

An Oracle White Paper
January 2011

Sustainable Compliance for the Payment Card Industry Data Security Standard

Introduction.....	2
Oracle Products and PCI Solution Map.....	3
The Challenges of PCI Data Protection	21
Cardholder Data at Risk.....	21
Encryption Comes of Age	21
Compliance Pushes Adoption	21
The Oracle Encryption Solution	22
Securing Data in Motion.....	22
Securing Backup Data	23
Using Masked Data.....	23
Solving Key Management.....	24
Encryption’s Achilles Heel.....	24
PCI Requirements for Key Management.....	24
Oracle Key Management	24
Building Protection Around Cardholder Data.....	25
Strong Authentication.....	25
Monitoring, Tracking and Auditing.....	25
Maintaining Secure Configurations	26
Managing Privileged Accounts.....	26
The Enemy Within	26
PCI Recognizes the Risk	27
Oracle Manages Privileged Accounts	28
Identity Management Enables Compliance, Empowers Business	29
Secure and Flexible Identity and Access Management.....	29
PCI Requirements for Identity and Access Management	29
Identity Management Challenges.....	30
Oracle’s Identity Management Solution.....	32
Conclusion.....	33

Introduction

Many organizations continue to struggle to achieve compliance with the Payment Card Industry Data Security Standard (PCI DSS), the credit card industry mandate to protect cardholder data and prevent fraud. It has been more than five years since the PCI Security Standards Council (PCI SSC) issued the first version of the standard, formulated by five major credit card companies to reconcile their individual programs into a single set of requirements. The council has since issued multiple updates, the most recent in October 2010.

Although the standard is more prescriptive than most government or industry security directives, with its 12 requirements and supporting guidance, compliance is often far more resource-intensive, expensive and error prone than it should be. On average, Tier 1 companies spend more than \$200,000 per year and Tier 2 companies \$100,000 on the annual Qualified Security Assessor (QSA) audits *alone*, according to an April 2010 Ponemon Institute survey of QSAs. This is the tip of the iceberg given the effort to manually collect and analyze log data, compile reports, and maintain and validate security controls and policies through inefficient labor-intensive processes.

The net result is, at best, a compliance “snapshot” that shows technical compliance at audit time, rather than an ongoing, continuous program that integrates security policies and controls into business operations. Access control, data protection and configuration management policies are difficult to implement and harder to maintain, manage and enforce in practice. Compliance and security procedures are too slow in response to business needs and inflexible in today’s dynamic business environments. Manual and/or poorly implemented and managed controls are costly and error-prone and actually impede business without significantly improving security. As a result, individuals and even departments may circumvent policy to “just get it done”. Not surprisingly, major data breaches have occurred in companies that were technically PCI compliant.

Companies can achieve an efficient repeatable and sustainable security program that satisfies both the technical requirements of their PCI obligations and achieves the level of cardholder data protection for which the standard was created. This white paper explains the essentials of a PCI compliance program, focusing on the critical but problematic areas that comprise much of the heart of the requirements:

- Protect cardholder data from unauthorized use
- Enforce strong controls around privileged users and data access
- Implement centralized, automated role-based access control, authorization and authentication

Oracle’s comprehensive portfolio of data security, identity management and configuration management products provide end-to-end security that fulfill the enterprise’s most critical PCI compliance needs.

Oracle Products and PCI Solution Map

Below is a summary table that shows how Oracle’s portfolio of data security, identity management, and configuration management solutions maps onto specific sections of the PCI DSS standard, helping to address customers’ compliance needs.

CHAPTER	PCI 2.0 REQUIREMENT	MATCHING ORACLE FEATURE
BUILD AND MAINTAIN A SECURE NETWORK		
2:	DO NOT USE VENDOR-SUPPLIED DEFAULTS FOR SYSTEM PASSWORDS AND OTHER SECURITY PARAMETERS	
	Malicious individuals (external and internal to a company) often use vendor default passwords and other vendor default settings to compromise systems. These passwords and settings are well known by hacker communities and easily determined via public information.	
2.1:	Always change vendor-supplied defaults before installing a system on the network, including but not limited to passwords, simple network management protocol (SNMP) community strings, and elimination of unnecessary accounts	Oracle locks and expires default accounts and passwords during installation. Passwords for administration accounts are prompted for during installation.

CHAPTER	PCI 2.0 REQUIREMENT	MATCHING ORACLE FEATURE
	2.2:	Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards
	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
	2.3:	Encrypt all non-console administrative access. Use technologies such as SSH, VPN, or SSL/TLS for web-based management and other non-console administrative access
	2.4:	Shared hosting providers must protect each entity's hosted environment and cardholder data. These providers must meet specific requirements as detailed in <i>Appendix A: "Additional PCI DSS Requirements for Shared Hosting Providers"</i>

PROTECT CARDHOLDER DATA		
3:	PROTECT STORED CARDHOLDER DATA	
	<p>Protection methods such as encryption, truncation, masking, and hashing are critical components of cardholder data protection. If an intruder circumvents other network security controls and gains access to encrypted data, without the proper cryptographic keys, the data is unreadable and unusable to that person. Other effective methods of protecting stored data should be considered as potential risk mitigation opportunities. For example, methods for minimizing risk include not storing cardholder data unless absolutely necessary, truncating cardholder data if full PAN is not needed, and not sending unprotected PANs using end-user messaging technologies, such as e-mail and instant messaging.</p> <p>Please refer to the <i>PCI DSS Glossary of Terms, Abbreviations, and Acronyms</i> for definitions of "strong cryptography" and other PCI DSS terms.</p>	
3.3:	<p>Mask PAN when displayed (the first six and last four digits are the maximum number of digits to be displayed).</p> <p><i>Notes:</i></p> <ul style="list-style-type: none"> • This requirement does not apply to employees and other parties with a legitimate business need to see the full PAN • This requirement does not supersede stricter requirements in place for displays of cardholder data—for example, for point-of-sale (POS) receipts 	<p>Application specific; Applications can leverage Virtual Private Database (VPD) with a column relevant policy to mask out the entire number.</p> <p>Security controls provided by Oracle Label Security can help determine who should have access to the number.</p> <p>Oracle Database Vault realms can be used to prevent privileged users from accessing application data.</p>

