Declan Carroll
S19003901

Introduction.

This is a report that refers to the Helpdesk database and assess its technical suitability in terms of data security, availability and integrity. It makes recommendation where required.

Database Security.

“Data is a valuable resource that must be strictly controlled and managed, as must any corporate resource. Part or all of the corporate data may have strategic importance to an organization and should, therefore, be kept secure and confidential.” [1]

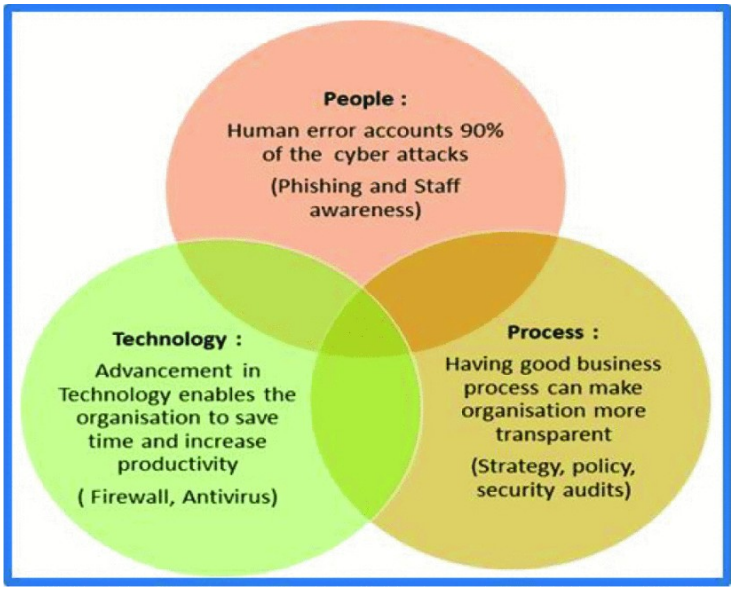

Fig 1.0

A SWOT analysis [6] was carried out by the Helpdesk development team. Top rating Threats [7] of SWOT were broken out into Tables 1A/B with Overall Consequences levels assigned.

			Overall Consequences			
		Impact	**Minor** Database available but with some deviations	**Moderate** Database unavailable >20mins. Possible Data integrity issue.	**Major** Database unavailable >2 Hours. Data integrity issue.	**Severe.** Database unavailable >1 day. Data integrity/corruption.
Likelihood	**Almost Certain**	Expected to occur >1 time in the future.	Medium	Large	Very Large	Very Large
	Likely	Expected to occur at some stage.	Medium	Large	Large	Very Large
	Possible	May occur in future.	Medium	Medium	Medium	Large
	Unlikely	Not likely in normal circumstances	Low	Medium	Medium	Medium
	Rare	Could happen but not probable	Low	Low	Low	Medium

Table 1A

Threat Number.			Likely Hood	Impact	Overall Consequences
	Hardware				
1		Part Fail	Likely	Major	Large
2		Power surge	Likely	Major	Large
3		Act God (weather, flooding etc)	Possible	Severe	Large
4		Theft	Possible	Major	Medium
	Software				
5		Malware	Possible	Moderate	Medium
6		End support/Forced upgrade	Likely	Minor	Medium
7		SQL injection	Possible	Major	Medium
	Users				
8		Shared Logins	Likely	Moderate	Large
9		Virus	Likely	Severe	Very Large
10		Human Error	Almost certain	Major	Very Large
11		Excessive privledges	Likely	Severe	Very Large

Table 1B.

Oracle provides two types of security, [1] [2];

System security – access to the database for authorized users only [1].

The Helpdesk features both Apple and Microsoft OS. Both feature 2FA [4,5] through phone/email and this was considered as a method to authenticate from the OS to Oracle.

The risk here is that updates to the OS or newly added equipment that does not support this feature would render the database unaccessible.

The Helpdesk is an internal system (not public facing) that has just employee access. Oracles own stand-alone database authentication can be used and includes;

Password Encryption, Account Locking, Password Lifetime, Expiration +Complexity Verification [2].

Point 8, table 1B is still relevant but this can be mitigated against in company security policies.

Data security – through Discretionary access control (DAC) [8].

This section addresses Threat point 11 .

DAC can be used to give specific access privileges based on a users needs.

While is is possible to set each user privileges individually Oracle supports CREATE ROLES [9]. A user is assigned to a ROLE, each ROLE is GRANTed [10] privileges.

Roles in the Helpdesk can be assigned as Table 2.

