Security Methods and How They Are Applied in Oracle Products

University of Eastern Finland
Department of Computer Science
# Contents

1 INTRODUCTION 1

2 INTRODUCTION TO DATABASE SECURITY 4
   2.1 History of Database Security .......................... 4
   2.2 Relational Database Model .............................. 5
   2.3 Database Security ..................................... 8

3 DESCRIPTION OF ORACLE DATABASE VAULT PRODUCT 16
   3.1 Realms ............................................. 17
   3.2 Mandatory Realms ..................................... 18
   3.3 Configuration Controls and Separation of Duties ... 18
   3.4 Application Protection Policies ...................... 20
   3.5 Oracle Database Vault Command Rules and Factors . 20
   3.6 Oracle Database Vault Reports ....................... 21
   3.7 Installation ....................................... 21

4 PCI-DSS STANDARD 23
   4.1 Build and Maintain a Secure Network ................ 25
   4.2 Protect Cardholder Data ............................... 27
   4.3 Maintain a Vulnerability Management Program .......... 27
   4.4 Implement Strong Access Control Measures ............ 28
   4.5 Regularly Monitor and Test Networks .................. 28
   4.6 Maintain an Information Security Policy ............... 29

5 PCI-DSS COMPLIANCE CHECK FOR ORACLE DATABASE VAULT 30
   5.1 Requirement 2: Password Management .................. 33
   5.2 Requirement 3: Stored Data ............................ 33
   5.3 Requirement 6: Secure Systems and Applications ....... 34
   5.4 Requirement 7: Access Restriction ..................... 34
ABSTRACT

The purpose of this work was to show how Oracle Database Vault product implements security methods to protect sensitive data from unauthorized access by privileged users and show how these security methods help to comply with Payment Card Industry Data Security Standard (PCI DSS).

This work is meant for enterprise clients from Payment Card Industry, who have strict requirements to comply with a large set of security standards. This work helps such clients to identify level of Oracle Database Vault compliance with PCI DSS and plan their security strategy accordingly.

Amount of threats caused by unauthorized access from privileged accounts have increased lately and became a big security risk for organizations who is not yet considered to protect their data from internal access. In this Thesis we will check what kind of methods Oracle Database Vault has for protecting organization from internal threats.

Moreover, we will do a research to identify how Oracle Database Vault complies with widely known Payment Card Industry Data Security Standard. Organizations that handle cards from major card industry vendors should be compliant with PCI DSS. PCI DSS standard was created to put more control around cardholder data.

Case Study method was used as a research method to investigate Oracle Database Vault PCI DSS compliance. Phases approach was used to achieve desired results. The first phase included research of Oracle Database Vault product, it’s methods and features. The second phase consisted of PCI DSS standard study and deep understanding of each requirement. During the third phase were identified and excluded all not relevant to Oracle Database Vault requirements of PCI DSS standard. The fourth phase consisted of more deep investigation of relevant requirements and possible Oracle Database Vault compliance with them, as well as deeper investigation on Oracle Database Vault methods and concept to
support compliance with PCI DSS standard requirements. All relevant requirements were supported with security methods and features description of Oracle Database Vault for clear understanding compliance of the product with such requirements. Last phase of the research was done to provide general analysis on Oracle Database Vault results.

Requirement by requirement will be handled and investigated to check Oracle Database compliance with each of the requirements presented in PCI DSS. As a result we will get list of requirements that Oracle Database Vault complies with and explanations what security method helps to comply with each requirement. In a conclusion of this Thesis final results of this work and topics for future investigations will be presented.

Reference part of the Thesis is relatively small due to specific topic of the Thesis. This Thesis is concentrated mainly on enterprise product of one company. Oracle Database Vault is supplied with Oracle Database Enterprise Edition, which means it is not publicly available. Thus materials that can be used are very limited, as well as they all are located on company’s official internet site.
This Thesis has taken many years of work from 2006 to 2015. It went through different stages and changes, but finally has come to logical end. During all these years many people have been participating in moving my reseach further directly or indirectly. I would like to thank everyone who helped me on my long way to the finish.

Special thanks to University of Eastern Finland staff for support and understanding, together with prompt help in all matters.

I would like to thank my supervisor Dr. Keijo Haataja and examiner professor Pekka Toivanen for their cooperation, their rapid help and their valuable comments and suggestions.

I would like to express my dearest thanks to my family for endless support, patience and believe in me. Everytime when I felt that I am stuck or I have no energy to proceed they motivated me and supported. Without my family support I would never been able to be where I am now. My biggest thanks goes to my husband Alexey and son Petr. My husband has provided very valuable comments and he has shown patience and understanding. My son has added fun to my life by being so happy.

Helsinki 2015
Maria Korsakova
ABBREVIATIONS

ASV    Approved Scanning Vendor.
CRSF   Cross-site Request Forgery.
DAM    Database Activity Monitoring.
DAMP   Database Activity Monitoring and Prevention.
DBA    Database Administrator.
DB     Database.
DBMS   Database Management System.
DDL    Data Definition Language.
DMZ    Demilitarized Zone.
DNS    The Domain Name System.
DoS    Denial of Service Attack.
GPRS   General Packet Radio Service.
IDMS   Integrated Database Management System.
IMS    IBM Information Management System.
IDS/2  Integrated Data Store.
IP     Internet Protocol.
IPS    Intrusion Prevention Systems.
IT     Information Technology.
LAN    A Local Area Network.
NAT    Network Address Translation.
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>Operating System.</td>
</tr>
<tr>
<td>OTN</td>
<td>Oracle Technology Network.</td>
</tr>
<tr>
<td>PAN</td>
<td>Primary Account Number.</td>
</tr>
<tr>
<td>PCI-DSS</td>
<td>Payment Card Industry Data Security Standard.</td>
</tr>
<tr>
<td>PIN</td>
<td>Personal Identification Number.</td>
</tr>
<tr>
<td>RADIUS</td>
<td>Remote Authentication and Dial-In User Service.</td>
</tr>
<tr>
<td>SDP</td>
<td>Site Data Protection Program.</td>
</tr>
<tr>
<td>SSH</td>
<td>Secure Shell.</td>
</tr>
<tr>
<td>SYSDBA</td>
<td>System Database Administrator.</td>
</tr>
<tr>
<td>TACACS</td>
<td>Terminal Access Controller Access Control System.</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator.</td>
</tr>
<tr>
<td>WEP</td>
<td>Wired Equivalent Privacy.</td>
</tr>
<tr>
<td>XSS</td>
<td>Cross-Site Scripting.</td>
</tr>
</tbody>
</table>
1 Introduction

Utilization of databases in everyday life has dramatically increased in recent years which makes database security a very important topic. We are using databases in our everyday life, but most of the services provided in Internet, in shops, or at work usually rely on databases. The increasing amount and type of data stored in databases makes the information that they contain more sensitive to illegal access. As IT market grows, database security will become more critical issue. Forms of malicious attacks and many different ways of intercepting sensitive data have changed in the recent years. New methods of security are required to protect sensitive data from unauthorized access.

It is common to consider viruses, worms and hackers as the major risk source for the database security. However, nowadays, the most potential threats are considered to be the outside treats like viruses, worms, and hackers, but potential danger can come from much closer and unexpected place. Nowadays privileged database accounts are very common way to access sensitive data stored in the database. Unauthorized access by privileged users has became a very big threat for organizations owning large volumes of sensitive data.

This Thesis will describe quite unique product in current market, which provides powerful security methods to protect access of sensitive data by privileged accounts. It helps to implement security solution to prevent unauthorized access and helps to comply with data security regulations [1].

As Oracle Database Vault is a database security product it is crucial for it to comply with security standards including PCI-DSS standard to protect sensitive information in database such as cardholder’s data. The purpose of this Thesis is to check if Oracle Database Vault is compliant with PCI-DSS standard and describe how compliance in Oracle Database Vault product is implemented.
Case study research method is used in this Thesis to investigate Oracle Database Vault product compliance with Payment Card Industry Data Security Standard (PCI-DSS).

Case study is a research method widely used in education studies nowadays. Originally, it was introduced in social science, but lately it grew in popularity in other areas as well, especially in education and in teaching processes. There are many definitions of Case study. They all complement each other, but the most simple and clear from my point of view was defined by Robert K. Yin. He actually highlighted the strength of this method and thereby pointed out main purpose of the case study: “Compared to other methods, the strength of the case study method is its ability to examine, in-depth, a “case” within its “real-life” context” [11].

Case study is looking into details, into depths of the case without trying to generalize from it. According to Robert K. Yin there are two general situation when you can easily detect that case study will help you in your research. First one is when such questions like “what?”, “why?”, “how?” arise, and “what happened, why and how?”.

Second situation is when you want to highlight some certain situation and understand it more deeply. Examples for these situations can be investigation of a person who lived especially long life, because you would like to know how could this person live for so long. Another one can be a study about colleague who became a director of the company and you would like to know how and why.

In all these situations case study can help you.

When you do research using case study method it will not be much difference if you would use other methods. All research methods would require collecting and reviewing information data analysis, and final reports, but there is one thing that makes it different in case study research. In some cases, it might be required to do data collection and data analysis together for adjustment in case of inconsistency found in research.

Good example of such situation would be if you have collected
some data earlier from one source and then later on another source provides you a contradictory information. In this case you would need to analyze data and check first source again to get correct information and resolve conflict.

This Thesis consists of the following sections. Section 2 describes database security history, relational model, and makes an introduction to database security. Section 3 describes Oracle Database Vault product, its unique features, and benefits of applying such product in organizations. Throughout this Thesis focus will be on Oracle Database Vault version 12c. Section 4 introduces the PCI-DSS standard and explains in details each standard’s requirement. Section 5 shows how Oracle Database Vault helps to comply with PCI-DSS standard. Section 6 concludes the Thesis and sketches future work.
2 Introduction to Database security

Section 2.1 will introduce to database security history. Section 2.2. will describe briefly a relational model, that is used as a basis for all relational databases. Section 2.3 will tell about database security and most common threats against database security.

2.1 HISTORY OF DATABASE SECURITY

IT technologies have improved in the area of the computer’s performance, this led to a fast growth in the performance and the abilities of databases. Several database technologies can be highlighted as most commonly used. They are based on different data models. Three technologies can be presented: navigational, SQL/relational, post-relational.

An example of navigational models can be taken hierarchical model, which was developed by IBM and used in IBM Information Management System (IMS), Windows Registry by Microsoft, the Codasyl model and so on [2].

The relational model was introduced by Edgar Codd during the time when he was working in IBM. He proposed the idea that data should be searched by content. The relational model started to spread widely in 90s, as computer’s hardware became powerful enough to run relational systems. SQL language is commonly used for relational databases. Most of the modern databases use relational model as a database foundation including Oracle database [2].

The third approach is post-relational. This category had different stages of development starting from object-relational databases, NoSQL databases, and continue with NewSQL databases. They
are based on relational model, but trying to achieve as high performance as NoSQL databases [2].

In this Thesis we are going to work only with relational databases, as they are taking a leading position at current market situation.

2.2 RELATIONAL DATABASE MODEL

Relational Database Model is a foundation for all relational databases. As we are going to look at relational database, it is worth to be briefly mentioned in this Thesis.

Relational Database Model is widely used in databases, including Oracle Database. Databases implementing such model are called relational databases. First time it was formulated and presented by Edgar F. Codd in 1969-1970. Among others who maintained and developed relational model were Chris Date and Hugh Darwen. They wrote big amount of books and gave a lot of lectures related to relational theory.

The SQL data definition and query language is used in most relational databases. SQL database’s implementation can be considered as an approximation to relational model. Even though SQL is initially positioned as a standard language for relational databases, it has several deviations from classical relational model, such as Duplicate rows, NULL, Duplicate column names, Columnless tables unrecognized and etc.