<p>3.4:</p>	<p>Render PAN unreadable anywhere it is stored (including on portable digital media, backup media, and in logs) by using any of the following approaches:</p> <ul style="list-style-type: none"> • One-way hashes based on strong cryptography (hash must be of the entire PAN) • Truncation (hashing cannot be used to replace the truncated segment of PAN) • Index tokens and pads (pads must be securely stored) • Strong cryptography with associated key-management processes and procedures 	<p>Oracle Advanced Security transparent data encryption (TDE) column encryption and tablespace encryption can be used to transparently encrypt the Primary Account Number on storage media.</p> <p>Oracle Advanced Security TDE has key management built-in. Encrypted data stays encrypted in the data files, redo logs, and backups.</p> <p>Oracle RMAN with Oracle Advanced Security can encrypt (and compress) the entire backup when backed up to disk.</p> <p>Oracle Data Pump with Oracle Advanced Security can encrypt (and compress) entire Database export files, either with the master encryption key from the source database, or a passphrase that can be securely shared with the receiving party.</p> <p>Oracle Secure Backup provides a solution for backing up and encrypting directly to tape storage.</p> <p>Encryption algorithms supported include AES with 256, 192, or 128 bit key length, as well as 3DES168.</p>
<p>3.5:</p>	<p>Protect encryption keys used for encryption of cardholder data against both disclosure and misuse</p>	<p>Oracle Advanced Security TDE table and tablespace keys are stored in the database and encrypted using a separate master encryption key that is stored in the Oracle Wallet (a PKCS#12 file on the operating system), or a Hardware Security Module (HSM).</p> <p>The Oracle Wallet is encrypted using the wallet password (based on PKCS#5); in order to open the wallet from within the database requires the 'alter system' privilege.</p> <p>Oracle Database Vault command rules can be implemented to further restrict who, when, and where the 'alter system' command can be executed.</p>

	3.5.1:	Restrict access to cryptographic keys to the fewest number of custodians necessary	<p>In Oracle, nobody needs access to the master encryption key. Designated individuals like a DBA or Database Security Administrator (DSA) need to know the wallet password or the HSM authentication string and have the 'alter system' privilege in order to open the wallet or HSM and make the master encryption key available to the database.</p> <p>Oracle Database Vault command rules can be implemented to restrict who, when, and where the 'alter system' command can be executed.</p>
	3.5.2:	Store cryptographic keys securely in the fewest possible locations and forms	<p>There is only one master encryption key per database. The key can be stored in the Oracle Wallet or an HSM device for centralized master encryption key management.</p>
3.6:	3.6.1:	Generation of strong cryptographic keys	<p>Oracle Advanced Security TDE utilizes RSA's industry-proven libraries for generating strong cryptographic keys. If a hardware security module (HSM) is used, then the HSM takes care of master encryption key management including key generation and storage.</p>
	3.6.2:	Secure cryptographic key distribution	<p>(Customer internal policy).</p> <p>In Oracle Data Guard environments, the wallet containing the master encryption keys needs to be copied to the secondary instances and opened.</p> <p>In Oracle Database 11g Release 2 RAC environments, it is recommended to store the Oracle Wallet in a shared location to fully leverage cross-node synchronization of master encryption key, re-key, or wallet open/close operations.</p>
	3.6.3:	Secure cryptographic key storage	<p>Oracle Advanced Security TDE column and tablespace keys are stored in the database and encrypted using the master key, which is stored in the Oracle Wallet or in a HSM device.</p>

		<p>3.6.4: Cryptographic key changes for keys that have reached the end of their crypto period (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)</p>	<p>Oracle Advanced Security TDE column encryption provides the ability to independently re-key the master encryption and/or table keys. Starting with Oracle Database 11g Release 2, the master encryption key for TDE tablespace encryption can be re-keyed as well. For PCI compliance, re-keying (rotating) the master encryption key often is sufficient.</p>
		<p>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</p>	<p>Master encryption keys should not be deleted from the wallet or HSM due to backup and recovery requirements.</p>
		<p>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)</p>	<p>With Oracle Advanced Security TDE, only the database needs 'access' to the master encryption key stored in wallet or HSM. Split knowledge of the key is not possible. However, the Oracle Wallet password can be split between multiple administrators and Oracle Database Vault can be used to enforce multiple separate logins to execute the 'alter system' command used to open the Oracle Wallet or HSM.</p>
		<p>3.6.7: Prevention of unauthorized substitution of cryptographic keys</p>	<p>Enforced by Oracle Advanced Security TDE</p>

4:	ENCRYPT TRANSMISSION OF CARDHOLDER DATA ACROSS OPEN, PUBLIC NETWORKS	
<p>Sensitive information must be encrypted during transmission over networks that are easily accessed by malicious individuals. Misconfigured wireless networks and vulnerabilities in legacy encryption and authentication protocols continue to be targets of malicious individuals who exploit these vulnerabilities to gain privileged access to cardholder data environments.</p>		
4.1:	<p>Use strong cryptography and security protocols (for example, SSL/TLS, IPSec, SSH, etc.) to safeguard sensitive cardholder data during transmission over open, public networks.</p> <p><i>Examples of open, public networks that are in scope of the PCI DSS include but are not limited to:</i></p> <ul style="list-style-type: none"> • <i>The Internet</i> • <i>Wireless technologies</i> • <i>Global System for Mobile communications (GSM), and</i> • <i>General Packet Radio Service (GPRS)</i> 	<p>Both Oracle Database and Oracle Fusion Middleware support encryption of network traffic using SSL/TLS.</p> <p>For the Oracle Database, SSL/TLS support is provided by Oracle Advanced Security.</p>
MAINTAIN A VULNERABILITY MANAGEMENT PROGRAM		
6:	DEVELOP AND MAINTAIN SECURE SYSTEMS AND APPLICATIONS	
<p>Unscrupulous individuals use security vulnerabilities to gain privileged access to systems. Many of these vulnerabilities are fixed by vendor-provided security patches, which must be installed by the entities that manage the systems. All critical systems must have the most recently released, appropriate software patches to protect against exploitation and compromise of cardholder data by malicious individuals and malicious software.</p> <p><i>Note: Appropriate software patches are those patches that have been evaluated and tested sufficiently to determine that the patches do not conflict with existing security configurations. For in-house developed applications, numerous vulnerabilities can be avoided by using standard system development processes and secure coding techniques.</i></p>		
6.1:	<p>Ensure that all system components and software have the latest vendor-supplied security patches installed. Install critical security patches within one month of release</p>	<p>Automated by Oracle Enterprise Manager Grid Control</p>
6.2:	<p>Establish a process to identify and assign a risk ranking to newly discovered security vulnerabilities.</p>	<p>(Customer internal policy)</p> <p>Subscribe to the 'Electronic Subscriptions' section on OTN and be sure to check the box next to the Oracle Security Alerts and click 'Continue' to confirm.</p>