We can create a user using SQL;

CREATE USER user1 IDENTIFIED BY password

DEFAULT TABLESPACE users

TEMPORARY TABLESPACE temp;

We can assign the Manager ROLE to user1 to a ROLE using

GRANT Manager TO User1;

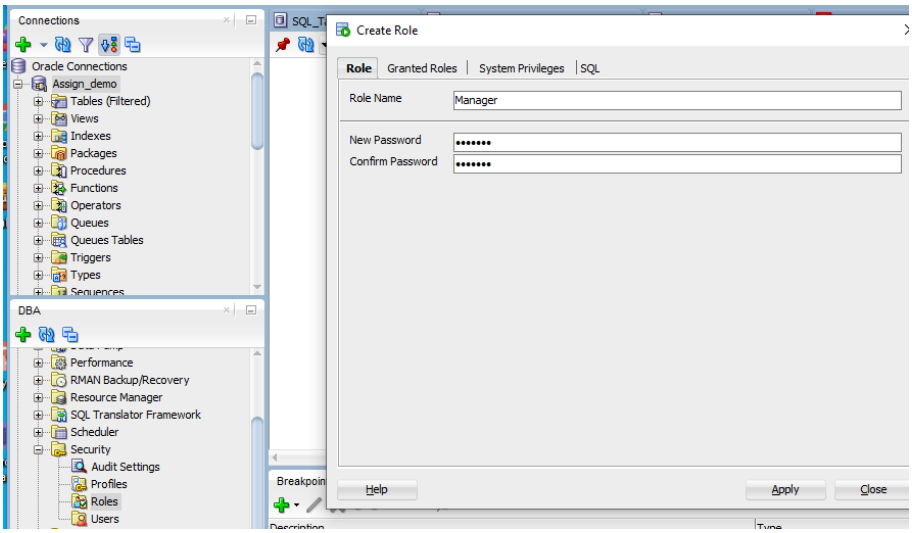

Fig 1. Creating a Manager role through DBA GUI in SQL developer.

Job	Department.	# Persons	Role	Table	Allow SELECT	Allow INSERT	Allow UPDATE	Allow DELETE	Allow REFERENCE	Allow ALL	Allow GRANT to other users
Manager	Sales	1	Manager	All Tables	✓	✗	✗	✗	✗	✗	✗
	Customer_Support	1	Manager								
	Accounting	1	Manager								
Office Admin	Accounting	1	Admin	All Tables	✓	✓	✓	✓	✓	✓	✓
Database Admin	IT	1	DBA		✓	✓	✓	✓	✓	✓	✓
HD_Oper	Tech Support	2	Operator	All Tables	✓	✗	✗	✗	✗	✗	✗
			Operator	Call_Details	✓	✓	✓	✗	✗	✗	✗
			Operator	Resolution_Codes	✓	✓	✓	✗	✗	✗	✗
			Operator	Problem_Category_Codes	✓	✓	✗	✗	✗	✗	✗
HD_Spec	Tech Support	2	Specialist	All Tables	✓	✗	✗	✗	✗	✗	✗
HD_Spec	Customer_Support	3	Specialist	Call_Details	✓	✗	✓	✗	✗	✗	✗
			Specialist	Resolution_Codes	✓	✓	✗	✗	✗	✗	✗
Tech	IT	2	IT	All Tables	✓	✗	✗	✗	✗	✗	✗
			IT	Equipment_Reg	✓	✓	✓	✓	✗	✗	✗
			IT	Equipment_Owners	✓	✓	✓	✓	✗	✗	✗
Total Employees in the company		**28**									

Note 1 : Helpdesk has staff whose their main job function is not solely the HelpDesk e.g. HD Specialists.

Note 2: DBA role can Create/Drop tables, New Users and Roles.

Table 2.

Views: Used to restrict access both user access to a view and restricted column view of table. Mangers are the ideal group in the Helpdesk to setup views for, as they typically are not altering information in the database. A recommendation here is that the information managers are interested in, have views created. Direct table access privileges can then be REVOKEd. This helps with threat point 11.

Input Validation and Sanitization [12]; (Threat points 7, 10). Helpdesk does not check all data input from the user is valid and so open to SQL injection attacks[13]. These recommendations from Oracle [14] should be implemented as a minimum . This also helps maintaining data integrity.

Back-up and recovery: (Threats 1-4, possibly 5,7,9 &10) In order to recover the database after a failure we need to know that all transactions were completed and data integrity maintained before the failure.

