sustainability

MDPI

Article

Sustainable Implementation of Access Control

Mihaela Muntean [1,*] and Laurenţiu Dijmărescu [2]

[1] Business Information Systems Department, Faculty of Economics and Business Administration, West University of Timisoara, 30023 Timisoara, Romania

[2] Faculty of Economic Cybernetics, Statistics and Informatics, The Bucharest University of Economic Studies, 010371 Bucharest, Romania; dijmarescu_laurentiu@yahoo.com

* Correspondence: mihaela.muntean@e-uvt.ro

Received: 26 April 2018; Accepted: 29 May 2018; Published: 30 May 2018

Abstract: Sustainable implementation of access control implies approaches at different levels. Beyond the authorization framework at application level, the demarche will be strengthened by providing security objects at database level. In terms of sustainable information systems, the proposal extends the integrated security approach of an SAP application with initiatives at database level programmed in PL/SQL. The organization's policies and procedures are taken into consideration.

Keywords: information security; authorization; access control; security objects

1. Introduction

According to Needham and Maybury, access control can be introduced at different levels: Application level, database level, operating system, or hardware [1]. Regardless of level, access control is a security technique that is used to establish who can use different resources in a computing environment. Vimercati, Foresti and Samarati are defining access control as "the process of controlling every request to a system and determining, based on specified rules (authorizations), whether the request should be granted or denied" [2].

Bourgeois and Bourgeois introduced different tools for information security as part of an overall information security policy [3]. "Information Security Policy (ISP) is a set of rules enacted by an organization to ensure that all users or networks of the IT structure within the organization's domain abide by the prescriptions regarding the security of data stored digitally within the boundaries the organization stretches its authority" [4]. IPS elements are: 1—purpose; 2—scope; 3—objectives (confidentiality, integrity, availability); 4—authority & access control policy; 5—classification of data (high risk class, confidential class, public class); 6—data support & operations; 7—security awareness sessions; 8—responsibilities, rights and duties of personnel; 9—reference to relevant legislation; 10—others, e.g., virus protection procedure, intrusion detection procedure, remote work procedure, technical guidelines, audit, employee requirements, consequences for non-compliance, disciplinary actions, terminated employees, physical security of IT, and references to supporting documents [4].

Managing access control implies different scenarios in order to minimize the vulnerability of being exploited by malicious users. According to Kayem, Akl and Martin, any access control method faces one of the following weaknesses: "Vulnerability to security violations, inefficiency in management resulting in delays as well as reduced availability, and a lack of inbuilt mechanisms that allow them handle new scenarios adaptively". Access control models, like the discretionary access control model (DAC) and the mandatory access control model (MAC), establish the general framework for access control. Role-based access control (RBAC), a combination of mandatory and discretionary access control, is a more flexible approach; several roles can be granted to a user and a role can be associated with several users [5]. Advanced initiatives are based on multilevel access control

models (MLS), e.g., cryptographic access control schemas (CAC) are MLS models that are capable of providing security in different contexts without requiring extensive changes to the fundamental architecture [5]. They use a combination of symmetric and asymmetric cryptography to provide user authentication, data confidentiality and integrity. Access control is determined by the possession of the cryptography keys. Despite their advantages in terms of security, their reliance on costly algorithms has represented an impediment in their use on a large scale. Recent research is trying to implement the benefits of cryptography in the RBAC model [6,7].

Enterprise information systems, developed nowadays as sustainable information systems, have integrated a security approach. Role-based access control models are commonly used. The administration of users and their access rights in large enterprises is complex and challenging. With respect to the governance framework, including corporate rules, practices, processes and people, an enterprise role-based access control (ERBAC) system will be designed. The ERBAC model must benefit of the whole support of the enterprise information system infrastructure, the Enterprise Resource Planning (ERP) system. As a market leader in ERP software, Systems, Applications and Products (SAP) helps companies in the all-day business.