Relational model has been implemented by using mathematical discipline. Main components of the original model are structure, integrity and manipulation.

The structural feature of relational model is a relation. This relation is commonly represented as a table. Relations are defined over types. Types can be defined as a pool of values from where attributes of a relation take their values. Table representing a n-ary relation, where n is any non-negative integer value, columns refers to an attributes and rows to tuples. [3]
Every relationship should have at least one candidate key. It is a unique identifier consisting of one attribute or combination of attributes and uniquely identifying tuples in a relation. Primary key is a candidate key, which has been selected for some special reasons and defined for some special treatment. It doesn’t differ from candidate key, especially when relation has only one candidate key. In case of many candidate keys existence, usually primary key is chosen from one of them to make candidate key to be preferred. Foreign key is a set of attributes in one relation, which should match values of candidate key in other relation. [3]

The integrity feature (integrity constrain) is a boolean expression that must evaluate to TRUE. In relational model exists by default two generic integrity constraints. [3]

One constraint related to primary key and it says that primary key attributes don’t permit nulls. It means that primary key can’t have unknown value, because otherwise it is impossible to identify this tuple in relation.

Another constraint related to foreign key and it says that there must not be any unmatched foreign key values. It means that it should always exist an equal value of corresponding candidate keys for every foreign key value.

The manipulative feature consist of two parts: relational algebra and a relational assignment operator. Relation algebra is a set of operators, which can be applied to relation. Relational assignment operator permits the values of some relational expression to be assigned to some relation. [3]

Relational algebra has a set of operators, which allows to take one or more relations as an input and produce another relation as an output. Figure 4.1 is a visual representation of such operators: [3]

- **Restrict** returns a relation with all tuples which match specified condition.

- **Project** returns a relation with all tuples that remains after specified attributes has been removed.
• **Product** returns a relation containing all possible combination of two tuples, one from each of two specified relations.

• **Intersect** returns a relation with all tuples that exist in both of two specified relations.

• **Union** returns a relation with all tuples that exist in both of two specified relations.

• **Difference** returns a relation with all tuples that exist in first relation and doesn’t exist in second specified relation.

• **Join** returns a relations containing all possible tuples that are a combination of two tuples, one from first relation and second from second relation. These two tuples has to have a common attributes with a common values of the two relations.

![Visual representation of operators.](image)

One should always remember that data model is a foundation. Implementation is a physical realization on machine components of that abstract model and it can have distinctions with this data model.
Relational Database Systems are very complex. Process management, disk management, file management, multi-threading, and user administration are common features resembling operational system’s features that can be performed on current Relational Database Systems. Managing of database instance, metadata, users, profiles, and schemas are features needed for relational database engine. Relational database vendors policy before 90s were providing very strong security at core level of database and its modules. Around late 1990s vendors have changed their policy and provided more security features at operating system level. Nowadays it is a combination of two ways and it is following multi-level security strategies. [4]

2.3 DATABASE SECURITY

Amount of information available through different resources are increasing and protection of such information becomes very important topic to people, responsible for administering this data. Potential threats to database security include data integrity, availability, and secrecy.

Data integrity requires that data inside database will be protected against any kind of unplanned modifications. The integrity violation happens through unauthorized changes done to data with intentional or accidental operations. Those kind of operations can be either data modification, insertion, or deletion. [5]

Data availability refers to the requirement that data are available to users and applications that have the legitimate right and authorization to access or modify the data. Loss of data availability can result in the unavailability of services and applications that operate on the database. [5]

Data secrecy stands for sensitive data protection from being disclosed by unauthorized access. An inappropriate actions can cause reputation and revenue losses, in some cases even legal actions to company due to business intervention. [5]
Security risks may vary and depends on different conditions. Most common threats can be different at different times. Security risk information always needs to be up to date. Here are listed latest found vulnerabilities.

Top 10 Database Vulnerabilities and Misconfigurations: [6]

- **Default, Blank & Weak Username Password**: Common practice on product market is to have some default users and passwords for different software. Product can create default password in a background for easy start with a software, which creates potential risk on security. Basic security steps should be taken to prevent stealing sensitive data like customization of credentials and creating strong password.

- **SQL Injections in the DBMS**: Usually happens through SQL commands. It can be a parameter of a function in SQL command or it can be a stored procedure parameter. Usually, SQL injection is done with system or administration privileges. SQL injection vulnerability can be avoided by applying latest patches to database.

- **Excessive User & Group Privilege**: This vulnerability refers to concept of assignment excessive privileges to users or user groups. Following the concept of assigning to one task more than one user and not giving excessive privileges to users can help to ensure that users will not misuse their privileges.

- **Unnecessary Enabled Database Features**: Database Management Systems have grown and became quite complex software system. These systems have a big variety of analyzing and reporting features, they support many languages and different Operating Systems access levels. As in many cases power instrument have two sides, one of them a good one, that application developers have more features and possibilities, but there is a down side, that there are more weak points, which can potentially be used by hackers. There are various
features, which should be checked. Disabling of these features can make DBMS more secure, except the cases when there is a good business reason to have them available.

- **Broken Configuration Management:** At earlier days, Database Management Systems haven’t had many configurations. The options available to DBA were pretty simple. Today’s picture has changed. Database Management Systems have plenty of different options, which might have direct or indirect impact on security. Configuration settings need to be correctly configured. This makes your critical business information protected and secure.

- **Buffer Overflows:** It happens when function’s input is not checked during the copy and it exceeds the volume that buffer can hold. This leads to cases when memory, meant for other uses, can possibly be overwritten. In such cases behavior can be unpredictable and can lead to server crash. It is very important to apply latest patches from database vendors and monitor your system for known attack signs to prevent these types of attacks.

- **Privilege Escalation:** Such attack happens when person who attacks uses known Database Management System vulnerabilities to execute instructions or query data using user account with low privileges, when in typical cases this kind of executions would require higher privileges. Different common vulnerabilities exist that makes privilege escalation attack possible. Sometimes, vulnerability is used to allow account with low level privileges to grant additional rights to itself. In some cases vulnerability is used by misusing a function that runs under system administrator (sysdba). As in many previous attack cases, the best practice is to apply latest vendor’s patches and monitor your system for known signs of attack.
• **Unpatched Databases:** Most big vendors of databases provide patch updates on regular basis. Usually, it happens every quarter. Sometimes third-party companies disclose vulnerabilities of databases whether vendor itself have patch to fix them or not. If patch is released, it should be used, otherwise hackers will find a way to use weaknesses to attack the database. Database will become more vulnerable after patch is released than before. Many companies delay in applying patches with common excuses like difficulty in testing new patches and downtime involved in patching process, but whatever excuse is used company should trace database activity more carefully between patches.

• **Denial of Service Attack (DoS):** Denial of Service Attack known as DoS is when Internet traffic on site or server gets overwhelmed, so site or server cannot function properly anymore and shuts down. Usually the main cases, when this attack is possible, are when database is left unpatched. The latest patches from database vendors should always be applied to avoid potential security risks.

• **Unencrypted sensitive data at rest and in motion:** Very important part in security is encryption, if you want to keep your sensitive data safe and unrevealed. Network traffic must always be encrypted to assure any information goes through securely and cannot be seen by eavesdroppers. Some database management systems may permit for some sensitive data to be transmitted as a clear text. As a preventive method you have to download latest software version and switch off text indexing.

Database security includes big variety of information security controls to protect database from potential risks. These information security controls might protect database application, database server, database data, and stored functions. The information security controls could be technical, procedural and physical.
Databases can have many information security layers, which are appropriate for database control, such as: Access control, Encryption, Auditing, Integrity controls, Authentication, Backups, Application security, and Database Security applying Statistical Method.

Usually, each vendor has recommendations and standards to follow to put control over their databases and increase security protection. These standards and guidances can provide security requirements, security policies, best practices, and security recommendations to follow. Guidelines normally specify other security related activities, such as design, configuration, development, use, maintenance of databases and management.

As many other vendors, Oracle also has own general guidelines and best practices documents to increase database security and to avoid common misconfigurations. One of such guidance document is called Oracle Database Security Checklist. It includes recommendations for protecting database environment. In this Thesis, Oracle checklist will be briefly presented. Please refer for a checklist to appendix A.(Full and more accurate checklist can be found on Oracle website). [7]

Hardening database is usually based on three main principles. The first principle involves locking access to important resources to prevent their misusage purposely or by mistake. Second principle involves disabling all unnecessary functions, which can be misused. And the third principle is a least privileges: Users, task and processes should be assigned with minimal required privileges for their normal functioning. [8] It is also good to remember that for protection of your critical data, strengthening of database alone is not enough. Security should be applied on all levels of your system.

Creating security layers helps to protect your critical data from unplanned modifications and potential disclosures. There are several techniques to follow for preventing different kind of attacks to succeed and to strengthen database.

Security layers strategy makes it harder for attacker to perform
an attack on environment, because it requires to make many things right at the same time. Attacker should find holes in all security layers at once, which should match each other to be able to succeed. Each security layer protects environment from different threats. Security layers could consist of such layers like authentication, administration, firewalls, authorization, VPNs, vulnerability assessment and patch management, intrusion detection and prevention, security management, application security, antivirus, database security, and encryption, which in fact includes aspects that belong to each layer. [8] Authentication, authorization, administration are also known as 3A. They are responsible for identifying who is trying to attempt access to the resource and ensures this person is authorized to access such resource. Administration software is centralizing management and administration of privilege and permissions.

Firewalls main principles are to keep unauthorized users off corporate network. They are hardening corporate network perimeter and protecting connection to Internet and extranets.

Virtual Private Networks (VPN) allow secure remote and mobile access to internal corporate network. VPN is nowadays presented in most of the organizations and it allows workers to work from home, work in remote offices, and when they travel. Intrusion detection and prevention refer to threats that happens inside internal network. They are based on communication stream’s deep inspection and on patterns of attack. They trace any deviation from normal behavior, which looks like a malicious event. One of the technique to evaluate database vulnerabilities is to perform vulnerabilities assessments against the database. Mainly it is done by testers group, who tries to find weak points of database by attempting to break into database, bypass security, and check common system vulnerabilities. Database administrators can perform automated vulnerability scan to find out misconfigurations, possible weaknesses in network’s security, common system vulnerabilities and so on. The result of vulnerability scans can be used to increase security of the database and get rid of found vulnerabilities.
Security is a complex and mandatory issue. There are many software products that can help to manage the process and centralize relevant information. They help to manage security systems, incident response systems, and reporting. Antivirus is a security layer that serves a protection of users from malicious code such as Trojans, viruses, and worms. Many types of antivirus software exist like antivirus in email, network antivirus, desktop antivirus, and so on. Application security in fact goes very tight with database security. Almost all application data is stored in relation databases and protecting application data in most cases mean protecting database data. Encryption is helping to create additional security layer to guarantee confidentiality and low risk in case when all of your other layers were overcome by hacker. [8]

Often, despite the need to create security layers, willingness to invest into security is not always high because of limited budgets. For those people who couldn’t realize importance of security investments, regulators created a set of regulations. The aim of these regulations is to enforce protection of information. Another important part of database security is database auditing. There are many auditing categories from which you can select and implement auditing trails according to your needs. Auditing categories options can be a audit of logon/logoff, audit source of database usage, audit database usage outside normal operating hours, audit DDL activity, audit database errors, audit changes to source of stored procedures and triggers, audit changes to privileges, user/login definitions, other security attributes, audit creations, changes, usage of database links and replication, audit changes to sensitive data, audit SELECT statements for privacy sets, and audit any changes made to the definition of what to audit. [8]

Implementation of these audit categories can be done using existing database audit capabilities or using independent database security technology, such as DAM(Database Activity Monitoring) that does not rely on native database auditing.

