional databases, the database administrator (DBA) is responsible for databases maintenance. In mobile DBMSs there can be no DBA to manage the database. Mobile DBMSs need to support self-management and automatically perform the DBA tasks. Some mobile DBMSs allow remote management that enables a DBA to manage the mobile databases from a remote location.

Embeddable in applications - Administrators does not have direct access to mobile devices. Mobile DBMSs should be an integral part of the application that can be delivered as a part of the applications. The database must be embeddable as a DLL file in the applications. It must be also possible to deploy the database as a stand-alone DBMS with support of multiple transaction

Functionality issues of mobile DBMS (Bhagat & B, 2014)

Security Standards - When working mobile, one is dependent on public networks, requiring careful use of Virtual Private Network (VPN). Security is a major concern while concerning the mobile computing standards on the fleet. One can easily attack the VPN through a huge number of networks interconnected through the line (Lubinski, n.d.).

Power consumption - When a power outlet or portable generator is not available, mobile computers must rely entirely on battery power. Combined with the compact size of many mobile devices, this often means unusually expensive batteries must be used to obtain the necessary battery life.

Insufficient Bandwidth - Mobile Internet access is generally slower than direct cable connections, using technologies such as GPRS and EDGE, and more recently 3G networks. These networks are usually available within range of commercial cell phone towers. Higher speed wireless LANs are inexpensive but have very limited range

Transaction models - In mobile environment, the issues of correctness of transactions and fault tolerance are aggravated. All transactions must satisfy the ACID properties, these are atomic, consistency, isolation, and durability. Depending upon the movement of the mobile unit, possibly on multiple data sets and through several base station, a mobile transaction is executed sequentially. When the mobile computers are disconnected, ACID properties gets hard to enforce

Replication issues - There is increase of costs for updates and signalling due to increase in number of replicas. Mobile hosts can move anywhere and anytime.

Division of labour - There is a certain change in the division of labour in query processing because of certain characteristics of the mobile environment. There are some of the cases in which the client must function independently of the server

Disconnection - Weather, terrain, and the range from the nearest signal point can all interfere with signal reception. Reception in tunnels, some buildings, and rural areas is still poor. Interaction between a mobile device and a database is directly affected by the device's network connectivity.

Limited storage - Due to mobility and portability, the sizes of memory and hard drive are smaller than the ones in the wired network. The consequences of this are less stored/cached/replicated data, fewer installed applications, and more communication

References

Amin, M., Yunus, M., Krishnan, S. K. V. G., Nawi, N. M., Salwana, E., & Surin, M. (2017). Study on Database Management System Security Issues, *1*(4), 192–194.

Anjard, R. P. (1994). The Basics of Database Management Systems (DBMS). *Industrial Management & Data Systems*, *94*(5), 11–15. https://doi.org/10.1108/02635579410063261

Bhagat, A. R., & B, P. B. V. (2014). Mobile Database Review and Security Aspects, *3*(3), 1174–1182.

Imam, A. A., Basri, S., Ahmad, R., Watada, J., & González-Aparicio, M. T. (2018). Automatic schema suggestion model for NoSQL document-stores databases. *Journal of Big Data*, *5*(1), 46. https://doi.org/10.1186/s40537-018-0156-1

Lubinski, A. (n.d.). Security issues in mobile database access.

Technologies, E. (2013). M ANAGEMENT I SSUES AND C HALLENGES IN M OBILE, *5*(1), 1–6.