SAP security design [8] is based on the SAP authorization concept. Authorization objects, authorizations, roles and profiles are consolidating the authorization framework [8]. Beyond the access control implementation at application level according to the authorization framework, the demarche proposes some security objects at database level for a sustainable ERP system. Thereby, a sustainable access control implementation is proposed.

2. Authorization and Access Control

A sustainable business implies also the sustainability of the information systems and technologies that are used to process the data, to support business transactions, to perform business analysis and generate business scenarios, and to speed up the decision-making processes. According to Tayeh and Myrah, sustainable information systems (SIS) are "information systems that are created, used and maintained to provide the greatest possible benefit to sustainable development" [9]. The agile development of these information systems is conducting to high-performing, secure systems, and is implicitly assumed. Identified as sustainability information systems by Junker and Farzad, they sustain the four sustainability demands on enterprises [10]. The extended business model enriched with the sustainability dimension [11–13] needs to be supported by advanced information systems with new capabilities. Identified as Enterprise Resource Planning (ERP) systems, Customer Relationship (CRM) systems, Supply Chain Management (SCM) systems and Business Intelligence (BI) systems, all these enterprise applications are developed according to best practices in information system design and development [14,15], including information system security management. Economically oriented ERP systems are sustainable information systems [10].

According to Bourgeois, the information security triad is composed of confidentiality, integrity and availability (CIA) [3]. A security policy is established at organization level based on business requirements, standards and guidelines [16–18]. Authentication and access control are two methods to enable the access into the SIS only for those who are authorized to perform different tasks and therefore have been defined as end-users of the information systems. We identify four end-user categories: Information consumer/business user; business analyst/power user; middle management; and C-level management and leadership. Both user authentication and access control are defined according to the authorization framework. Referring to SAP applications, authorizations are the key building blocks of SAP security [19].

Business objects and transactions in SAP are protected by authorization objects, users require corresponding authorization to access the business objects or to execute the transactions.

The authorization framework establishes the relationships between the users and the authorization objects (Figure 1).

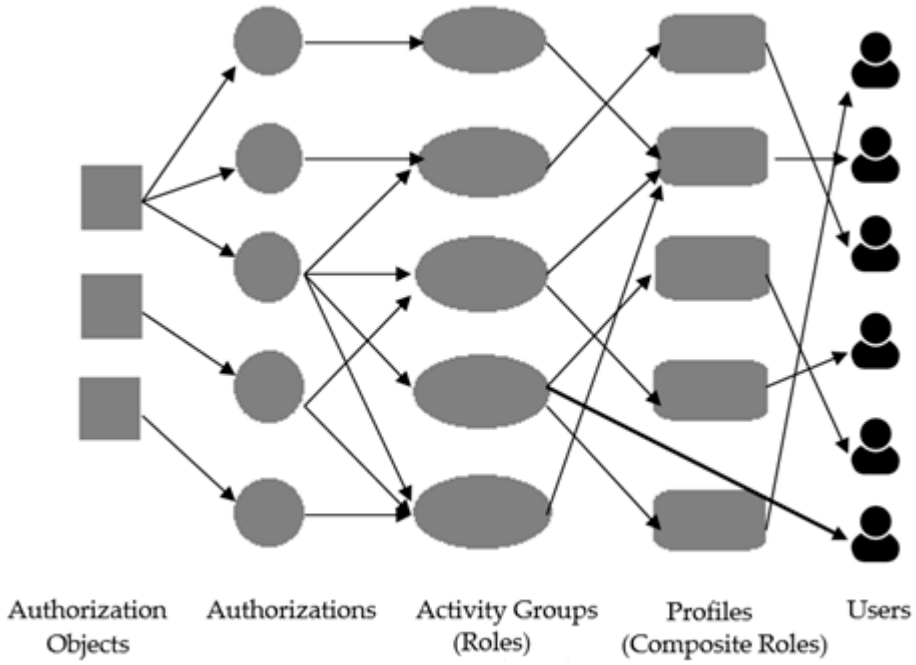

Figure 1. Authorization framework (adapted from [19]).