Database activity monitoring (DAM) is a database security tech-
nology for monitoring and analyzing database activity that operates independently of the database management system (DBMS) and does not rely on any form of native (DBMS-resident) auditing or native logs such as trace or transaction logs. DAM is typically performed continuously and in real-time. Database activity monitoring and prevention (DAMP) is an extension to DAM that goes beyond monitoring and alerting to also block unauthorized activities [2].

Another important security layer is database activity monitoring, which is usually done in real-time and continuously [2].
3 Description of Oracle Database Vault product

Section 3.1 will introduce the concept of Realm and how it helps to secure Database. Section 3.2 will introduce concept of Mandatory Realms and how they help to protect Database during maintenance tasks. Section 3.3 will describe Configuration Controls feature of Oracle Database Vault. Additionally Section 3.3 will explain how separation of duties concept is implemented in Oracle Database Vault. Section 3.4 will describe Application Protection Policies feature of Oracle Database Vault. Section 3.5 will introduce Command Rules and Factors concept of Oracle Database Vault. Section 3.6 will describe Report possibilities in Oracle Database Vault. Section 3.7 will describe installation process of Oracle Database Vault.

Most of the existing database monitoring systems provide DBA ability to trace database’s activity. It provides ability to follow who did what, when, and what kind of data were accessed. Also, it is possible to create reports and alerts, if an attempt is made to access sensitive data or perform a prohibited action on database.

However, monitoring systems do not always allow to trace and block prohibited activity on the fly, including activity of privilege accounts like DBA, even though privilege accounts are the most common path to access wide area of sensitive data in database. Prohibited activity does not always mean malicious attempt, it can also be a DBA curiosity or mistakes of junior DBA, which can create vulnerabilities in database.

Whatever reason for this activity Oracle Database Vault helps to create secure environment by restricting access of privilege users to sensitive data (such as financial information, credit card information and etc.) By using Oracle DB Vault you can harder you Oracle
DB instance, protect your data from privilege users while still allow them to perform their everyday duties and help to apply best practices as well as comply with regulations from multiple countries.

ODV has a lot of features to increase database security. It can help to prevent privileged users that are trying to access sensitive data from accessing it. It has a feature that can “freeze” database configurations. ODV can protect sensitive objects by blocking access to them during threats or maintenance. ODV has a feature to help identify unused privileges and prevent their misusage. Please, refer to appendix B for more accurate list of Key Features and Benefits of Oracle Database Vault. [9]

Oracle DB Vault together with Oracle DB provides protection against many common threats, such as: threats of lost credentials, inside threats, that are based on privileges misusage, threats that can happen during database maintenance and much more. Please, refer to appendix C for more accurate list of common threats that Oracle Database Vault provides essential safeguards against. [9]

3.1 REALMS

Most common audit element is a control for unauthorized changes to database grants and creations like grants of DBA role, creation of database objects, and new accounts. Unauthorized changes can weaken database and create easy path for hacker to make a malicious attempt and they are also violating privacy and compliance regulations. To increase control on privilege accounts, Oracle Database Vault is able to create inside Oracle DB very restricted application environment Realm. A realm can include database schemas, objects, and roles that must be secured. By functional grouping of these schemas and roles, Realm gives you a control of use of system privileges to access these functional groups. Oracle Database Vault is able to block unauthorized DBA access to application data, which helps to protect from insider’s unauthorized
access as well as outsider’s threats. Oracle Database Vault enables secure consolidation. It allows to consolidate applications into one database and it ensures that privileged applications can’t access each other’s data. Oracle Database Vault can also control who can manage Virtual Private Database policies to achieve even higher application security. [9]

3.2 MANDATORY REALMS

Mandatory realms were created for the need to block access to highly sensitive data during maintenance task performed by DBA or IT support. Production environment usually requires access for the patching, diagnostics, and other activities reasons. During such access, all tables and views with highly sensitive data can be blocked with Mandatory realms and even people who has direct object grants or application owners can’t access sensitive data, only in case if access has been granted specifically. Mandatory realms are configured before maintenance tasks and can be enabled during performance of such maintenance. Mandatory realms are able to keep unchangeable rights granted to database role and none of the privileged users will be able to modify them, except authorized users. In case of violation Mandatory realm can protect data by blocking access to data to anyone even for owner and for users with direct grants. [9]

3.3 CONFIGURATION CONTROLS AND SEPARATION OF DUTIES

One of the important Oracle Database Vault’s feature is to control of commands usage such as ALTER SYSTEM, ALTER USER, CREATE USER, DROP USER, and so on. It allows customers to have a strong operational controls inside their database. It helps to prevent undesirable changes to database configuration that could lead to configuration inconsistency, insecurity, and compliance violation.
Additionally, Oracle Database Vault can control SQL commands to increase availability and security of the database and applications. It is achieved by applying additional checks and rules before execution of any SQL command. SQL command controls provide a big variation of restrictions and limitation to achieve desired results in security beginning from access restriction to database for some certain subnets, programs, and application servers ending with using some functions and factors such as session user name, Oracle Label Security factors, host name, Oracle APEX applications’ native functions and factors, and IP address. [9]

Oracle Database Vault adds possibility to separate Account Management role from DBA account and assign it to a new role, and thereby implement separation of duties. This implementation gives more possibility to control account management operations such as create user and change password. If used together with additional factors like IP address, time and etc. increases security to even higher level. [9]

Mandatory realm can prevent drift in database roles and configurations by freezing database roles settings. It prevents any grants or revokes possible inside database. Realm can protect database role from misuse and prevent privilege grant by unauthorized users. Oracle Database Vault has a new feature called Privilege Analysis. It serves the purpose to trace use of roles and privileges based on actual usage and identify privileges and roles that are not in use. As well it helps to identify roles and privileges of actual usage. Knowledge of roles and privileges usage helps to understand the minimum number of privileges application require to perform. Implementing the least privilege strategy helps to increase application security and reduce risks. Privilege Analysis feature can be also used on development stage to identify privileged use for the new application. Later on application and user privileges could be corrected according to the least strategy approach. [9]

Cloud environments can provide many benefits such as effi-
ciency, flexibility, cost reduction, and faster results, but at the same time they bring a security risk due to bigger amount of applications, data, and users accessing the same database. Oracle Database Vault helps to benefit from having cloud environments while protecting such environment’s sensitive data. Oracle Database Vault implements Separation of Duty for different administrative groups. It can prevent privileged users and applications from accessing other’s applications data. It protects database’s configuration from modifications and restricts account management. All these in complex are secure database, protect from internal treats, harden database to prevent vulnerabilities. Oracle Database Vault can also be deployed to Oracle Exadata machine together with other Oracle products to achieve even more secure environment. Also it can be used with pluggable databases to prevent access to sensitive data by privileged users. [9]

3.4 APPLICATION PROTECTION POLICIES

Application Protection Policies are meant to protect application’s sensitive data. It is done by creating realms around application’s sensitive tables or around full schema. Oracle has a predefined policies for some Oracle applications as well as for a partners applications. Oracle continue to work on application policies for another applications. Policies for application can be always checked from Oracle resources such as Oracle Support and OTN (Oracle Technology Network). [9]

3.5 ORACLE DATABASE VAULT COMMAND RULES AND FACTORS

ODV command rules can be used to implement restriction of who can execute ‘ALTER SYSTEM ’, ‘SELECT’, and data manipulation language statements. The command rule protects its SQL statement from anyone who is trying to execute protected SQL statement, even
if object is located in different realm. For the command rule to take effect, you need to associate it with the rule set. A rule set is a collection of rules (could be just one or more). The rule set is a PL/SQL expressions that can take the value either true or false and final value of the whole rule set will be true or false based on each such PL/SQL expression and evaluation type ‘all true’ or ‘any true’.

Oracle Database Vault has also factors, which are attributes or named variable. Factors can be attributes such as database IP address, hostname, and domain, as well as user’s session, location, and so on. Factors can be used as an additional authorization parameter for database account connecting to database. Also, factors can be used as a filters for data access restrictions.

Oracle Database Vault Reports can be run on command rules, factors, rule set for realm violation recognition, auditing results, and for finding configuration problems [10].

3.6 ORACLE DATABASE VAULT REPORTS

ODV Reports is able to present incidents related to attempt of getting access to protected application tables, blocked SQL statements, and configuration changes made by administrator including changes to Oracle Database Vault itself. Oracle Database Vault will prevent access to application’s protected tables and will add audit record about this incident. Such audit reports can be viewed by Real Audit Report. Oracle also provides possibilities to view ready-made views of runtime analysis available in Oracle Audit Vault, Oracle Enterprise Manager Cloud Control 12c, and Database Firewall. [9]

3.7 INSTALLATION

Oracle Database Vault doesn’t require any additional installation or reconfigurations. It is delivered together with Oracle DB 12c and can be switched on simply from command line. After enabling it requires only database restart, then ODV becomes active. It can be
used in highly available architectural solutions. Oracle Database Vault is providing low performance overhead and according to customer reports, major production applications have no changes in application’s response time [9]. Everyday DBA operations can be done in normal way until DBA needs to do something with sensitive data. In this case DBA will need an authorization before any operation can be done with protected data. For additional information and best practices please refer to Oracle white papers. Oracle Database Vault controls will still remain active even if database files are exported or restored to another Oracle home environment [9].

ODV is a security product for protection database sensitive data from internal and external threats. Oracle DB controls can be used for high security requirements implementation. Additionally, Oracle Database Vault supports cloud computing and consolidation. Oracle Database controls can be preconfigured and enabled when needed.
4 PCI-DSS Standard

Sections 4.1 to 4.13 will introduce the 12 Requirements of PCI-DSS standard.

The Payment Card Industry Data Security Standard (PCI-DSS) is well known information security standard. It is developed and administered by the Payment Card Industry Security Standards Council (PCI-SSC) for the organizations that need to work with credit cards and store, process, and transmit card information.

Due to the wide spread of credit payment systems, PCI-DSS standard has become one of the most important standard in the world. It was created to increase safety of cardholder’s data and protect this data from threats.

Originally, it is started to form in 5 different shapes: Visa, MasterCard, American Express, Discover and JCB began each their own security programs to create minimum amount of security requirements for merchants who store, process and transmit card holder data. These 5 programs called Visa’s Cardholder Information Security Program, MasterCard’s Site Data Protection, American Express’ Data Security Operating Policy, Discover’s Information Security and Compliance, and the JCB’s Data Security Program.

Visa’s Cardholder Information Security Program mandated since June 2001. This program intended to protect Visa cardholder’s data and to maintain high security standards [12]. Same year Master Card launched its Site Data Protection program [13]. The SDP Program provides data security requirements and compliance requirements to protect MasterCard stored, processed and transmitted cardholder’s data. Soon other vendors developed their own security programs with same intention to implement and maintain efficient data security requirements to protect card holder’s data from exposure.

Multiple security programs have created big difficulties for the merchants and other organizations that process card data, as they
all would need to go through multiple validation of different vendor’s compliance. As well all payment card brands would suffer from need to validate multiple organization to check their compliance with security programs, when in fact many of security program requirements were quite similar. Arisen situation has led to the need to create a single unified security standard and single way of compliance assessment and validation. In 2004, The PCI-SSC (Payment Card Industry Security Standards Council) was created and as a result of cooperative effort of Visa, Master Card, American Express, Discoverer, and JCB the unified security standard, known as PCI-DSS, was published in order to create common industry security requirements. First version of PCI-DSS consisted of 12 requirements forming 6 logically related groups called control objectives. In this Thesis we will provide information about requirements of version 2.0.