6.4:	6.4.3:	Production data (live PANs) are not used for testing or development	Oracle Data Masking de-identifies credit card numbers and other sensitive information for testing and development environments.
	6.4.5	<p>Change control procedures for the implementation of security patches and software modifications. Procedures must include the following:</p> <p>6.4.1 Documentation of impact</p> <p>6.4.2 Documented change approval by authorized parties</p> <p>6.4.3 Functionality testing to verify that the change does not adversely impact the security of the system</p> <p>6.4.4 Back-out procedures</p>	<p>Database change control procedures can be automated with Oracle Change Management. Also BPEL Process Manager can be used for process management of change control, security procedures in general.</p> <p>Before implementing changes, Change Management analyzes change dependencies and generates an impact report</p> <p>Provides information to undo changes</p>
	6.5:	6.5.1:	Injection flaws, particularly SQL injection. Also consider OS Command Injection, LDAP and XPath injection flaws as well as other injection flaws
	6.5.3:	Insecure cryptographic storage	The Oracle Advanced Security TDE master encryption key is stored in an encrypted file, the Oracle Wallet, which is encrypted by a passphrase (the wallet 'password'), based on the PKCS#5 standard. For high-assurance (FIPS 140-2 or 3 certification), the TDE master encryption key can be stored in a HSM.
	6.5.4:	Insecure communications	Oracle Advanced Security network encryption enables encrypted network connections to and from the Oracle database with SSL/TLS.

IMPLEMENT STRONG ACCESS CONTROL MEASURES		
7:	RESTRICT ACCESS TO CARDHOLDER DATA BY BUSINESS NEED TO KNOW	
	<p>To ensure critical data can only be accessed by authorized personnel, systems and processes must be in place to limit access based on need to know and according to job responsibilities.</p> <p>"Need to know" is when access rights are granted to only the least amount of data and privileges needed to perform a job.</p>	
	7.1:	<p>Limit access to system components and cardholder data to only those individuals whose job requires such access. Access limitations must include the following:</p>
	7.1.1:	<p>Restriction of access rights to privileged user IDs to least privileges necessary to perform job responsibilities</p> <p>Oracle Database Vault realms restrict access to application / cardholder data from privileged users / DBAs.</p>
7.1.2:	<p>Assignment of privileges is based on individual personnel's job classification and function</p> <p>Oracle Database Vault factors and commands rules enable strict controls on access to applications, data and databases.</p> <p>Oracle Database Vault separation of duties prevents unauthorized administrative actions in the Oracle Database.</p> <p>Oracle Label Security provides additional security attributes for determining need-to-know.</p> <p>Oracle Virtual Private Database provides basic runtime masking.</p> <p>Oracle Database object privileges and database roles provide basic security.</p> <p>Oracle Identity Manager provides enterprise user provisioning only to permitted computing and application resources and data.</p> <p>Oracle Identity Analytics defines roles to provide granular definition of jobs and functions, as well as short-term assignments.</p>	

	<p>7.2:</p>	<p>Establish an access control system for systems components with multiple users that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed.</p> <p>This access control system must include the following:</p> <p>7.2.1 Coverage of all system components</p> <p>7.2.2 Assignment of privileges to individuals based on job classification and function</p> <p>7.2.3 Default "deny-all" setting</p>	<p>Oracle Database Vault realms restrict access to application / cardholder data from privileged users / DBA.</p> <p>Oracle Database Vault factors and commands rules enable strict controls on access to applications, data and databases.</p> <p>Oracle Database Vault separation of duties prevents unauthorized administrative actions in the Oracle Database.</p> <p>Oracle Label Security provides additional security attributes for determining need-to-know.</p> <p>Oracle Virtual Private Database provides basic runtime masking.</p> <p>Oracle Database object privileges and database roles provide basic security.</p> <p>Oracle Access Manager provides centralized access control, authorization and authentication.</p> <p>Oracle Identity Manager provides enterprise user provisioning only to permitted computing and application resources and data.</p> <p>Oracle Identity Analytics defines roles to provide granular definition of jobs and functions, as well as short-term assignments.</p>
<p>8:</p>	<p>ASSIGN A UNIQUE ID TO EACH PERSON WITH COMPUTER ACCESS</p>		
<p>Assigning a unique identification (ID) to each person with access ensures that each individual is uniquely accountable for his or her actions. When such accountability is in place, actions taken on critical data and systems are performed by, and can be traced to, known and authorized users.</p> <p><i>Note: These requirements are applicable for all accounts, including point-of-sale accounts, with administrative capabilities and all accounts used to view or access cardholder data or to access systems with cardholder data. However, requirements 8.1, 8.2 and 8.5.8 through 8.5.15 are not intended to apply to user accounts within a point-of-sale payment application that only have access to one card number at a time in order to facilitate a single transaction (such as cashier accounts).</i></p>			