Authorizations are instances of generic authorization objects and are combined in activity groups that are associated with roles. According to the users needs, the profiles are designed based on singe roles or composite roles.

Authorizations can be useful in limiting access to items such as: billing and vendor information, personnel and payroll information, key financial data, and critical system areas such as basis, configuration, development, and security. Users obtain their authorizations by being assigned to roles and users cannot start a transaction or complete a transaction without the proper authorization role assignment. In order to perform an action, a user may need several authorizations. For example, in order to create a sales order, the user will need access to the transaction, the "create" authorization, general authorization for the sales org, and the authorization for the specific sales document type. Therefore, the relationships required in order to meet user access requirements can become very complex.

3. Sustainable Implementation of Access Control

Sustainability is a new dimension of the information systems and their robust operation in the business environment. Users' access is performed according to the security configurations in SAP. Beyond the role-based access control mechanism at application level, the access to data at database level is also controlled. Oracle native security tools, like Oracle Advanced Security and Oracle Database Vault, are implied.

Additionally, for monitoring users' access, stored procedures and triggers in Oracle have been defined. Also, a way for locking and unlocking users has been introduced at database level. The proposed stored procedures and triggers are programmed in the procedural language extension to Structured Query Language (PL/SQL). Due to their defined functionality, they are referred as security objects in the present demarche (Figure 2). A higher speed for the monitoring process is achieved and troubleshooting intervention is more quickly possible. Our contribution in defining the security objects is meant to minimize the chances of the system being exploited by malicious users (Figure 2). The hybrid access control framework is based on an access control model at application level, native database security mechanism and the security objects. The security framework has been integrated into an ERP application that has been put into operation.

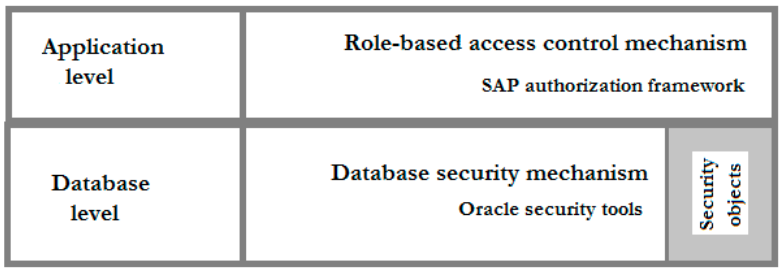

Figure 2. Security framework.

After analyzing the business processes and the end user tasks, the following main roles in a client company have been identified: CEO user, HR user, Accountant user, SAP Super User (with full rights) and ABAP developer. According to these main roles, the SAP users were defined. Their profile is automatically generated by generating the underlying composite roles. A composite role is a container which can collect several different roles [19]. In SAP a role can be defined based on a predefined user role template or based on a modified one as per user needs. Role templates are associated with activity groups in SAP consisting of transactions, reports and web addresses [20]. For each main role identified in the company a role template has been established Considering, for example an ABAP developer, the profile of his/her SAP user has been defined based on the role template in Figure 3.

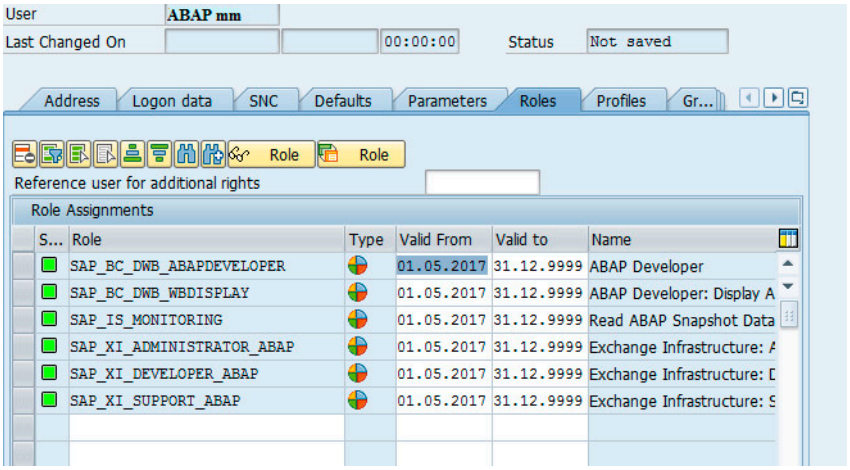

Figure 3. Defining a SAP user. Role template for an ABAP programmer.