The PCI-DSS Requirements are the Following: [12]

- **Build and Maintain a Secure Network group** contains 2 requirements:
  1. Install and maintain firewall configuration to protect card-holder data.
  2. Do not use vendor-supplied defaults for system passwords and other security parameters

- **Protect Cardholder Data group**:
  3. Protect stored data
  4. Encrypt transmission of cardholder data and sensitive information across public networks

- **Maintain a Vulnerability Management Program group**:
  5. Use and regularly update anti-virus software
  6. Develop and maintain secure systems and applications

- **Implement Strong Access Control Measures group**:
7. Restrict access to data by business need-to-know
8. Assign a unique ID to each person with computer access
9. Restrict physical access to cardholder data

- Regularly Monitor and Test Networks group:
  10. Track and monitor all access to network resources and cardholder data
  11. Regularly test security systems and processes

- Maintain an Information Security Policy group:
  12. Maintain a policy that addresses information security.

### 4.1 BUILD AND MAINTAIN A SECURE NETWORK

Requirement 1: The first requirement, describes different requirements of network security that organization need to comply with PCI-DSS. Network is a very first gateway of the organization for the outside world to get connection to internal organization resources and at the same time very first protection wall to protect organization’s information. Network security can’t be compromised when we talk about privacy and protection of the cardholder’s information. Network security softening or compromises can lead to serious consequences both in financial losses and in loss of organization’s reputation. Nowadays a lot of tools and methods exist to help organization secure their enterprise network, such as firewalls, anti-virus software, intrusion detection systems, and access control systems. Network security is just a part of layered security architecture approach where each layer is supporting other layer to achieve higher security protection and decrease possibility for an attack to succeed. Layered security approach is also called Defense In Depth. Below there is a picture with example of layered security approach. [14]

Requirement 2: Default application and component passwords can be a big threat and vulnerability for organization. Applications,
devices, and other components are always supplied with default passwords and default configurations. These default passwords and configurations are available in vendors’ documentation, which means they are publicly available for everyone who wants to know them. If these default passwords, configurations and settings are not changed during configuration process attackers can use this weak point in the organization and get access to cardholders’ data. So this PCI-DSS requirement is created to prohibit the use of such default passwords in the PCI environment and consists of recommendations how default settings should be changed. [14]
4.2 PROTECT CARDHOLDER DATA

Requirement 3: This requirement is considered as a key requirement in PCI-DSS standard. It is related to stored cardholders’ data protection, which is the key goal of this standard. Requirement 3 is trying to highlight the need for each organization that stores cardholder data to reduce cardholder sensitive data storage places and make sure that such sensitive data are stored only if it is really necessary. Also it provides guidance and requirements how to handle encryption keys that are used to encrypt stored cardholder data. [14]

Requirement 4: Previous requirement provides guidance on how to protect data that are static, means it is just stored somewhere. This requirement describes requirements for protection of moving data that transfers through network including public networks by using encryption. [14]

4.3 MAINTAIN A VULNERABILITY MANAGEMENT PROGRAM

Requirement 5: Antivirus software is very important part of environment protection. Not in time antivirus software update can provide a wholes in security for the attackers. Especially it is very important, because nowadays malware threats are very common and can lead to fails and compromises of applications and system. This requirement describes how to handle antivirus software. [14]

Requirement 6: Applications are the necessary part of any organization in payment card industry. Applications are used in e-commerce, in service providers’ organizations and in many others that are part of the payment card industry. This requirement details a secure management processes for secure software development lifecycle and secure practices for applications. [14]
4.4 IMPLEMENT STRONG ACCESS CONTROL MEASURES

Requirement 7: Information restrictions should be taken very seriously and implemented with deep understanding of compliance scope. Individual’s access to PCI-environment should be controlled and maintained carefully with good understanding of best security practices and their implementation. This requirement describes such access control best practices for organization environment. [14]

Requirement 8: This requirement continues to guide through access control topic we started in previous requirement, but in this requirement emphasis on more specific implementation requirements for authentication, user management and passwords requirements. [14]

Requirement 9: This requirement is addressing physical security that are also important part of any security program. The requirement specify physical security media management requirements such as tapes, cd, hard drives etc. and, physical security such as monitoring systems, video cameras, visitor badges, access cards and so on. Security media management also includes secure procedure for destruction procedures. Improper destruction procedures can lead to attacker get access to data that has not been properly destroyed. [14]

4.5 REGULARLY MONITOR AND TEST NETWORKS

Requirement 10: This requirement related to auditing of access to resources of PCI environment. Audit information provide important information of system usage. It can help to understand how system is used or identify unusual behavior of the system. This requirement specifies log management rules, logging practices and logging data protection. [14]

Requirement 11: This requirement mandates to perform security assessment for of PCI environment to be made by internal and external resources. External vulnerability assessments should
be performed quarterly by Approved Scanning Vendor (ASV). It also dictates the penetration testing annually by internal and external resources, usage of network intrusion detection and/or intrusion prevention techniques to detect and prevent intrusions into network. One of the primary methods to measure the effectiveness of security controls in an environment is to perform security testing against the infrastructure. [14]

4.6 MAINTAIN AN INFORMATION SECURITY POLICY

Requirement 12: All previous requirements had technical context. This requirement shows the need to have operational security procedure to ensure that all technical requirements are taken into account, implemented security practices are efficient and risk assessments are performed on regular basis. [14]

The last requirement is the following: *Shared hosting providers must protect the cardholder data environment*

Requirement A.1, Protect each entity’s (that is merchant, service provider, or other entity) hosted environment and data, per A.1.1 through A.1.4. A hosting provider must fulfill these requirements as well as all other relevant sections of the PCI DSS. [14]
Section 5.1 to 5.7 will describe Requirements that are related to Oracle Database Vault compliance with PCI-DSS standard and information how Oracle Database Vault comply with these Requirements.

PCI-DSS is the most commonly known security standard and is the most important in our everyday life. I believe almost everyone has deal with credit cards and use them on daily basic. We come to supermarkets and pay for our food and goods with our credit cards. We are more and more visiting websites and are ordering goods from internet. Privacy of our sensitive information and protection of this information directly concerns all of us. We all can become victim of data breach in the organization that maintains and stores our sensitive data. That means that we all would like organization that handles, transfer or stores our data is taking PCI-DSS compliance very seriously.

Oracle is a huge IT company, that provides market with innovative and efficient software products for many years, including products delivering stronger security to the organizations that are ready to implement serious security programs. One of such product is an Oracle Database Vault. In previous chapter we already got introduced to this product and now we came to the main part of this Theis. In this chapter we will go through PCI-DSS requirements and will check Oracle Database Vault compliance with this standard.

First of all Oracle Database Vault is not concerned about all PCI-DSS requirements and it will be more logical, from my point of view, if we exclude requirements that are not related to Oracle
Database Vault and then consider in details requirements that are directly related to our product.

Due to similarity validation issues all requirements description were moved to appendix. Please, refer to appendix D for requirements specification.

Requirement 1: First requirement is related to different requirements of network security that organization need to comply with PCI-DSS. Oracle Database Vault product is not related and does not cover such security area.

Requirement 2: Related.

Requirement 3: Related.

Requirement 4: This requirement is related to encryption of transmitted data, which is not an Oracle Database Vault security area. It is not related to our product.

Requirement 5: This requirement is related to anti-virus software maintenance. It is not related to Oracle Database Vault.

Requirement 6: Related.

Applications are the necessary part of any organization in payment card industry. Applications are used in e-commerce, in service providers’ organizations and many others that are part of the payment card industry. This requirement detailing a secure management processes for secure software development lifecycle and secure practices for applications.

Requirement 7: Related.

Information restrictions should be taken very seriously and implemented with deep understanding of compliance scope. Individual’s access to PCI-environment should be controlled and maintained carefully with good understanding of best security practices and their implementation. This requirements describes such access control best practices for organization environment.

Requirement 8: Related.

This requirement continue to guide through access control topic we started in previous requirement, but in this requirement emphasis on more specific implementation requirements for authentica-
<table>
<thead>
<tr>
<th>Requirement</th>
<th>Relevant or not</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement 1</td>
<td>No</td>
</tr>
<tr>
<td>Requirement 2</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 3</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 4</td>
<td>No</td>
</tr>
<tr>
<td>Requirement 5</td>
<td>No</td>
</tr>
<tr>
<td>Requirement 6</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 7</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 8</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 9</td>
<td>No</td>
</tr>
<tr>
<td>Requirement 10</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement 11</td>
<td>No</td>
</tr>
<tr>
<td>Requirement 12</td>
<td>No</td>
</tr>
<tr>
<td>Requirement A.1</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 5.1: Table of PCI-DSS Requirements relevant to Oracle Database Vault

...tion, user management and passwords requirements.

Requirement 9: This requirement is related to physical security and not related to Oracle Database Vault.

Requirement 10: Related.

This requirement related to auditing of access to resources of PCI environment. Audit information provide important information of system usage. It can help to understand how system is used or identify unusual behavior of the system. This requirement specify log management rules, logging practices and logging data protection.

Requirement 11: This requirement is related mandatory of security assessment for of PCI environment. It is not an Oracle Database Vault area.

Requirement 12: This requirement is related to operational security procedures and not related to Oracle Database Vault product.

Requirement A.1: Related.
We went through all requirements and found out that requirements 1, 4, 5, 9, 11, 12 are not related to our product compliance area, while requirements 2, 3, 6, 7, 8 and Appendix are related to Oracle Database Vault compliance area. Let’s go through each requirement one by one and check Oracle Database Vault compliance with these requirements.

5.1 REQUIREMENT 2: PASSWORD MANAGEMENT

Requirement 2.1,
Oracle Database Vault: Passwords for Oracle Database Vault administration accounts are prompted for during installation. Also, Oracle guidelines to use secure password based on Oracle Database Security Guide. Oracle Database Vault has additional password requirements that will be displayed during password creation process if password does not comply with these requirements.

Requirement 2.2.3,
Oracle Database Vault: Oracle Database Vault implements separation of duties concept. This concept helps to prevent unauthorized administrative actions in the Oracle Database. (For the separation of duties concept, please refer to chapter Description of Oracle Database Vault product)

Requirement 2.6,
Oracle Database Vault: See Appendix A.

5.2 REQUIREMENT 3: STORED DATA

Requirement 3.3,
Oracle Database Vault: Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data. (For the realms component description, please refer to chapter Description of Oracle Database Vault product)

Requirement 3.5,
Requirement 3.5.1,
Oracle Database Vault: For the encryption of data Oracle uses Oracle Wallet. It is encrypted using the wallet password. To be able to open wallet from within the database requires the "alter system" privilege. Oracle Database Vault command rules can be used to implement restriction of who can execute "alter system" command. Additionally time execution restriction and place restriction can be implemented. (For the command rules component description, please refer to chapter Description of Oracle Database Vault product)

5.3 REQUIREMENT 6: SECURE SYSTEMS AND APPLICATIONS

Requirement 6.4.1,
Oracle Database Vault: Oracle Database Vault can help to protect DBA access to production data in Oracle Databases.

5.4 REQUIREMENT 7: ACCESS RESTRICTION

Oracle Database Vault: This is key feature of the Oracle Database Vault. It is meant to support requirement 7. Only requirement part 7.1.3, 7.1.4 and 7.3 are not related to Oracle Database Vault.

Requirement 7.1.1,
Oracle Database Vault:

- Oracle Database Vault has security component called realms. It creates possibility to prevent privileged users from accessing application data.

- Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.
• Oracle Database Vault implements separation of duties concept. This concept helps to prevent unauthorized administrative actions in the Oracle Database.

• Oracle Database Vault Variables can be used for dynamic access right determination.