	8.1:	Assign all users a unique ID before allowing them to access system components or cardholder data	<p>Oracle Database authentication supports dedicated user accounts.</p> <p>Oracle Identity Manager provides enterprise user provisioning and unique identifiers as well as correlates all unique identifiers back to the individuals.</p> <p>Oracle Access Manager provides centralized access control, authorization and authentication.</p>
	8.2:	<p>In addition to assigning a unique ID, employ at least one of the following methods to authenticate all users:</p> <ul style="list-style-type: none"> • Something you know, such as a password or passphrase • Something you have, such as a token device or smart card • Something you are, such as a biometric 	<p>Oracle Advanced Security provides strong authentication via Kerberos, PKI, and RADIUS.</p> <p>Oracle Access Manager supports strong authentication (tokens, smart cards, X. 509 certificates, forms) as well as passwords. Provides hierarchies of authentication for different levels of security requirements.</p> <p>Oracle Adaptive Access Manager provides two-factor authentication in software form. Also performs real time risk analysis to determine if additional authentication challenges are necessary.</p>
	8.3:	<p>Incorporate two-factor authentication for remote access (network-level access originating from outside the network) to the network by employees, administrators, and third parties. (For example, 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)</p>	<p>Oracle Advanced Security provides strong authentication via Kerberos, PKI, and RADIUS.</p> <p>Oracle Access Manager supports strong authentication (tokens, smart cards, X. 509 certificates, forms) as well as passwords. Provides hierarchies of authentication for different levels of security requirements.</p> <p>Oracle Adaptive Access Manager provides two-factor authentication in software form. Also performs real time risk analysis to determine if additional authentication challenges are necessary.</p>

8.4:	Render all passwords unreadable during transmission and storage on all system components using strong cryptography.	Oracle Database encrypts database user passwords during the authentication process and in storage within the database.	
8.5:	Ensure proper user identification and authentication management for non-consumer users and administrators on all system components as follows:	Oracle Identity Manager Oracle Access Manager	
	8.5.1:	Control addition, deletion, and modification of user IDs, credentials, and other identifier objects	Limit access to Oracle administration account SYSDBA. Deploy Oracle Database Vault for additional separation of duties. Deploy Oracle Identity Manager for enterprise user provisioning.
	8.5.2:	Verify user identity before performing password resets	Oracle Database authentication Oracle Database Vault separation of duties Oracle Access Manager challenge/response for password reset
	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	Create Oracle database account with password expired. Oracle Identity Manager
	8.5.4:	Immediately revoke access for any terminated users	Oracle Identity Manager and Oracle Database Enterprise User Security
	8.5.5:	Remove/disable inactive user accounts at least every 90 days	Oracle Identity Manager and Oracle Database Enterprise User Security
	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	Oracle Enterprise Manager account management Oracle Database Vault factors and command rules Oracle Database Vault realms to prevent access to application data Oracle Database Vault separation of duties Oracle Identity Manager

	8.5.8:	Do not use group, shared, or generic accounts and passwords, or authentication methods	(Customer internal policy); Oracle Database dedicated user accounts; Oracle Database Enterprise User Security; Oracle Database Proxy authentication; Oracle Identity Manager
	8.5.9:	Change user passwords at least every 90 days	Oracle Database profiles Oracle Identity Manager Oracle Access Manager
	8.5.10:	Require a minimum password length of at least seven characters	Oracle Database password complexity check (See here for recommendations for a strong password)
	8.5.11:	Use passwords containing both numeric and alphabetic characters	Oracle Identity Manager
	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	Oracle Database profiles Oracle Identity Manager Oracle Access Manager
	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	Oracle Database authentication Oracle Advanced Security Kerberos, PKI, and RADIUS support
REGULARLY MONITOR AND TEST NETWORKS			
10:	<i>TRACK AND MONITOR ALL ACCESS TO NETWORK RESOURCES AND CARDHOLDER DATA</i>		
	Logging mechanisms and the ability to track user activities are critical in preventing, detecting, or minimizing the impact of a data compromise. The presence of logs in all environments allows thorough tracking, alerting, and analysis when something does go wrong. Determining the cause of a compromise is very difficult, if not impossible, without system activity logs.		

10.1:	Establish a process for linking all access to system components (especially access done with administrative privileges such as root) to each individual user	<p>(Customer internal policy); Establish dedicated DBA accounts in the database.</p> <p>Oracle Database Vault separation of duties for more stringent controls on database administration.</p> <p>Oracle Audit Vault audit data consolidation for enterprise reports and alerting.</p> <p>Oracle Identity Manager provides enterprise user provisioning.</p> <p>Oracle Identity Analytics defines roles that determine user access and authorization levels.</p> <p>Oracle Access Manager controls access, authorization and authentication permissions to system components.</p>	
10.2:	Implement automated audit trails for all system components to reconstruct the following events:	(see below for individual replies)	
	10.2.1:	All individual user accesses to cardholder data	<p>Oracle Database Auditing; Oracle Database Fine Grained Auditing (FGA) on cardholder data; Oracle Audit Vault reports and alerts</p> <p>Oracle Identity Manager</p> <p>Oracle Access Manager audit reports</p> <p>Oracle Identity Analytics</p>

	10.2.2:	All actions taken by any individual with root or administrative privileges	<p>Establish dedicated DBA accounts in the database. Optionally use proxy authentication to limit the number of accounts with DBA privileges but still audit.</p> <p>Oracle Database Vault realms and separation of duties for more stringent controls on database administration</p> <p>Oracle Database Vault realm reports</p> <p>Oracle Audit Vault audit data consolidation for enterprise reports and alerting</p> <p>Oracle Identity Manager</p> <p>Oracle Access Manager audit reports</p> <p>Oracle Identity Analytics</p>
	10.2.3:	Access to all audit trails	<p>Database audit data can be stored in Oracle Audit Vault.</p> <p>Oracle Access Manager, Oracle Identity Analytics, and Oracle Identity Manager audit reports</p>
	10.2.4:	Invalid logical access attempts	Oracle Access Manager audit reports
	10.2.5:	Use of identification and authentication mechanisms	<p>Oracle Database authentication</p> <p>Oracle Advanced Security Kerberos, PKI, RADIUS authentication</p> <p>Oracle Access Manager audit reports</p> <p>Oracle Identity and Access Management Suite</p>
	10.2.7:	Creation and deletion of system-level objects	Oracle Database auditing
	10.3:	Record at least the following audit trail entries for all system components for each event:	
	10.3.1:	User identification	Oracle Access Manager audit reports
	10.3.2:	Type of event	Oracle Database auditing
	10.3.3:	Date and time	Oracle Audit Vault audit data consolidation, reporting, alerting and protection
	10.3.4:	Success or failure indication	Oracle client identifiers for identity propagation across single connection
	10.3.5:	Origination of event	