User type (service or dialog), password and validity period of the password are established—Logon data (Figure 3). All information about the users is stored in SAP table USR02. Modifications of the logon data are stored in table USH02 [21].

Further, the hybrid approach is based on some facilities programmed in PL/SQL. Using a keyboard-interactive authentication the access is granted directly to the underlying Oracle server, where the communication with the system is made by SQL*Plus. To prevent damages made by a malicious user, it is necessary to have the possibility to lock immediately that user. In emergencies or during maintenance periods locking all users is necessary. An evidence of all access modifications is proposed, the solution implies an additional table, UflagHistory (code sequence 1), and two triggers: One, Changes_Lock_Unlock, associated with any update event on SAP table USR02, is activated before the event and stores the access modifications in the history table (code sequence 2); and the second one, IdHistory, associated with the insert event on the history table, is acting before the event and is automatically generating the primary key values (code sequence 3). Table UflagHistory is created by the following SQL command.

```
CREATE TABLE UflagHistory (id NUMBER(12) CONSTRAINT pk_iduflag PRIMARY KEY,
mandt VARCHAR2(3), bname VARCHAR2(36), uflag NUMBER(3),                    (1)
datac VARCHAR2(36), user_modif VARCHAR2(20) DEFAULT USER NOT NULL),
```

where *user_modif* is the user that has locked (uflag = 64) or unlocked (uflag = 0) user *bname*. Last connection of user *bname* to the SAP system was registered on *datac*.

```
CREATE OR REPLACE TRIGGER Changes_Lock_Unlock
BEFORE UPDATE ON SAPSR3.usr02
FOR EACH ROW
BEGIN                                                                      (2)
   INSERT INTO UflagHistory (mandt, bname, uflag, datac)
   VALUES (:OLD.mandt, :OLD.bname, :NEW.uflag,
        TO_CHAR (SYSDATE,'MM-DD-YYYY HH24:MI:SS'));
END;
```

SAP table USR02 contains information about the users: mandt-client; bname-user name in user master record; gltgv-user valid from; gltgb-user valid to; ustyp-user type; class-user group; uflag-user lock status; aname-creator of the user master record; trdat-last login date; and ltime-last logon time. In the BEFORE UPDATE trigger the OLD and NEW transition variables allow access to the user information in table USR02 (:OLD.mandt; :OLD.bname) and to the new updates (:NEW.uflag) that will be performed (code sequence 2). :NEW.uflag represents the modified status of the users according to the locking or unlocking procedure (code sequence 4).

```
CREATE OR REPLACE TRIGGER IdHistory
BEFORE INSERT ON UflagHistory
REFERENCING OLD AS OLD NEW AS NEW
FOR EACH ROW                                                               (3)
   BEGIN
      SELECT NVL(MAX(id),0)+1 INTO :NEW.id FROM UflagHistory;
   END;
```

A locking/unlocking procedure of all users can be formulated as following (code sequence 4).

```
CREATE OR REPLACE PROCEDURE lock_unlock_all
(p_mandt IN SAPSR3.usr02.mandt%TYPE, p_uflag IN SAPSR3.usr02.uflag%TYPE) AS
v_nr NUMBER(2):=0;
BEGIN
SELECT COUNT(mandt) INTO v_nr FROM SAPSR3.usr02 WHERE mandt = p_mandt;
IF v_nr >= 1 THEN
      UPDATE SAPSR3.usr02 SET uflag = p_uflag WHERE mandt = p_mandt;       (4)
      COMMIT;
      DBMS_OUTPUT.PUT_LINE ('In table [USR02] the access of all users
           with mandt ['||p_mandt||'] was changed in ['||p_uflag||']');
ELSE
      DBMS_OUTPUT.PUT_LINE ('Mandt ['||p_mandt||'] does not exist');
END IF;
END;
```

Two parameters are introduced: p_mandt for specifying the client and the corresponding group of users, and p_uflag for locking/unlocking these users.