• Oracle Database Vault Mandatory Realm can protect data by blocking access to data to anyone even for owner and for users with direct grants.

Requirement 7.1.2,
Oracle Database Vault:

• Oracle Database Vault has security component called realms. It creates possibility to prevent privileged users from accessing application data.

• Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.

• Oracle Database Vault implements separation of duties concept. This concept helps to prevent unauthorized administrative actions in the Oracle Database.

• Oracle Database Vault Variables can be used for dynamic access right determination.

• Oracle Database Vault Mandatory Realm can protect data by blocking access to data to anyone even for owner and for users with direct grants.

Requirement 7.2.1,
Oracle Database Vault:
Oracle Database Vault covers only its component area.
• Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.

• Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.

• Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.

• Oracle Database Vault Variables can be used for dynamic access right determination.

• Oracle Database Vault Mandatory Realm can protect data by blocking access to data to anyone even for owner and for users with direct grants.

Requirement 7.2.2,
Oracle Database Vault:

• Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.

• Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.

• Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.

• Oracle Database Vault Variables can be used for dynamic access right determination.

• Oracle Database Vault Mandatory Realm can protect data by blocking access to data to anyone even for owner and for users with direct grants.
Requirement 7.2.3,
Oracle Database Vault: Oracle Database Vault will deny all requests that are not following specific rules.

- Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.
- Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.
- Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.
- Oracle Database Vault Variables can be used for dynamic access right determination.
- Oracle Database Vault Mandatory Realm can protect data by blocking access to data to anyone even for owner and for users with direct grants.

5.5 REQUIREMENT 8: UNIQUE IDENTIFICATION

Requirement 8.5.6,
Oracle Database Vault: Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications. Time period limitations can be also implemented. Oracle Database Vault does not disable accounts, it helps to configure access rules for such accounts.

5.6 REQUIREMENT 10: MONITORING

Requirement 10.1,
Oracle Database Vault: Oracle Database Vault has an audit trails implementation that helps to identify violation occur on protected by Oracle Database Vault realm objects in Database.

Requirement 10.2.2,
Oracle Database Vault:

- Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.
- Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.
- Oracle Database Vault Realm reports.

5.7 REQUIREMENT A.1: SHARED HOSTING PROVIDERS

Requirement A.1,
Requirement A.1.1,
Oracle Database Vault:

- Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.
- Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.
- Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.
• Oracle Database Vault has security component called realms. It create possibility to prevent privileged users from accessing application data.

• Oracle Database Vault command rules and factors can be used to implement restriction on access to data, database and applications.

• Oracle Database Vault implements separation of duties concept. This concept helps to prevents unauthorized administrative actions in the Oracle Database.

5.8 ANALYSIS

Requirement 2 is supported partly by Oracle Database Vault and it is not a most significant part of product compliance. Some parts of the Requirement 2 is not relevant to the product. Oracle Database Vault has mechanisms to ensure secure password creation and separation of duties concept to prevent misuse of administrative actions, but other system’s users password management is out of this product scope.

Requirement 3 is also supported only partly due to the fact that some parts again are not relevant to the product. Oracle Database Vault puts additional control on who, when and from where can access stored data and cryptographic keys to support Requirement 3.

Requirement 6 is mostly not relevant to Oracle Database Vault. Product only supports part that requires separation of environments and it’s protection. Oracle Database Vault can help to protect most important production environment from privileged user’s access.

Requirement 7 is the most supported by Oracle Database Vault. This requirement is the main focus of the product. Only several parts of this requirement are not relevant to Oracle Database Vault. Access restriction is the biggest feature of Oracle Database Vault.
It has a lot of methods and concepts to support access control for privileged users. Oracle Database Vault can help to define who, when and from where can access sensitive data. It can even block access to data to anyone in case of violation.

Requirement 8 is supported only partly by Oracle Database Vault, as most of the parts are not relevant to the product. It supports only time access restriction for data access part of the requirement.

Requirement 10 is supported only partly by Oracle Database Vault, as most of the parts are not relevant to the product. Oracle Database Vault has a focus on monitoring privileged users activity and blocking unauthorized activity.

Requirement A.1 is another requirement that is mostly supported by Oracle Database Vault. This requirement states access separation restrictions and Oracle Database Valt has a lot of features to support such restrictions.

After going through PCI-DSS security standard and checking Oracle Database Vault compliance with this security standard, it became more obvious, that Oracle Database Vault is compliant with PCI-DSS security standard in its sphere of influence. It is meant to support only part of such standard with focus area on privilege users access controls. As we saw earlier, it is Requirement 7 and Appendix 1. Those are the main area of interest for Oracle Database Vault product. Oracle Database Vault has a lot of instruments, concepts, and methods for privileged users access control, monitoring, and reports. All these help Oracle Database Vault to support very important part of any security area - internal security. Additionally, it helps organization to comply with PCI-DSS standard.
In this work the main subject was an Oracle Database Vault, it’s security features and research on compliance check with PCI DSS standard.

First, I introduced the history of database security, its evolution through the years with current trends in database development. Then, I described foundation for the most popular and used database technology - relational databases. I referenced to the most current trends in threats against databases. This part gave a good picture of what database security should address in current market situation.

Next, I listed what kind of general recommendation Oracle has for protecting databases and sensitive data against most common threats. Analysis of threats and protection techniques helped to realize that database has a gap in security layer, which address internal security threats.

Oracle Database Vault product was introduced to fill the gap in internal security. Oracle Database Vault security methods and features were described to understand, how this product protects databases from internal threats and puts control on priviledge users access.

Next, was a PCI DSS description part, that gave a clear picture on purpose of the standard and requirements that organization should meet to comply with PCI DSS standard. Each requirement was describe in enough details to help in future understanding of compliance check research.

Section 5, demonstrates the results of the compliance check research. This section lists all the requirements that Oracle Database Vault supports and provides details on what kind of security
methods and features support each requirement. The results have a significant value for the organizations that are planning their security program and need to comply with PCI DSS standard. Organization should have a good understanding on what level each product supports certain security standard to plan their security stack. The results of this work will help organizations to identify requirements that Oracle Database Vault supports and not least important part of the PCI DSS standard that needs to be covered by other security products.

Consider such fact that Oracle Database Vault is quite unique product on the market and address organization internal threats protection, it is definitely deserve a consideration, but it would be more suitable for a bigger organization, as it requires bigger resources for implementation and support.

Implementation of Oracle Database Vault is another topic and it will not be considered here. However, this is a good topic for future research. Future work can include development of detailed implementation plan of Oracle Database Vault product. It should include plan of resources required to do basic implementation and it could have most recommended implementation. Also, it might include recommended configuration of Oracle Database Vault and step-by-step implementation plan.

Oracle Database Vault has been created as an enhancement to Oracle Database to add separation of duties. The idea behind Oracle Database Vault is to let DBAs perform their everyday duties without being able to read or write data to database tables, but as all existing products Oracle Database Vault has its own vulnerabilities. Research of existing product vulnerabilities and possible solutions for them could be a good topic for another research.
References


[12] VISA, official site, [http://usa.visa.com](http://usa.visa.com)

A Oracle Database Security Checklist

- **INSTALL ONLY WHAT IS REQUIRED:** Oracle recommends to install minimum set of features for production systems and in case if additional features or options needed, Oracle installer can be simply re-ran and features or options can be added. Oracle recommends not to install sample schemas to production systems and if you have them, then lock or remove it.

- **LOCK AND EXPIRE DEFAULT USER ACCOUNTS:** The Oracle database installs with some default user accounts. Database configuration assistant automatically locks and expires most of the users accounts, but in case of manual database installations these accounts are not locked. You need to lock and expire these user accounts. The list of all default accounts and their statuses can be found in original checklist paper on Oracle website.

- **CHANGING DEFAULT USER PASSWORDS:** Creating secure password and performing good password policies are very important part in protecting you database against security threats. The most important password criteria is length and complexity. Oracle recommendations for creating secure and complex password’s are: [7]
  - At least 10 values in length.
  - A mixture of letters and numbers.
  - Contain mixed case (Supported in Oracle Database 11g).
  - Include symbols (Supported in Oracle Database 11g).
  - Little or no relation to an actual word.
– Create passwords from the 1st letters of the words of an easy-to-remember sentence.
– Combine 2 weaker passwords.
– Repeat a character at the beginning or end of the password.
– Add or append a string of some sort.
– Append part of the same password.
– Double some or all of the letters.

These recommendations make it harder for the attacker to crack the password.

• **CHANGE PASSWORDS FOR ADMINISTRATIVE ACCOUNTS:** Oracle recommends to use different and strong passwords for administrative accounts such as SYSTEM, SYSMAN, DBSNMP, whenever it is production or test environment.

• **CHANGE DEFAULT PASSWORDS FOR ALL USERS:** If there are unlocked accounts after installation, they have a default password which is the same as the user account. New strong password should be assign to such accounts.

• **ENFORCE PASSWORD MANAGEMENT:** Oracle recommends enforce failed login, password expiration, password complexity and reuse policies using Oracle profiles, and follow best practices defined by Oracle Applications. Oracle recommends that all user’s password should be changed periodically and basic password management rules applied to all user passwords.

• **SECURE BATCH JOBS:** Oracle Database has a new feature - the Secure External Password Store, starting from database version 10 g release 2. This feature helps to secure batch jobs and tasks performed without user interaction, and which uses credential identification to connect to database. The passwords
and user names are stored in Oracle Wallet and used for running batch jobs and other tasks. This technic removes the need to hard code credentials in batch jobs, but location and permissions to wallet should be carefully considered in favor of least-privileges.

- **MANAGE ACCESS TO SYSDBA AND SYSOPER ROLES:** Oracle recommends to avoid connecting with SYSDBA role, unless it is absolutely necessary in such cases like patching and called for by an existing Oracle feature. Attention should be paid to managing access and monitoring for unsuccessful SYSDBA and SYSOPER connections. Oracle recommends for large and small organizations to create their own separate administrative account, instead of using SYSDBA.

- **ENABLE ORACLE DATA DICTIONARY PROTECTION:** Oracle recommends that data dictionary protection should be implemented to prevent misusing and harming data dictionary by users, who has "ANY" system privileges. Set O7_DICTIONARY_ACCESSIBILITY parameter to FALSE to enable data dictionary protection. In version Oracle 9i and later this parameter default settings is set to FALSE. If some applications need to use data dictionary, they can be granted the SELECT ANY DICTIONARY system privilege individually.

- **FOLLOW THE PRINCIPLE OF LEAST PRIVILEGE:** Oracle recommends to avoid giving high privileges to new database users, even if those users are privileged users. Special attention should be given when assigning privileges to application schemas. SYSDBA role should be granted with high caution and to the people in most trusted position. All users’ activities, who connects with SYSDBA role or other administrative roles should be constantly monitored.

- **PUBLIC PRIVILEGES:** The topic of PUBLIC privileges is part of Oracle’s overall secure-by-default initiative that started with
Oracle Database 9i. New in the Oracle Database 11g release are granular authorizations for numerous PL/SQL network utility packages granted to PUBLIC.

- **RESTRICT PERMISSIONS ON RUN-TIME FACILITIES**: When granting permissions on run-time facilities such as the Oracle Java Virtual Machine (OJVM), grant permissions to the explicit or actual document root file path. This code can be changed to use the explicit file path: 
  
  ```sql
  dbms_java.grant_permission ('SCOTT', 'SYS:java.io.FilePermission', '<<ALL FILES >>', 'read');
  dbms_java.grant_permission ('SCOTT', 'SYS:java.io.FilePermission', '<<actual directory path ><?', 'read');
  ```

- **AUTHENTICATE CLIENTS**: Oracle recommends to make sure that database initialization parameter REMOTE_OS_AUTHENT is set to FALSE. It is a default value for this parameter and should not be changed. This configuration is more secure. It is enforcing server-based authentication of clients connecting to an Oracle database.