	10.3.6:	Identity or name of affected data, system component, or resource	
	10.5:	Secure audit trails so they cannot be altered.	Oracle Audit Vault audit data consolidation protects audit data in transit and storage.
	10.5.1:	Limit viewing of audit trails to those with a job-related need	Oracle Audit Vault separation of duties limits access to audit data.
	10.5.2:	Protect audit trail files from unauthorized modifications	Oracle Audit Vault separation of duties prevents access and modification of audit data by administrators (DBA)
	10.5.3:	Promptly back-up audit trail files to a centralized log server or media that is difficult to alter	Oracle Audit Vault audit data consolidation provides a scalable and secure audit warehouse
	10.5.5:	Use file integrity monitoring and 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)	Oracle Audit Vault audit data consolidation protects audit data in transit Oracle Audit Vault separation of duties prevents access and modification of audit data by administrators (DBA)
	10.6:	Review logs for all system components at least daily. Log reviews must include those servers that perform security functions like intrusion detection system (IDS) and authentication, authorization, and accounting protocol (AAA) servers (for example, RADIUS)	Oracle Audit Vault provides out-of-box reports, customizable alerts and an alert dashboard for monitoring audit data. Customized reports can be generated using Oracle Application Express, Oracle BI Publisher and 3 rd party tools. The Oracle Audit Vault warehouse schema is published. Oracle Access Manager and Identity Manager provide logs of all user activity and provisioning/de-provisioning.
	10.7:	Retain audit trail history for at least one year, with a minimum of three months immediately available for analysis (for example, online, archived, or restorable from back-up)	Oracle Audit Vault provides a scalable and secure audit warehouse for storing large volumes (Terabytes) of audit data.

APPENDIX A: PCI DSS APPLICABILITY FOR SHARED HOSTING PROVIDERS			
A:	HOSTING PROVIDERS PROTECT CARDHOLDER DATA ENVIRONMENT		
	As referenced in Requirement 12.8, all service providers with access to cardholder data (including shared hosting providers) must adhere to the PCI DSS. In addition, Requirement 2.4 states that shared hosting providers must protect each entity's hosted environment and data. Therefore, shared hosting providers must additionally comply with the requirements in this Appendix.		
	A.1:	<p>Protect each entity's (that is merchant, service provider, or other entity) hosted environment and data, per A.1.1 through A.1.4:</p> <p>A hosting provider must fulfill these requirements as well as all other relevant sections of the PCI DSS.</p>	
	A.1.1:	Ensure that each entity only runs processes that have access to that entity's cardholder data environment	<p>Oracle Database Vault realms restrict access to application / cardholder data from highly privileged users / DBA.</p> <p>Oracle Database Vault factors and commands rules enable strict controls on access to applications, data and databases.</p> <p>Oracle Database Vault separation of duties prevents unauthorized administrative actions in the Oracle Database.</p> <p>Oracle Label Security provides additional security attributes for determining need-to-know.</p> <p>Oracle Virtual Private Database provides basic runtime masking based on need-to-know.</p> <p>Oracle Database object privileges and database roles provide basic security.</p> <p>Oracle Identity Manager provides enterprise user provisioning to computing resources.</p>

		<p>A.1.2:</p>	<p>Restrict each entity's access and privileges to own cardholder data environment only</p>	<p>Oracle Database Vault realms restrict access to application / cardholder data from highly privileged users / DBA.</p> <p>Oracle Database Vault factors and commands rules enable strict controls on access to applications, data and databases.</p> <p>Oracle Database Vault separation of duties prevents unauthorized administrative actions in the Oracle Database.</p> <p>Oracle Label Security provides additional security attributes for determining need-to-know.</p> <p>Oracle Virtual Private Database provides basic runtime masking based on need-to-know.</p> <p>Oracle Database object privileges and database roles provide basic security.</p> <p>Oracle Identity Manager provides enterprise user provisioning to computing resources.</p> <p>Oracle Access Manager provides centralized access control, authorization and authentication.</p> <p>Oracle Identity Analytics provides role definition for granular job assignment and access.</p>
		<p>A.1.3:</p>	<p>Ensure logging and audit trails are enabled and unique to each entity's cardholder data environment and consistent with PCI DSS Requirement 10</p>	<p>Oracle Audit Vault policies can be easily deployed to enterprise databases, enabling consistent auditing of access to cardholder data.</p> <p>Oracle Access Manager provides audit reporting for all user activity.</p>

		A.1.4:	Enable processes to provide for timely forensic investigation in the event of a compromise to any hosted merchant or service provider	<p>Oracle Audit Vault provides out-of-box reports, customizable alerts and an alert dashboard for monitoring audit data.</p> <p>Customized reports can be generated using Oracle Application Express, Oracle BI Publisher and 3rd party tools. The Oracle Audit Vault warehouse schema is published.</p>
--	--	--------	---	---

The Challenges of PCI Data Protection

Cardholder Data at Risk

PCI DSS requires companies to protect cardholder data. Essentially, that covers the primary account number (PAN), cardholder name, service code and expiration date — if stored in conjunction with the card numbers (as they typically are, in order to be of practical business use).

This information has been highly prized by criminals since the days when merchants took carbon copy imprints of cards on manual devices. Today, external hackers or malicious insiders can harvest millions of credit card records stored in backend databases, selling them on the international Internet black markets and/or purchasing high-ticket items.

Unauthorized or, more typically, unnecessary database and/or operating system access exposes cardholder data, often to trusted privileged users, such as database and system administrators, as well as developers and other employees who have no need to access or even see this information. Insecure database configurations, the result of flawed deployment or subsequent errors, leave cardholder data highly susceptible to theft.