Similar, further processing and monitoring procedures can be developed and stored in the database server.

4. Discussion

Access control is a central element of computer/system security implying the prevention of an unauthorized use of a resource. There are a number of techniques that can be used for controlling the access to resources, e.g., Role-based Access Control (RBAC), Mandatory Access Control (MAC), Discretionary Access Control (DAC), Cryptographic Access Control (CAC) [1,22]. Despite the benefits of the CAC model, access control in SAP systems is role-based oriented. The RBAC implementation depends on the complexity of the SAP environment, number of organizational units, number of users & roles, role design concept used, and organizational culture.

In the current RBAC approach, role definitions and associated access rights are based upon a thorough understanding of the organization's security policy. In fact, roles and the access rights that go with them are directly related to elements of the security policy. In general, an authorization model establishes the relationships between the user IDs and a range of system authorizations with which they can be associated [23,24]. Authorization frameworks in SAP systems are widely treated in SAP technical papers and documentations [19,20,25], advanced studies on this topic and best practices on RBAC implementation in SAP applications are developed e.g., in references [26,27].

Marnewick and Labuschagne proposed a security framework for the ERP systems [16], the demarche is applicable to SAP applications. Security is an integral part of an ERP system and security issues are implemented along with the implementation of the ERP. But, ERP security is an ongoing process, continue monitoring of all activities is indispensable to avoid incidents. Also, the security measures need to be upgraded due to the evolution of technologies and information systems. Not at least, the ERP system is an integral part of the company and therefore it takes the organization's policies and procedures into consideration [28].

ERP systems have a three-tier or four-tier client/server architecture, the database layer is the bottom layer [29]. Also, the SAP provides a robust framework for controlling and managing user's authorizations through access control mechanisms [20], supplementary approaches at database server level will strengthen the demarche (Figure 4). Database security can help fix application security issues.

Some references have been identified that discuss Oracle security issues for SAP applications [30,31]. Oracle's security products and features are required to prevent data accesses that bypasses the SAP applications. Oracle Advanced Security and Oracle Database Vault prevent illegal accesses made using a copy of the database files, respectively using SQL statements and database tools [30]. Our demarche proposes beyond these considerations some additional security objects at database level.

Security objects at database level are data tables for storing security control information, stored procedures and/or functions, and triggers programmed in PL/SQL. Stored procedures provide a powerful way to code application logic that can be stored on the server. Stored functions are similar to procedures, except that a function returns a value to the environment in which it is called. Triggers are procedures that are implicitly fired when a triggering event occurs. The trigger action can be run before or after the triggering event. A trigger can be associated with a data table and be programmed to fire before or after an INSERT, DELETE or UPDATE statement is performed on the table [32,33].

Regarding the authorization framework and access control in SAP applications, user management include following tasks/operations [20]: 1—creating and deleting users; assigning and resetting passwords; locking and unlocking users; 2—creating roles using different methods; 3—analyzing and fixing missing authorizations; 4—creating or restoring data backups; and 5—managing the database space allocation. Daily responsibilities include troubleshooting; analyze load, alert monitoring and configuration. The subject is wide, complex and supports regular improvements [34,35].