- **RESTRICT OPERATING SYSTEM ACCESS**: Oracle recommendations to restrict amount of users, who have operating systems access to Oracle Database host and to limit the ability to modify permissions of default file, Oracle Database installation directory, and its content. Oracle owner and privileged system users should not change these permissions. They can be modified only in case if Oracle give instructions to do so. Only trusted users should be able to own and modify path or file to Oracle database, such as DBA or another trusted operating system account.

- **SECURE THE ORACLE LISTENER**: The Oracle Listener should be properly configured for optimal security. Oracle Database 10g Release 1 and higher uses local OS authentication as the default authentication mode. This mode requires the
Oracle Net administrator to be a member of the local DBA group. Setting a password for the TNS listener in Oracle Database 10g Release 1 and higher simplifies local administration. However, setting a password requires good password management to prevent unauthorized users from guessing the password and potentially gaining access to privileged listener operations. Customers may wish to consider not setting a password for the TNS listener starting with Oracle Database 10g Release 1. Passwords should be used for databases prior to Oracle Database 10g Release 1 or for remote administration of listeners on Oracle Database 10g Release 1 and higher databases. You should also consider using a firewall. Proper use of a firewall will reduce exposure to security related information including port openings and other configuration information located behind the firewall. Oracle Net supports a variety of firewalls.

- **SECURE EXTERNAL PROCEDURES:** The default configuration for external procedures no longer requires a network listener to work with Oracle Database and EXTPROC agent. The EXTPROC agent is spawned directly by Oracle Database and eliminates the risks that extproc might be spawned by Oracle Listener, unexpectedly. This default configuration is recommended for maximum security. Having your EXTPROC agent spawned by Oracle Listener is necessary if you use: [7]
  - Multi-threaded Agent.
  - Oracle Database in MTS mode on Windows.
  - AGENT clause of the LIBRARY specification or AGENT IN clause of the PROCEDURE specification such that you can redirect external procedures to a different EXTPROC agent. Please refer to the Oracle Net Services Guide for instructions on properly configuring Oracle Net Services for external procedures.
• **PREVENT RUNTIME CHANGES TO LISTENER:** Recommended and default value for ADMIN.RESTRICTIONS.LISTENER is ON and runtime changes to the listener parameters are disabled. To be able to do any changes LISTENER.ORA file should be modified and manually reloaded.

• **CHECKING NETWORK IP ADDRESSES:** Oracle recommends to use Oracle Net valid note security feature to control access from specific IP addresses to Oracle server processes. Setting parameters in Oracle Net configuration file protocol.ora allows or denies specific client IP addresses to make connections to Oracle listener: tcp.validnote.checking = YES, tcp.excluded.nodes = list of IP addresses, tcp.invited.nodes = list of IP addresses.

• **HARDEN THE OPERATING SYSTEM:** Make sure that User Datagram Protocol (UDP) and Transmission Control Protocol (TCP) ports are closed for unnecessary services, which UNIX and Windows systems provide. Both type of ports should be closed for each disabled service. Disabling only one type of port does not make operating system more secure.

• **ENCRYPT NETWORK TRAFFIC:** Oracle recommendation is to encrypt network traffic between application servers, databases, and clients. Oracle supports SSL certificates and network encryption without certificates.

• **APPLY ALL SECURITY PATCHES:** Operating systems and Oracle latest patches should be applied. Follow up Oracle Technology Network (OTN) security site for information on latest security alerts provided by Oracle. Also, check Oracle Worldwide Supports services site, Metalink, for more detailed information, upcoming security patches, and specific security configuration information.
• **REPORT SECURITY ISSUES TO ORACLE:** Any vulnerabilities founded in Oracle Database can be reported as a request to Oracle Worldwide Support Services using Metalink or email: secalert_us@oracle.com. Description of the problem, product version, platform, scripts, and examples are necessary to provide in a request. [7]
B Key Features and Benefits of Oracle Database Vault

- Preventive controls to block privileged users and DBAs from accessing sensitive data in the databases.
- Enforce operational controls inside the database, lock down the configuration, and prevent audit findings.
- Seal off highly sensitive application objects during maintenance periods and in response to cyber threats.
- Increase security controls for consolidation, cloud environments, and outsourcing, and help to comply with regulations from multiple countries.
- Identify used and unused privileges and roles with Privilege Analysis.
- Application-specific protection policies for enterprise applications including Fusion Applications, E-Business Suite, PeopleSoft, Siebel, and SAP.
Oracle Database Vault provides essential safeguards against common threats

- Threats that exploit stolen credentials obtained from social engineering, key-loggers, and other mechanisms to get access to privileged accounts in your database.

- Threats from insiders that misuse privileged accounts to access sensitive data, or to create new accounts, and grant additional roles and privileges for future exploits.

- Threats from insiders who bypass the organization’s usage policies (including IP address, date, and time of usage), or from unintended mistakes from junior DBAs who might use unauthorized SQL commands that change the database configuration and put the database in a vulnerable state.

- Threats to sensitive data during maintenance window from the application administrators.

- Threats that exploit weaknesses in the application to escalate privileges and attack other applications on the same database.
The PCI-DSS Requirements

Requirement 1: Install and maintain firewall configuration to protect card-holder data.

The first requirement describes different requirements of network security that organizations need to comply with PCI-DSS. [14]

Requirement 1.1, Establish and implement firewall and router configuration standards formalize testing whenever configurations change, identifies all connections between the cardholder data environment and other networks (including wireless) with documentation and diagrams, documents business justification and various technical settings for each implementation, that diagram all cardholder data flows across systems and networks, and stipulate a review of configuration rule sets at least every six months: [14]

- 1.1.1 A formal process for approving and testing all network connections and changes to the firewall and router configurations
- 1.1.2 Current network diagram with all connections to cardholder data, including any wireless networks
- 1.1.3 Requirements for a firewall at each Internet connection and between any demilitarized zone (DMZ) and the internal network zone
- 1.1.4 Description of groups, roles, and responsibilities for logical management of network components
- 1.1.5 Documentation and business justification for use of all services, protocols, and ports allowed, including documentation of security features implemented for those protocols considered to be insecure
• 1.1.6 Requirement to review firewall and router rule sets at least every six months

Requirement 1.2, Build firewall and router configurations that restrict all traffic, inbound and outbound, from untrusted networks (including wireless) and hosts, and specifically deny all other traffic except for protocols necessary for the cardholder data environment: [14]

• 1.2.1 Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment.

• 1.2.2 Secure and synchronize router configuration files.

• 1.2.3 Install perimeter firewalls between any wireless networks and the cardholder data environment, and configure these firewalls to deny or control (if such traffic is necessary for business purposes) any traffic from the wireless environment into the cardholder data environment.

Requirement 1.3, Prohibit direct public access between the Internet and any system component in the cardholder data environment: [14]

• 1.3.1 Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

• 1.3.2 Limit inbound Internet traffic to IP addresses within the DMZ.

• 1.3.3 Do not allow any direct connections inbound or outbound for traffic between the Internet and the cardholder data environment.

• 1.3.4 Do not allow internal addresses to pass from the Internet into the DMZ.
• 1.3.5 Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

• 1.3.6 Implement stateful inspection, also known as dynamic packet filtering. (That is, only established connections are allowed into the network.)

• 1.3.7 Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.

• 1.3.8 Do not disclose private IP addresses and routing information to unauthorized parties. Note: Methods to obscure IP addressing may include, but are not limited to:
  
  – Network Address Translation (NAT)
  
  – Placing servers containing cardholder data behind proxy servers/firewalls or content caches,
  
  – Removal or filtering of route advertisements for private networks that employ registered addressing,
  
  – Internal use of RFC1918 address space instead of registered addresses.

Requirement 1.4, Install personal firewall software on any mobile and/or employee-owned devices that connect to the Internet when outside the network, and which are also used to access the network. [14]

Requirement 1.5, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 2 is the following: Do not use vendor-supplied defaults for system passwords and other parameters. [14]

Requirement 2.1, Always change ALL vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network. This includes wireless devices
that are connected to the cardholder data environment or are used to transmit cardholder data: [14]

- 2.1.1 For wireless environments connected to the cardholder data environment or transmitting cardholder data, change wireless vendor defaults, including but not limited to default wireless encryption keys, passwords, and SNMP community strings.

Requirement 2.2, Develop configuration standards for all system components that address all known security vulnerabilities and are consistent with industry-accepted definitions. Update system configuration standards as new vulnerability issues are identified: [14]

- 2.2.1 Implement only one primary function per server to prevent functions that require different security levels from co-existing on the same server. (For example, web servers, database servers, and DNS should be implemented on separate servers.) Note: Where virtualization technologies are in use, implement only one primary function per virtual system component.

- 2.2.2 Enable only necessary and secure services, protocols, daemons, etc. as required for the function of the system. Implement security features for any required services, protocols or daemons that are considered to be insecure.

- 2.2.3 Configure system security parameters to prevent misuse

- 2.2.4 Remove all unnecessary functionality, such as scripts, drivers, features, subsystems, file systems, and unnecessary web servers.

Requirement 2.3, Using strong cryptography, encrypt all non-console administrative access such as browser/web-based management tools. [14]
Requirement 2.4, Maintain an inventory of system components that are in scope for PCI DSS. [14]

Requirement 2.5, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 2.6, Shared hosting providers must protect each entity’s hosted environment and cardholder data (details are in PCI DSS Appendix A: Additional PCI DSS Requirements for Shared Hosting Providers.) [14]

Requirement 3 is the following: Protect stored cardholder data. [14]

Requirement 3.1, Limit cardholder data storage and retention time to that which is required for business, legal, and/or regulatory purposes, as documented in your data retention policy. Purge unnecessary stored data at least quarterly: [14]

- 3.1.1 Implement a data retention and disposal policy that includes: Limiting data storage amount and retention time to that which is required for legal, regulatory, and business requirements.
  - Processes for secure deletion of data when no longer needed.
  - Specific retention requirements for cardholder data.
  - A quarterly automatic or manual process for identifying and securely deleting stored cardholder data that exceeds defined retention requirements.

Requirement 3.2, Do not store sensitive authentication data after authorization (even if it is encrypted). See table below. Render all sensitive authentication data unrecoverable upon completion of the authorization process. Issuers and related entities may store sensitive authentication data if there is a business justification, and the data is stored securely: [14]

- 3.2.1 Do not store the full contents of any track from the magnetic stripe (located on the back of a card, equivalent data
contained in a chip, or elsewhere). This data is alternatively called full track, track, track 1, track 2, and magnetic-stripe data.

- 3.2.2 Do not store the card-verification code or value (three-digit or four-digit number printed on the front or back of a payment card) used to verify card-not-present transactions.

- 3.2.3 Do not store the personal identification number (PIN) or the encrypted PIN block.

Requirement 3.3, Mask PAN when displayed (the first six and last four digits are the maximum number of digits you may display), so that only authorized people with a legitimate business need can see the full PAN. This does not supersede stricter requirements that may be in place for displays of cardholder data, such as on a point-of-sale receipt. [14]

Requirement 3.4, Render PAN unreadable anywhere it is stored including on portable digital media, backup media, in logs, and data received from or stored by wireless networks. Technology solutions for this requirement may include strong one-way hash functions of the entire PAN, truncation, index tokens with securely stored pads, or strong cryptography. (See PCI DSS Glossary for definition of strong cryptography [14].) Note: It is a relatively trivial effort for a malicious individual to reconstruct original PAN data if they have access to both the truncated and hashed version of a PAN. Where hashed and truncated versions of the same PAN are present in an entity’s environment, additional controls should be in place to ensure that the hashed and truncated versions cannot be correlated to reconstruct the original PAN. [14]

- 3.4.1 If disk encryption is used (rather than file- or column-level database encryption), logical access must be managed independently of native operating system access control mechanisms (for example, by not using local user...
account databases). Decryption keys must not be tied to user accounts.