Encryption Comes of Age

Compliance Pushes Adoption

Data encryption is an essential part of an enterprise PCI compliance program. However, in the past, enterprises have shied away from encryption, despite its obvious security benefits, because it has been difficult to deploy and manage, resulting in abandoned projects and weak, insecure implementation. Key management issues, in particular, have bedeviled encryption projects. Native or home-grown encryption solutions typically do not scale well for the enterprise, and

companies may have to turn to a mix of tools to address data at rest, in motion and in backups. Home-grown or cobbled-together solutions have to face the challenges of integrating strong authentication for more robust security for database communications.

PCI DSS and other compliance mandates have changed the landscape forever. The question has become not whether to encrypt, but how to encrypt in a way that is secure, scalable and manageable.

The Oracle Encryption Solution

Oracle Advanced Security provides multiple encryption capabilities in a single seamless product, encrypting data in storage, in transit and on backup media. It also provides strong authentication as an alternative to passwords.

Requirement 3.4 directs organizations to “render PAN unreadable anywhere it is stored.” While several options are available, “Strong cryptography with associated key management processes and procedures” is the logical solution for protecting cardholder data in database stores, using robust automated tools to address data protection on an enterprise scale.

Oracle Advanced Security transparent data encryption (TDE) transparently encrypts data when written to disk and decrypts it after a user has been successfully authenticated and authorized. It supports both column and tablespace encryption. TDE prevents attempts to bypass the database and access files directly. Oracle supports transparently encrypting specific sensitive columns with TDE column encryption or encrypting entire applications with TDE tablespace encryption. Using Oracle Enterprise Manager, a column can be quickly and easily encrypted or an entire encrypted tablespace can be created to store all application tables.

Oracle Advanced Security meets the PCI requirement for strong cryptography — that is, “based on industry-tested and accepted algorithms” — supporting AES (256, 192 and 128 bits) and 3DES168.

Securing Data in Motion

Requirement 4 addresses the encryption of cardholder data across open, public networks (Internet, GSM, etc.). Specifically, 4.1 requires use of strong cryptography and security protocols such as SSL/TLS.

Oracle Advanced Security provides protection for communication to and from the Oracle Database, preserving privacy and confidentiality of data by preventing data sniffing, data loss, replay and person-in-the-middle attacks. It provides both native network encryption and SSL/TLS based encryption for enterprises with PKI infrastructure. The Oracle Database can be configured to reject connections from clients that do not encrypt data or optionally to allow unencrypted connections for deployment flexibility.

Securing Backup Data

PCI requires encryption of backup cardholder information to protect against lost or stolen tapes or other backup media.

Existing backup procedures will backup the TDE protected tablespaces as encrypted, and table columns protected using TDE column encryption will automatically remain encrypted on backup media. Encryption of all database files, including the SYSTEM tablespace, can be achieved by using Oracle RMAN and TDE together. Oracle RMAN provides the ability to use the TDE encryption algorithms and the master encryption key to encrypt the entire database backup.

In addition, Oracle Secure Backup encrypts tapes and provides centralized tape backup management.

Using Masked Data

Credit card information must be identifiable to both the customer and the merchant for transaction purposes via computer screen displays, receipts, faxes, and reports, but display of full PANs could expose the data to theft.

However, cardholder information and other production data is important for development and testing activities, potentially putting large amounts of cardholder data in the hands of developers, QA personnel, etc. who should not be allowed to see it according to the PCI requirements. To mitigate this scenario, developers will sometimes generate fake data to simulate live production data, but this is not always as reliable, especially for testing purposes.

In any of these cases, the trick is to mask the information so that the visible data can neither compromise security nor privacy. Data masking simply substitutes false values for real ones, keeping the data formats, regardless of the number and type of fields.

PCI section 3.3 requires that PANs be masked, revealing at most the first six and last four digits. Requirement section 6.4.3 prohibits the use of live PANs for development.

However, data masking without automated tools can become challenging given the number of new and ongoing development projects in an enterprise and the demands of PCI and other regulatory requirements.

Oracle Data Masking replaces credit card numbers and associated cardholder information with realistic but false values, allowing production data to be safely used for transactions, development, testing, or sharing in conjunction with out-sourced or off-shore partners for non-production purposes. Oracle Data Masking uses a library of templates and format rules, transforming data consistently to maintain referential integrity for applications.

Solving Key Management

Encryption's Achilles Heel

Generating and protecting encryption keys while maintaining data availability has traditionally been a major barrier to implementing encryption, especially on an enterprise scale.

Key management is complex and challenging, and often fails because issuance, storage, and renewing are difficult. Worse yet, lost keys can make important data permanently unrecoverable.

The frequent result is that key management becomes a hallow security control, as IT managers give way to more pressing priorities and pressure from the business side to relax key management controls. As a consequence, keys become widely available to multiple users, rendering encryption ineffective.

PCI Requirements for Key Management

The PCI Standard goes into some detail on the subject of key management in requirements 3.5 and 3.6. Section 3.5 requires strong protection for the keys to prevent unauthorized use and, therefore, access to cardholder data. Specifically, organizations must restrict access to keys to the fewest number of people possible (key custodians) and store keys securely in the fewest locations possible. Section 3.6 details key implementation considerations including:

- Generation of strong cryptographic keys
- Secure key distribution (not in the clear and only to key custodians)
- Secure, encrypted storage
- Periodic, preferably automated key changes
- Retirement or replacement of unused keys or keys that may have been compromised
- Split knowledge and dual control to prevent one person having access to the whole key

Oracle Key Management

Oracle Advanced Security Transparent Data Encryption (TDE) addresses these requirements with built-in key management. TDE automatically creates an encryption key behind the scenes. The two-tier system includes a master encryption key that protects the data encryption keys. The master key is securely stored outside of the database in an Oracle Wallet, a PKCS#12 formatted file which is password-encrypted. The master key can be stored in a Hardware Security Module (HSM) device for higher assurance. Oracle uses the industry standard PKCS#11 interface to communicate with numerous HSM and external key management systems, including those provided by Thales, RSA, Safenet, and Utimaco.