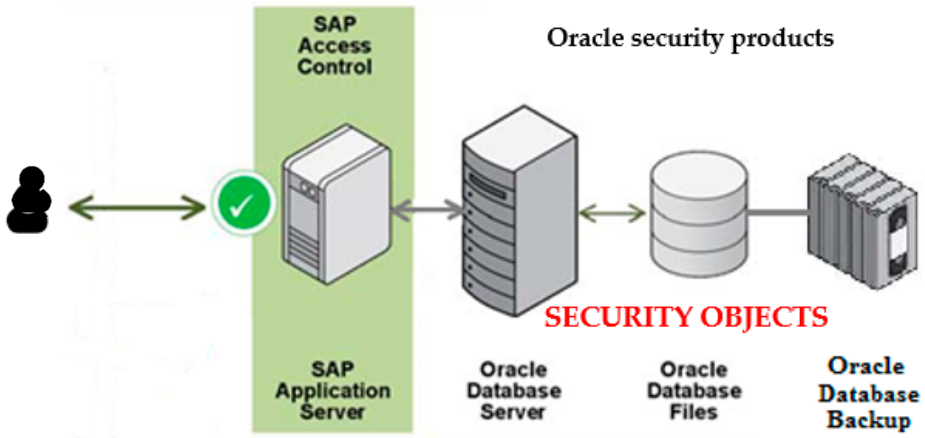

Figure 4. Sustainable access control.

Our proposal is an alternative for locking and unlocking users, alternatively implemented as a stored procedure in PL/SQL. Acting directly at database level on table USR02 in order to modify field uflag (uflag = 64 for locking or uflag = 0 for unlocking the referred users), the procedure has an immediate effect. Additionally, two triggers are fired and the modification is archived in table UflagHistory.

Resuming,

$$\text{SECURITY OBJECTS} = \{\textbf{table}\ \text{UflagHistory}, \textbf{trigger}\ \text{Changes\_Lock\_Unlock}, \textbf{trigger}\ \text{IdHistory}, \textbf{procedure}\ \text{lock\_unlock\_all}\} \quad (5)$$

Our demarche comes to consolidate a sustainable access control in SAP applications with an Oracle database server. SAP applications are sustainable information systems, and security is part of the ERPs. The proposal is not restricted to the Oracle database server, a similar approach can be adopted to a SAP application with a different database server, e.g., MS SQL Server, IBM DB2.

More than that, a hybrid approach of authorization and access control can be applied to any information system. Similar SECURITY objects can be developed at database level.

5. Conclusions

Information security is continuously evolving due the evolution of technology and information systems. Authorization and access control represent a pillar of information security and are implemented at different levels. ERP systems have native mechanisms for access control at applications level, e.g., SAP applications benefit from an authorization framework based on authorization objects, authorizations, roles and profiles. At database level, e.g., Oracle server offer security tools to protect the data. Best practices in Oracle security for SAP offer support in protecting the data, isolating, if necessary, the applications, limiting user actions, and reporting on system activities. Any additional approach in consolidating a sustainable access control increases the robustness of the system. Thereby, the proposed demarche increases the capabilities of the overall security system.

Author Contributions: M.M. conceived and designed the study; L.D. fulfilled the user management implementation; M.M. and L.D. wrote the paper.

Funding: This research received no external funding.