Requirement 3.5, Document and implement procedures to protect any keys used for encryption of cardholder data from disclosure and misuse: [14]

- 3.5.1 Restrict access to cryptographic keys to the fewest number of custodians necessary
- 3.5.2 Store cryptographic keys securely in the fewest possible locations and forms

Requirement 3.6, Fully document and implement key management processes and procedures for cryptographic keys used for encryption of cardholder data: [14]

- 3.6.1 Generation of strong cryptographic keys
- 3.6.2 Secure cryptographic key distribution
- 3.6.3 Secure cryptographic key storage
- 3.6.4 Cryptographic key changes for keys that have reached the end of their cryptoperiod (for example, after a defined period of time has passed and/or after a certain amount of cipher-text has been produced by a given key), as defined by the associated application vendor or key owner, and based on industry best practices and guidelines (for example, NIST Special Publication 800-57).
- 3.6.5 Retirement or replacement (for example, archiving, destruction, and/or revocation) of keys as deemed necessary when the integrity of the key has been weakened (for example, departure of an employee with knowledge of a clear-text key), or keys are suspected of being compromised. Note: If retired or replaced cryptographic keys need to be retained, these keys must be securely archived (for example, by using a key
encryption key). Archived cryptographic keys should only be used for decryption/verification purposes.

- **3.6.6** If manual clear-text cryptographic key management operations are used, these operations must be managed using split knowledge and dual control (for example, requiring two or three people, each knowing only their own key component, to reconstruct the whole key). Note: Examples of manual key management operations include, but are not limited to: key generation, transmission, loading, storage and destruction.

- **3.6.7** Prevention of unauthorized substitution of cryptographic keys

- **3.6.8** Requirement for cryptographic key custodians to formally acknowledge that they understand and accept their key-custodian responsibilities

Requirement 3.7, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 4 is the following: *Encrypt transmission of cardholder data across open, public networks.* [14]

Requirement 4.1, Use strong cryptography and security protocols such as SSL/TLS, SSH or IPSec to safeguard sensitive cardholder data during transmission over open, public networks (e.g. Internet, wireless technologies, cellular technologies, General Packet Radio Service (GPRS), and satellite communications). Ensure wireless networks transmitting cardholder data or connected to the cardholder data environment use industry best practices (e.g., IEEE 802.11i) to implement strong encryption for authentication and transmission. The use of WEP as a security control is prohibited: [14]

- **4.1.1** Ensure wireless networks transmitting cardholder data or connected to the cardholder data environment, use industry
best practices (e.g., IEEE 802.11i) to implement strong encryption for authentication and transmission. Note: The use of WEP as a security control was prohibited as of 30 June, 2010.

Requirement 4.2, Never send unprotected PANs by end user messaging technologies (for example, e-mail, instant messaging, chat, etc.). [14]

Requirement 4.3, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 5 is the following: Use and regularly update antivirus software or programs. [14]

Requirement 5.1, Deploy anti-virus software on all systems commonly affected by malicious software (particularly personal computers and servers). For systems not affected commonly by malicious software, perform periodic evaluations to evaluate evolving malware threats and confirm whether such systems continue to not require anti-virus software: [14]

- 5.1.1 Ensure that all anti-virus programs are capable of detecting, removing, and protecting against all known types of malicious software.

Requirement 5.2, Ensure that all anti-virus mechanisms are kept current, perform periodic scans, generate audit logs, which are retained per PCI DSS Requirement 10.7. [14]

Requirement 5.3, Ensure that anti-virus mechanisms are actively running and cannot be disabled or altered by users, unless specifically authorized by management on a case-by-case basis for a limited time period. [14]

Requirement 5.4, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 6 is the following: Develop and maintain secure systems and applications. [14]
Requirement 6.1, Establish a process to identify security vulnerabilities, using reputable outside sources, and assign a risk ranking (e.g. high, medium, or low) to newly discovered security vulnerabilities. [14]

Requirement 6.2, Protect all system components and software from known vulnerabilities by installing applicable vendor-supplied security patches. Install critical security patches within one month of release. [14]

Requirement 6.3, Develop internal and external software applications including web-based administrative access to applications in accordance with PCI DSS and based on industry best practices. Incorporate information security throughout the software development life cycle. This applies to all software developed internally as well as bespoke or custom software developed by a third party: [14]

- 6.3.1 Removal of custom application accounts, user IDs, and passwords before applications become active or are released to customers.
- 6.3.2 Review of custom code prior to release to production or customers in order to identify any potential coding vulnerability. Note: This requirement for code reviews applies to all custom code (both internal and public-facing), as part of the system development lifecycle. Code reviews can be conducted by knowledgeable internal personnel or third parties. Web applications are also subject to additional controls, if they are public facing, to address ongoing threats and vulnerabilities after implementation, as defined at PCI DSS Requirement 6.6.

Requirement 6.4, Follow change control processes and procedures for all changes to system components: [14]

- 6.4.1 Separate development/test and production environments
- 6.4.2 Separation of duties between development/test and production environments
• 6.4.3 Production data (live PANs) are not used for testing or development

• 6.4.4 Removal of test data and accounts before production systems become active

• 6.4.5 Change control procedures for the implementation of security patches and software modifications. Procedures must include the following: [14]
  
  – 6.4.5.1 Documentation of impact.
  – 6.4.5.2 Documented change approval by authorized parties.
  – 6.4.5.3 Functionality testing to verify that the change does not adversely impact the security of the system.
  – 6.4.5.4 Back-out procedures.

Requirement 6.5, Prevent common coding vulnerabilities in software development processes by training developers in secure coding techniques and developing applications based on secure coding guidelines including how sensitive data is handled in memory: [14]

• 6.5.1 Injection flaws, particularly SQL injection. Also consider OS Command Injection, LDAP and XpPath injection flaws as well as other injection flaws.

• 6.5.2 Buffer overflow.

• 6.5.3 Insecure cryptographic storage.

• 6.5.4 Insecure communications.

• 6.5.5 Improper error handling.

• 6.5.6 All High vulnerabilities identified in the vulnerability identification process (as defined in PCI DSS Requirement 6.2).
Note: This requirement is considered a best practice until June 30, 2012, after which it becomes a requirement.

Requirements 6.5.7 through 6.5.9, below, apply to web applications and application interfaces (internal or external):

- 6.5.7 Cross-site scripting (XSS).
- 6.5.8 Improper Access Control (such as insecure direct object references, failure to restrict URL access, and directory traversal).
- 6.5.9 Cross-site request forgery (CRSF).

Requirement 6.6, Ensure all public-facing web applications are protected against known attacks, either by performing application vulnerability assessment at least annually and after any changes, or by installing an automated technical solution that detects and prevents web-based attacks (for example, a web-application firewall) in front of public-facing web applications, to continually check all traffic. [14]

Requirement 6.7, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 7 is the following: *Restrict cardholder information by business need to know.* [14]

Requirement 7.1, Limit access to system components and cardholder data to only those individuals whose job requires such access: [14]

- 7.1.1 Restriction of access rights to privileged user IDs to least privileges necessary to perform job responsibilities
- 7.1.2 Assignment of privileges is based on individual personnel’s job classification and function
- 7.1.3 Requirement for a documented approval by authorized parties specifying required privileges
7.1.4 Implementation of an automated access control system

Requirement 7.2, Establish an access control system for systems components that restricts access based on a user’s need to know, and is set to deny all unless specifically allowed: [14]

- 7.2.1 Coverage of all system components
- 7.2.2 Assignment of privileges to individuals based on job classification and function
- 7.2.3 Default deny-all setting

Requirement 7.3, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 8 is the following: Assign a unique ID to each person with computer access. [14]

Requirement 8.1, Define and implement policies and procedures to ensure proper user identification management for users and administrators on all system components. Assign all users a unique user name before allowing them to access system components or cardholder data. [14]

Requirement 8.2, Employ at least one of these to authenticate all users: something you know, such as a password or passphrase; something you have, such as a token device or smart card; or something you are, such as a biometric. Use strong authentication methods and render all passwords unreadable during transmission and storage using strong cryptography. [14]

Requirement 8.3, Implement two-factor authentication for all remote network access that originates from outside the network, by employees, administrators, and third parties including vendor access for support or maintenance. Examples of two-factor technologies include remote authentication and dial-in service (RADIUS) with tokens; terminal access controller access control system (TACACS) with tokens; or other technologies that facilitate
two-factor authentication. Using one factor twice (e.g. using two separate passwords) is not considered two-factor authentication. [14]

Requirement 8.4, Develop, implement, and communicate authentication procedures and policies to all users. [14]

Requirement 8.5, Do not use group, shared, or generic IDs, or other authentication methods. Service providers with access to customer environments must use a unique authentication credential (such as a password/passphrase) for each customer environment: [14]

- 8.5.1 Control addition, deletion, and modification of user IDs, credentials, and other identifier objects
- 8.5.2 Verify user identity before performing password resets.
- 8.5.3 Set passwords for first-time use and resets to a unique value for each user and change immediately after the first use.
- 8.5.4 Immediately revoke access for any terminated users.
- 8.5.5 Remove/disable inactive user accounts at least every 90 days.
- 8.5.6 Enable accounts used by vendors for remote access only during the time period needed. Monitor vendor remote access accounts when in use.
- 8.5.7 Communicate authentication procedures and policies to all users who have access to cardholder data.
- 8.5.8 Do not use group, shared, or generic accounts and passwords, or other authentication methods.
- 8.5.9 Change user passwords at least every 90 days.
- 8.5.10 Require a minimum password length of at least seven characters.
8.5.11 Use passwords containing both numeric and alphabetic characters.

8.5.12 Do not allow an individual to submit a new password that is the same as any of the last four passwords he or she has used.

8.5.13 Limit repeated access attempts by locking out the user ID after not more than six attempts.

8.5.14 Set the lockout duration to a minimum of 30 minutes or until administrator enables the user ID.

8.5.15 If a session has been idle for more than 15 minutes, require the user to re-authenticate to re-activate the terminal or session.

8.5.16 Authenticate all access to any database containing cardholder data. This includes access by applications, administrators, and all other users. Restrict user direct access or queries to databases to database administrators.

Requirement 8.6, Use of other authentication mechanisms such as physical security tokens, smart cards, and certificates must be assigned to an individual account. [14]

Requirement 8.7, All access to any database containing cardholder data must be restricted: all user access must be through programmatic methods; only database administrators can have direct or query access; and application IDs for database applications can only be used by the applications (and not by users or non-application processes). [14]

Requirement 8.8, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 9 is the following: Restrict physical access to cardholder data. [14]
Requirement 9.1, Use appropriate facility entry controls to limit and monitor physical access to systems in the cardholder data environment: [14]

- 9.1.1 Use video cameras and/or access control mechanisms to monitor individual physical access to sensitive areas. Review collected data and correlate with other entries. Store for at least three months, unless otherwise restricted by law.

- 9.1.2 Restrict physical access to publicly accessible network jacks. For example, areas accessible to visitors should not have network ports enabled unless network access is specifically authorized.

- 9.1.3 Restrict physical access to wireless access points, gateways, handheld devices, networking/communications hardware, and telecommunications lines.