Building Protection Around Cardholder Data

Strong Authentication

The need for strong authentication is a recurring theme throughout PCI DSS. The use of strong authentication is encouraged in requirement 8, which calls for a unique user ID for each person with computer access and commensurate authentication.

Specifically, 8.5.16 requires authentication for access to databases containing cardholder data.

Oracle Advanced Security provides strong authentication solutions including Kerberos, PKI, and RADIUS.

Monitoring, Tracking and Auditing

Strong data security policies and controls around cardholder data requires continuous monitoring, tracking and auditing to assure that they are operating as intended and are being properly enforced. Having the ability to verify that controls are effective and detect and address unauthorized activity and errors completes a robust cardholder data protection program while enabling organizations to verify to QSAs that their policies and controls are in full force.

PCI DSS requirement 10 requires organizations to track and monitor access to cardholder data. The standard places strong emphasis on audit capabilities, particularly requirement 10.2, which mandates implementation of audit trails for individual user access to cardholder data.

Oracle Audit Vault provides security personnel with the ability to detect and alert on activities that may indicate attempts to gain unauthorized access and/or abuse system privileges. It continuously monitors the audit data collected, evaluating the activities against defined alert conditions, including system events such as changes to application tables and creating privileged users.

Oracle Audit Vault provides powerful built-in reports to monitor a wide range of activity to support QSA assessments, internal audits, security programs and operational requirements. Oracle Audit Vault provides powerful built-in reports to monitor a wide range of activity. Rules can be put in place to automatically highlight specific rows so that report users can quickly spot suspicious or unauthorized activity. Out-of-the-box reports include information on database account management, roles and privileges, object management, and login failures.

Oracle Audit Vault provides centralized management of Oracle database audit settings, simplifying the job of the IT security and internal auditors in managing audit settings across the enterprise and demonstrating compliance and repeatable controls to QSAs.

Maintaining Secure Configurations

Attackers can and will exploit configuration weaknesses in database instances, one of several ways malicious outsiders can harvest cardholder data. Weak configuration settings, lack of configuration enforcement standards, missing patches, errors and configuration “drift” as changes are made, including unauthorized changes made outside of change management procedures, all can leave cardholder data vulnerable to attack.

Organizations working to monitor configurations and detect and correct issues are hampered by asset discovery, tracking, and configuration analysis procedures which often are manual, slow and prone to error.

PCI DSS addresses the need for secure configuration practices. Requirement 2 directs companies to change vendor-supplied defaults such as passwords and SNMP community strings and to eliminate any unnecessary accounts. Section 2.2 requires development of configuration standards that are consistent with accepted hardening standards and that address known configuration vulnerabilities. Requirement 6 mandates up-to-date patching.

Oracle Configuration Management provides asset discovery, tracking and change detection for database configurations (as well as operating system, hardware, application server and packaged application configurations). It collects detailed configuration information at regular intervals, and provides analytics and search capabilities for specific configuration values across the enterprise. Oracle Configuration Management allows IT administrators to detect change, to correct “configuration drift,” and validate planned changes. Oracle Configuration Management also provides strong compliance reporting to validate controls and configuration procedures for audit.

Managing Privileged Accounts

The Enemy Within

In a real-world environment, important items such as privileged user access controls, privileged user authorization, and separation of duties are difficult to manage and maintain when you are left to rely on manual procedures. This is especially true when you consider common factors like changing personnel, changing requirements, new assets, and privilege “drift” as administrators share user IDs and as privileges are assigned in violation of policy or outside of required workflow processes. Reviewing logs for unauthorized or inappropriate activity is time-consuming, resource-intensive and prone to error, as is fulfilling audit and reporting requirements and requests for information.

Organizations, ironically, often tend to do a better job managing general user account security than privileged accounts — system administrators, DBAs, etc. — although these accounts

present by far the higher security risk based on their ability to not only access sensitive information, such as cardholder data, but to configure systems, modify databases and grant privileges to others.

There are a number of reasons privileged user accounts are often poorly controlled. For years, the perception has been that high-authorization individuals, such as database administrators, are trusted individuals, while other employees, as a generalization, are less trusted. While the overwhelming majority of privileged users are, of course, trustworthy, there have been startling examples of abuse for personal gain or retribution. In addition, those attempting to break into systems will target privileged user accounts for their potentially wide ranging access to systems such as databases containing sensitive application data.

The logistics of managing these accounts are difficult. It is easier to grant too much privilege than to risk impeding operations by granting too little. Privileged authorization frequently is granted in an emergency — at least, a perceived emergency — but never is revoked. Privileged accounts and their passwords tend to be shared among multiple individuals as an expedience.

As a result of losing of control over privileged accounts, monitoring, tracking and auditing of individual user activity, always problematic, becomes next to impossible.

The potential consequences of poorly controlled, poorly defined privileged accounts become obvious when you think in terms of complete access to and control over sensitive databases.

Regulatory pressures force organizations to take stock of privileged users and privileged accounts and to remediate these long-standing security risks. More than anyone in the organization, privileged user roles must be defined on a highly granular level, controlling what systems and data they can access and what operations they are permitted to perform. Particular attention must be paid to separation of duties.

On the other hand, the systems for managing and monitoring privileged users must be flexible enough to respond to legitimate needs without slowing operations or interrupting the business, or else there will be a rapid reversion to the problems of overly broad and widely shared high-privilege accounts.

PCI Recognizes the Risk

The concern over privileged user access and authorization is reflected heavily in PCI DSS. The global requirements for access control, user authentication, and tracking user activity should be emphasized in the case of privileged accounts, and it is underscored numerous times throughout the requirements. Organizations will have won a hollow victory if they assert a reasonable level of access control over the general user population but fail to bring privileged user accounts into line.

Pay particular attention to privileged users when enforcing requirement 7 directives to restrict access to cardholder data by business need to know. Section 7.1 explicitly directs organizations to restrict access rights to privileged user IDs to the least privileges necessary to perform job responsibilities. Privileges, the requirement continues, should be based on job classification and function. It is critical to review and, if necessary, redefine these classifications, particularly for high-privilege accounts where the risk is greatest.