Acknowledgments: This work was conducted by Mihaela Muntean as an associate member of ECREB—East European Center for Research in Economics and Business, Faculty of Economics and Business Administration, West University of Timisoara (http://ecreb.ro/). We would also like to acknowledge the NTT Data Romania support in realizing the sustainable implementation of access control.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kayem, A.V.D.M.; Akl, S.G.; Martin, P. *Adaptive Cryptographic Access Control*; Springer: New York, NY, USA, 2010; ISBN 978-1-4419-6654-4.
2. Needham, R.; Maybury, R. Security Control—Chapter 4. In *Security Engineering: A Guide to Building Dependable Distributed Systems*; Wiley Publishing Inc.: Hoboken, NJ, USA, 2008; pp. 93–108. ISBN 978-0-470-06852-6.
3. Vimercati, S.D.; Foresti, S.; Samarati, P. Authorization and access Control. In *Security, Privacy, and Trust in Modern Data Management*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 39–53. ISBN 978-3-540-69860-9. Available online: https://link.springer.com/chapter/10.1007/978-3-540-69861-6_4 (accessed on 20 March 2018).
4. Bourgeois, D.; Bourgeois, D.T. Information systems security—Chapter 6. In Information Systems for Business and Beyond; The Open Textbook Challenge by the Saylor Academy. 2014. Available online: https://bus206.pressbooks.com/chapter/chapter-6-information-systems-security/ (accessed on 27 January 2018).
5. Kostadinov, D. Key Elements of an Information Security Policy. 2014. Available online: https://resources.infosecinstitute.com/key-elements-information-security-policy/#gref (accessed on 12 March 2018).
6. Crampton, J. Cryptographic Enforcement of Role-Based Access Control. In International Workshop on Formal Aspects in Security and Trust. 2010; pp. 191–205. Available online: https://link.springer.com/chapter/10.1007/978-3-642-19751-2_13 (accessed on 20 December 2017).
7. Sree, P.V.; Babu, G.P. Role-Based Cryptography. *Int. J. Comput. Sci. Mob. Comput.* **2014**, *3*, 799–803.
8. Levitt, J. SAP Security Concepts, Segregation of Duties, Sensitive Access & Mitigations Control; The Institute of Internal Auditors Los Angeles Chapter; PricewaterhouseCoopers LLP. 2015. Available online: https://chapters.theiia.org/los-angeles/Events/Documents/IIA%20%20Los%20Angeles%20%20SAP%20Security%20Presentation%20.pdf (accessed on 23 November 2017).
9. Tayeh, G.A.; Myrah, T. Properties of Sustainable Information Systems. In Proceedings of SIGGreen Pre-ICIS 2016 Workshop. 2016. Available online: https://boris.unibe.ch/98846/1/ICIS2016.pdf (accessed on 28 January 2018).
10. Junker, H.; Farzad, T. Towards Sustainability Information Systems. *Procedia Comput. Sci.* **2015**, *64*, 1130–1139. Available online: https://ac.els-cdn.com/S1877050915027222/1-s2.0-S1877050915027222-main.pdf?_tid=bbacb3a5-0798-4ff0-ac4d-21b6ba037b2f&acdnat=1524206064_5a4ce9aec5a6984ea3eb7dddbf70917f (accessed on 10 September 2017). [CrossRef]
11. Muntean, M. Business Intelligence Issues for Sustainability Projects. *Sustainability* **2018**, *10*, 335. [CrossRef]
12. Petrini, M.; Pozzebon, M. Integrating Sustainability into Business Practices. *Braz. Adm. Rev.* **2010**, *7*, 362–378. [CrossRef]
13. Evans, S.; Vladimirova, D.; Holgado, M.; Fossen, K.; Yang, M. Business Model Innovation for Sustainability: Towards a Unified Perspective for Creation of Sustainable Business Models. *Bus. Strategy Environ.* **2017**, *26*, 507–608. [CrossRef]
14. Ahituv, N.; Neumann, S.; Zviran, M. A System Development Methodology for ERP Systems. *J. Comput. Inf. Syst.* **2016**, *42*, 56–67.
15. Kim, D.; Solomon, M.G. *Fundamentals of Information Systems Security*, 3rd ed.; World Headquarters: Burlington, MA, USA, 2018; pp. 112–135. ISBN 978-1284116458.
16. Stahl, S.; Pease, K.A. Seven Requirements for Successfully Implementing Information Security Policies and Standards. In A Guide for Executives. 2011. Available online: https://citadel-information.com/wp-content/uploads/2010/12/seven-requirements-for-successfully-implementing-information-security-policies-2012.pdf (accessed on 15 November 2017).
17. Labuschagne, L.; Marnewick, C. A Security Framework for an ERP System. In Proceedings of the ISSA 2005 New Knowledge Today Conference. 2005. Available online: https://www.researchgate.net/publication/220803192_A_security_framework_for_an_ERP_system (accessed on 13 January 2018).
18. Cram, W.A.; Proudfoot, J.G.; D'Arcy, J. Organizational information security policies: A review and research framework. *Eur. J. Inf. Syst.* **2017**, *26*, 605–641. [CrossRef]
19. SAP BASIS and Security Administration. 2016. Available online: http://www.acc.ncku.edu.tw/chinese/faculty/shulc/courses/cas/SAP_BASIS_and_Security_Administration.pdf (accessed on 16 December 2017).