Concurrency Control: In a multi-user database we need to prevent data being updated by one user while it is being queried by another user. (See Dirty, Non-repeatable & Phantom reads [15]). One way to do this is to run transactions serially i.e. one transaction cannot start until the previous transaction is complete. This however is not practical as it severely limits throughput. [15] It would be possible to add a 'Lock' column to each table and setting that attribute to flag shared (perhaps a number >0 for the number of shared locks active) or exclusive lock (e.g. -1). Transactions could then check the attribute lock state before running the query/update. See [1] PP683. Fortunately Oracle maintains locks automatically in two different ways. Read Committed and Serializable Isolation [15]. There is a trade-off in terms of concurrent users vs data read consistency. See Table 3 [15].

Isolation Level	Dirty Read	Nonrepeatable Read	Phantom Read
Read uncommitted	Possible	Possible	Possible
Read committed	Not possible	Possible	Possible
Repeatable read	Not possible	Not possible	Possible
Serializable	Not possible	Not possible	Not possible

Table 3.

For the HelpDesk we can set serialisation using;

```
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
```

As there are only some users with INSERT/UPDATE privileges we have gone for data constancy here. An error will results if there are serialisation conflicts and as there are a low number of users are located in the same office, this can be resolved locally.

Data Integrity - Constratinst on data being inserted into tables. i.e. nulls, datatypes, PK for every FK and num chars.

Sanitation of data

Transaction consistancy - Livelock + Starvation

Conclusion

polyinstantiation not necessary.

References.

[1] T.M, Connolly; C.E, Begg, Database Systems - A Practical Approach to Design Implementation and Management, PP623, Pearson Education, 2015.

[2] A. H. Khan, P. Bahl Sawhney, S. Das, and D. Pandey, 'SartCyber Security Awareness Measurement Model (APAT)', in 2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC), Feb. 2020, pp. 298–302, doi: 10.1109/PARC49193.2020.236614

[3]'Database Concepts'. https://docs.oracle.com/cd/B19306_01/server.102/b14220/security.htm (accessed Nov. 01, 2020).

[4] W. D. Blog, 'Convenient two-factor authentication with Microsoft Passport and Windows Hello', *Windows Developer Blog*, Jan. 26, 2016. https://blogs.windows.com/windowsdeveloper/2016/01/26/convenient-two-factor-authentication-with-microsoft-passport-and-windows-hello/ (accessed Nov. 06, 2020).

[5] 'Two-factor authentication for Apple ID', *Apple Support*. https://support.apple.com/en-gb/HT204915 (accessed Nov. 01, 2020).

[6] 'ProjectManagement.com - SWOT Analysis'. https://www.projectmanagement.com/wikis/233091/SWOT-Analysis (accessed Nov. 07, 2020).

[7] 'Database Security: An Essential Guide', Dec. 13, 2019. https://www.ibm.com/cloud/learn/database-security (accessed Nov. 06, 2020).

[8] S.-K. Chin, S. B. Older, and D. R. Stinson, *Access Control, Security, and Trust: A Logical Approach*. PP79, Philadelphia, PA, UNITED STATES: CRC Press LLC, 2010.

[9] 'Database SQL Reference'. https://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_6012.htm (accessed Nov. 01, 2020).

[10] 'GRANT statement'. https://docs.oracle.com/javadb/10.8.3.0/ref/rrefsqljgrant.html (accessed Nov. 01, 2020).

[11] R. Lowenthal, 'How much Database Security is Enough? Know where to start'. https://blogs.oracle.com/cloudsecurity/how-much-database-security-is-enough-know-where-to-start (accessed Nov. 01, 2020).

[12] 'What is input validation and sanitization?' https://download.oracle.com/oll/tutorials/SQLInjection/html/lesson1/les01_tm_ovw3.htm (accessed Nov. 07, 2020).

[13] 'What Is SQL Injection? 8 Tips on How to Prevent SQL Injection Attacks', *InfoSec Insights*, Jun. 05, 2020. https://sectigostore.com/blog/what-is-sql-injection-8-tips-on-how-to-prevent-sql-injection-attacks/ (accessed Nov. 08, 2020).

[14] 'Preventing SQL Injection'. https://docs.oracle.com/cd/E80738_01/pt854pbh2/eng/pt/tpcd/task_PreventingSQLInjection-0749b7.html#topofpage (accessed Nov. 08, 2020).

[15] 'Database Concepts'. https://docs.oracle.com/cd/B19306_01/server.102/b14220/consist.htm (accessed Nov. 08, 2020).