Requirement 9.2, Develop procedures to easily distinguish between onsite personnel and visitors, such as assigning ID badges. [14]

Requirement 9.3, Control physical access for onsite personnel to the sensitive areas. Access must be authorized and based on individual job function; access must be revoked immediately upon termination, and all physical access mechanisms, such as keys, access cards, etc. returned or disabled: [14]

- 9.3.1 Authorized before entering areas where cardholder data is processed or maintained

- 9.3.2 Given a physical token (for example, a badge or access device) that expires and that identifies the visitors as not onsite personnel

- 9.3.3 Asked to surrender the physical token before leaving the facility or at the date of expiration

69
Requirement 9.4, Ensure all visitors are authorized before entering areas where cardholder data is processed or maintained; given a physical token that expires and that identifies visitors as not onsite personnel; and are asked to surrender the physical token before leaving the facility or at the date of expiration. Use a visitor log to maintain a physical audit trail of visitor information and activity, including visitor name, company, and the onsite personnel authorizing physical access. Retain the log for at least three months unless otherwise restricted by law. [14]

Requirement 9.5, Physically secure all media; store media back-ups in a secure location, preferably off site. [14]

Requirement 9.6, Maintain strict control over the internal or external distribution of any kind of media. [14]

Requirement 9.7, Maintain strict control over the storage and accessibility of media: [14]

- 9.7.1 Classify the media so the sensitivity of the data can be determined.

- 9.7.2 Send the media by secured courier or other delivery method that can be accurately tracked.

Requirement 9.8, Destroy media when it is no longer needed for business or legal reasons. [14]

Requirement 9.9, Protect devices that capture payment card data via direct physical interaction with the card from tampering and substitution. This includes periodic inspections of POS device surfaces to detect tampering, and training personnel to be aware of suspicious activity: [14]

- 9.9.1 Properly maintain inventory logs of all media and conduct media inventories at least annually.

Requirement 9.10, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties: [14]
• 9.10.1 Shred, incinerate, or pulp hardcopy materials so that cardholder data cannot be reconstructed.

• 9.10.2 Render cardholder data on electronic media unrecoverable so that cardholder data cannot be reconstructed.

Requirement 10 is the following: *Track and monitor all access to network resources and cardholder data.* [14]

Requirement 10.1, Implement audit trails to link all access to system components to each individual user. [14]

Requirement 10.2, Implement automated audit trails for all system components for reconstructing these events: all individual user accesses to cardholder data; all actions taken by any individual with root or administrative privileges; access to all audit trails; invalid logical access attempts; use of and changes to identification and authentication mechanisms (including creation of new accounts, elevation of privileges), and all changes, additions, deletions to accounts with root or administrative privileges; initialization, stopping or pausing of the audit logs; creation and deletion of system-level objects: [14]

• 10.2.1 All individual accesses to cardholder data

• 10.2.2 All actions taken by any individual with root or administrative privileges

• 10.2.3 Access to all audit trails

• 10.2.4 Invalid logical access attempts

• 10.2.5 Use of identification and authentication mechanisms

• 10.2.6 Initialization of the audit logs

• 10.2.7 Creation and deletion of system-level objects

Requirement 10.3, Record audit trail entries for all system components for each event, including at a minimum: user identification, type of event, date and time, success or failure indication,
origination of event, and identity or name of affected data, system component or resource: [14]

- 10.3.1 User identification
- 10.3.2 Type of event
- 10.3.3 Date and time
- 10.3.4 Success or failure indication
- 10.3.5 Origination of event
- 10.3.6 Identity or name of affected data, system component, or resource

Requirement 10.4, Using time synchronization technology, synchronize all critical system clocks and times and implement controls for acquiring, distributing, and storing time: [14]

- 10.4.1 Critical systems have the correct and consistent time.
- 10.4.2 Time data is protected.
- 10.4.3 Time settings are received from industry-accepted time sources

Requirement 10.5, Secure audit trails so they cannot be altered: [14]

- 10.5.1 Limit viewing of audit trails to those with a job-related need.
- 10.5.2 Protect audit trail files from unauthorized modifications.
- 10.5.3 Promptly back up audit trail files to a centralized log server or media that is difficult to alter.
- 10.5.4 Write logs for external-facing technologies onto a log server on the internal LAN.
• 10.5.5 Use file integrity monitoring or change detection software on logs to ensure that existing log data cannot be changed without generating alerts (although new data being added should not cause an alert).

Requirement 10.6, Review logs and security events for all system components to identify anomalies or suspicious activity. Perform critical log reviews at least daily. [14]

Requirement 10.7, Retain audit trail history for at least one year; at least three months of history must be immediately available for analysis. [14]

Requirement 10.8, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 11 is the following: Regularly test security systems and processes. [14]

Requirement 11.1, Implement processes to test for the presence of wireless access points (802.11) and detect and identify all authorized and unauthorized wireless access points on a quarterly basis. Maintain an inventory of authorized wireless access points and implement incident response procedures in the event unauthorized wireless access points are detected. [14]

Requirement 11.2, Run internal and external network vulnerability scans at least quarterly and after any significant change in the network. Perform rescans as needed, until passing scans are achieved. After passing a scan for initial PCI DSS compliance, an entity must, in subsequent years, pass four consecutive quarterly scans as a requirement for compliance. Quarterly external scans must be performed by an Approved Scanning Vendor (ASV). Scans conducted after network changes and internal scans may be performed by internal staff: [14]

• 11.2.1 Perform quarterly internal vulnerability scans.
• 11.2.2 Perform quarterly external vulnerability scans via an Approved Scanning Vendor (ASV) approved by the Payment
Card Industry Security Standards Council (PCI SSC).

- 11.2.3 Perform internal and external scans after any significant change.

Requirement 11.3, Develop and implement a methodology for penetration testing that includes external and internal penetration testing at least annually and after any upgrade or modification. If segmentation is used to reduce PCI DSS scope, perform penetration tests to verify the segmentation methods are operational and effective: [14]

- 11.3.1 Network-layer penetration tests
- 11.3.2 Application-layer penetration tests

Requirement 11.4, Use network intrusion detection and/or intrusion prevention techniques to detect and/or prevent intrusions into the network. Monitor all traffic at the perimeter of the cardholder data environment as well as at critical points inside of the cardholder data environment, and alert personnel to suspected compromises. IDS/IPS engines, baselines, and signatures must be kept up to date. [14]

Requirement 11.5, Deploy a change detection mechanism (for example, file integrity monitoring tools) to alert personnel to unauthorized modification of critical system files, configuration files or content files. Configure the software to perform critical file comparisons at least weekly. Implement a process to respond to any alerts generated by the change-detection solution. [14]

Requirement 11.6, Ensure that related security policies and operational procedures are documented, in use, and known to all affected parties. [14]

Requirement 12 is the following: Maintain a policy that addresses information security for all personnel. [14]

Requirement 12.1, Establish, publish, maintain, and disseminate a security policy; review the security policy at least annually and update when the environment changes: [14]
• 12.1.1 Addresses all PCI DSS requirements

• 12.1.2 Includes an annual process that identifies threats, and vulnerabilities, and results in a formal risk assessment

• 12.1.3 Includes a review at least annually and updates when the environment changes

Requirement 12.2, Implement a risk assessment process that is performed at least annually and upon significant changes to the environment that identifies critical assets, threats, and vulnerabilities, and results in a formal assessment. [14]

Requirement 12.3, Develop usage policies for critical technologies to define their proper use by all personnel. These include remote access, wireless, removable electronic media, laptops, tablets, handheld devices, email and Internet: [14]

• 12.3.1 Explicit approval by authorized parties

• 12.3.2 Authentication for use of the technology

• 12.3.3 A list of all such devices and personnel with access

• 12.3.4 Labeling of devices to determine owner, contact information, and purpose

• 12.3.5 Acceptable uses of the technology

• 12.3.6 Acceptable network locations for the technologies

• 12.3.7 List of company-approved products

• 12.3.8 Automatic disconnect of sessions for remote access technologies after a specific period of inactivity

• 12.3.9 Activation of remote access technologies for vendors and business partners only when needed by vendors and business partners, with immediate deactivation after use
• 12.3.10 For personnel accessing cardholder data via remote access technologies, prohibit copy, move, and storage of cardholder data onto local hard drives and removable electronic media, unless specifically authorized for a defined business need.

Requirement 12.4, Ensure that the security policy and procedures clearly define information security responsibilities for all personnel. [14]

Requirement 12.5, Assign to an individual or team information security responsibilities defined by 12.5 subsections: [14]

• 12.5.1 Establish, document, and distribute security policies and procedures.

• 12.5.2 Monitor and analyze security alerts and information, and distribute to appropriate personnel.

• 12.5.3 Establish, document, and distribute security incident response and escalation procedures to ensure timely and effective handling of all situations.

• 12.5.4 Administer user accounts, including additions, deletions, and modifications

• 12.5.5 Monitor and control all access to data.

Requirement 12.6, Implement a formal security awareness program to make all personnel aware of the importance of cardholder data security: [14]

• 12.6.1 Educate personnel upon hire at least annually. Note: Methods can vary depending on the role of the personnel and their level of access to the cardholder data.

• 12.6.2 Require personnel to acknowledge at least annually that they have read and understood the security policy and procedures.
Requirement 12.7, Screen potential personnel prior to hire to minimize the risk of attacks from internal sources. Example screening includes previous employment history, criminal record, credit history, and reference checks. [14]

Requirement 12.8, Maintain and implement policies and procedures to manage service providers with which cardholder data is shared, or that could affect the security of cardholder data: [14]

- 12.8.1 Maintain a list of service providers.
- 12.8.2 Maintain a written agreement that includes an acknowledgement that the service providers are responsible for the security of cardholder data the service providers possess.
- 12.8.3 Ensure there is an established process for engaging service providers including proper due diligence prior to engagement.
- 12.8.4 Maintain a program to monitor service providers’ PCI DSS compliance status at least annually.

Requirement 12.9, Additional requirement for service providers only: Service providers acknowledge in writing to customers that they are responsible for the security of cardholder data that they possess or otherwise store, process, or transmit on behalf of the customer, or to the extent they could impact the security of the customer’s cardholder data environment: [14]

- 12.9.1 Create the incident response plan to be implemented in the event of system reach. Ensure the plan addresses the following, at a minimum:
  - Roles, responsibilities and communication and contact strategies in the event of a compromise including notification of the payment brands, at a minimum
  - Specific incident response procedures
  - Business recovery and continuity procedures
– Data backup processes
– Analysis of legal requirements for reporting compromises
– Coverage and responses of all critical system components
– Reference or inclusion of incident response procedures from the payment brands

• 12.9.2 Test the plan at least annually.
• 12.9.3 Designate specific personnel to be available on a 24/7 basis to respond to alerts.
• 12.9.4 Provide appropriate training to staff with security breach response responsibilities.
• 12.9.5 Include alerts from intrusion detection, intrusion prevention, and file integrity monitoring systems.
• 12.9.6 Develop a process to modify and evolve the incident response plan according to lessons learned and to incorporate industry developments.

Requirement 12.10, Implement an incident response plan. Be prepared to respond immediately to a system breach. [14]

The last requirement is the following: Shared hosting providers must protect the cardholder data environment

Requirement A.1, Protect each entity’s (that is merchant, service provider, or other entity) hosted environment and data, per A.1.1 through A.1.4. A hosting provider must fulfill these requirements as well as all other relevant sections of the PCI DSS: [14]

• A.1.1 Ensure that each entity only runs processes that have access to that entity’s cardholder data environment.
• A.1.2 Restrict each entity’s access and privileges to its own cardholder data environment only.
• A.1.3 Ensure logging and audit trails are enabled and unique to each entity’s cardholder data environment and consistent with PCI DSS Requirement 10.

• A.1.4 Enable processes to provide for timely forensic investigation in the event of a compromise to any hosted merchant or service provider [14].