20. SAP BASIS ADMIN Roles & Responsabilities. 2012. Available online: http://goddbstechnologiesltd.blogspot.ro/2012/05/sap-basis-admin-roles-responsibilities.html (accessed on 17 December 2017).
21. USH02 SAP Change History for Logon Data Table. Available online: http://www.se80.co.uk/saptables/u/ush0/ush02.htm (accessed on 23 December 2017).
22. Gonzales, S.M. Fraud Prevention through Segregation of Duties: Authorization model in SAP GRC Access Control. 2016. Available online: https://e-archivo.uc3m.es/bitstream/handle/10016/23673/TFG_Sandra_Morillejo_Gonz%C3%A1lez.pdf (accessed on 20 February 2018).
23. Ugur, A.; Sogukpinar, I. Multilayer Authorization Model and Analysis of Authorizations Methods. *Turk. J. Electr. Eng. Comput. Sci.* **2016**, *24*, 4915–4934. [CrossRef]
24. Xu, M.; Qin, Z.; Yan, F.; Fu, S. Trusted Commputing and Information Security. In Proceedings of the 11th Chinese Conference, CTCIS 2017, Changsha, China, 14–17 September 2017; Springer: Berlin, Germany, 2017; pp. 397–408.
25. Hernandez, J.; Martinez, F.; Keogh, J. User Management and Security in SAP Environments—Chapter 8. In *SAP/R3 Handbook*; Mc Graw Hill: New York, NY, USA, 2006; pp. 351–399. Available online: http://cdn.ttgtmedia.com/searchSAP/downloads/chapter-february2.pdf (accessed on 24 March 2018).
26. Vacca, J.R. *Computer and Inforrmation Security*; Morgen Kaufman: Burlington, MA, USA, 2017; pp. 13–33, 391–418, 1041–1044. ISBN 978-0-12-803843-7.
27. Kagermann, H.; Kinney, W.; Kuting, K.; Weber, K.L. *Internal Audit Handbook. Managing with the SAP-Audit Roadmap*; Springer: New York, NY, USA, 2008; ISBN 978-3-540-70887-2.
28. Eric, S.; Noblet, J.P. Integrating ERP into the Organization: Organizational Changes and Side-Effect. *Int. Bus. Res.* **2012**, *5*, 51.
29. Kirchmer, M. *Business Process Oriented Implementation of Standard Software: How to Achieve Competitive Advantage Quickly and Efficiently*; Springer: New York, NY, USA, 2012; ISBN 978-3642977176.
30. Oracle Database Security for SAP Applications. 2017. Available online: http://www.oracle.com/us/products/database/nl20-database-security-396167.pdf (accessed on 30 May 2018).
31. Oracle Solution Center. Oracle Solution for SAP Environments. White Paper. 2014. Available online: http://www.oracle.com/us/solutions/sap/oracle-security-for-sap-2148703.pdf (accessed on 20 January 2018).
32. Kyte, T.; Kuhn, D. *Expert Oracle Database Architecture*; Kindle, Ed.; APress: New York, NY, USA, 2014; ISBN 978-1430262985.
33. Gupta, S.K. *Advanced Oracle PL/SQL Developer's Guide*; Packt Publishing: Maharashtra, India, 2016; ISBN 978-1785284809.
34. Designing SAP Application Security. Leveraging SAP Access Monitoring Solution during SAP Implementations; Upgrades or Security Redesign Projects. 2017. Available online: https://www.protiviti.com/sites/default/files/united_states/insights/designing-sap-application-security-protiviti.pdf (accessed on 15 February 2018).
35. Wessing, D. A Case Study into the Application of a Good Practice SAP (Access) Security Design Framework. 2015. Available online: http://www.vurore.nl/images/vurore/downloads/scripties/2046-Def-scriptie--Dennis-Wessing.pdf (accessed on 29 January 2018).

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).