The risk attached to privileged users also puts an exclamation point on the directives for robust authentication in requirement 8; in particular, the need to authenticate all access to databases containing cardholder data.

Requirement 10, which covers tracking and monitoring of access to network resources and cardholder data, specifies, “Establish a process for linking all access to system components (especially access done *with administrative privileges*, such as root) to each individual user.”

Oracle Manages Privileged Accounts

Oracle Database Vault enables organizations to assert control over privileged users, protecting critical assets such as cardholder data from unnecessary risk and exposure. Oracle Database Vault provides powerful, yet flexible and easy-to-manage mechanisms to strictly define privileged user access and authorization and to enforce separation of duties.

Oracle Database Vault provides strong privileged user access control, using the concept of “realms” to create a firewall within the database. Sensitive tables or applications can be placed in a realm, so privileged users can do their jobs without having access to cardholder data.

Command rules enable multi-factor authorization controls to restrict access to databases to a specific subnet or application server, creating a trusted path for data access. Built-in factors such as IP address, time-of-day and authentication method can be used in a flexible and adaptable manner to enforce access control to meet PCI compliance and business requirements.

Oracle Database Vault provides baseline separation of duties out of the box, with three separate roles defined by responsibilities within the database: account management, security administration and database administration. Each of these roles can be customized to meet specific business requirements.

Oracle Database Vault provides numerous out-of-the box reports that provide information on such things as data access attempts blocked by realms. So, for example, if a DBA attempts to access a data application table protected by a realm, Oracle Database Vault will block that access and create an audit record that can be viewed in the realm violation report.

As discussed earlier, Oracle Audit Vault provides robust monitoring, auditing and reporting capabilities, which can, of course, be applied to all types of privileged users.

Identity Management Enables Compliance, Empowers Business

Secure and Flexible Identity and Access Management

Database security is at the core of your PCI compliance program, directly protecting the crown jewels: cardholder data. Around the core, robust identity management, built around access control, authorization and authentication, creates the enterprise environment in which security is tightly integrated into freely flowing business processes. The result is end-to-end security that meets your compliance requirements while empowering your business.

Controlling user and application access is essential to meeting not only the letter of PCI DSS requirements but the real intent of actually protecting cardholder data. The reality is that cardholder data is not inert — it is one of the keys to commerce in highly complex, distributed enterprises that extend to partners, suppliers and vendors as well as your employees. Identity management is not just a collection of user IDs, passwords and file permissions. It is a dynamic ecosystem of users, applications, data, and private and public networks. It is in constant flux, as new access requests, new applications, new business initiatives and the constant inflow/outflow of people such as employees, contractors, and partners creates the need for limited or temporary privileges. This flux creates a fluid, high-risk environment which can quickly become unmanageable.

Your plan for identity management should include:

- Centrally managed access control separated from individual applications so that controls can be maintained efficiently, according to policy, across the enterprise.
- Well-defined, granular role-based access control (RBAC). Roles are created for particular job functions and the necessary permissions defined for each role. Roles can then be assigned to individuals, making it easy to add or change responsibilities.
- Timely and accurate provisioning and de-provisioning of employees, contractors and authorization privileges based on a well-defined evaluation and approval workflow.
- Real-time monitoring and alerting, comprehensive and timely auditing, and strong reporting.

PCI Requirements for Identity and Access Management

As discussed earlier in regard to privileged users, PCI DSS reflects the importance that the PCI council places on access control.

Requirement 7 restricts access to cardholder data based on a user's need to know. This access should be based on the principle of least privilege to perform their jobs and based on an individual's job classification and function.

Requirement 8 mandates a unique user ID to each person with computer access and sets requirements for authentication, including two-factor authentication where deemed appropriate (mandatory for remote access) and rules governing passwords. The requirement also covers provisioning. Important provisioning functions include:

- Control addition, deletion, and modification of user IDs, credentials and other identifier objects and limiting management to a small group with specific authority
- Immediately revoke access for any terminated users
- Remove/disable inactive user accounts at least every 90 days
- Enable accounts used by vendors for remote maintenance only during the time period needed

Think about this beyond the notion of direct database access and or privileged user activity. Without well defined roles and the authorization and authentication requirements that support them, how do you know who may have access, or may *gain* access, to cardholder data or to applications that have access to cardholder data? Which applications, in fact, have that access?

Requirement 10 addresses the critical area of tracking and monitoring all access to network resources and cardholder data. Note that PCI DSS does not focus solely on direct access to the data, recognizing that cardholder information exists in a live, dynamic production environment with many players and a lot of moving, changing parts.

It also is essential to note that first comes access control, authorization, provisioning and authentication rules, then monitoring user and application activities. If you do not have policies and processes that control correct, authorized activity, then how can you monitor for incorrect or unauthorized activity?

The requirement calls for a process that links all access to system components to individual users and for comprehensive audit trails to reconstruct all individual user access to cardholder data. Access to audit trails, invalid access attempts, initialization of audit logs, and creation and deletion of system-level objects also are covered in the requirements.

Identity Management Challenges

The identity management-related requirements of PCI DSS are perhaps the most difficult to implement, and they are even more difficult to manage and maintain. These problems are among the chief reasons that organizations are unable to sustain compliance efforts on a continuous basis. They may lead organizations to spend unreasonable time, manpower and money gathering and analyzing data (*if* it is available) in attempt to fix broken systems and remediate violations. Identity management problems lead to great difficulties verifying controls to QSAs and explaining why the controls do not, in fact, work much of the time.

Let's examine some of these challenges, which generally are the result of trying to enforce good policies with inefficient manual processes.

Access control, authorization and authentication is, of necessity, generally performed on a per application basis rather than centrally managed. The outcome is uneven policy enforcement, manpower-intensive administration, slow/inefficient response to changing business requirements, error prone results, and weakened security.

It is difficult to apply appropriate authentication controls consistently across users, groups, applications and data access in the absence of centralized, policy-based management.

Role-based access control is excellent in theory, but difficult to implement and maintain. It requires an enormous commitment of time and resources to define roles and establish a role-based approach across the enterprise. This is exacerbated because information about users, responsibilities and lines of reporting typically resides in silos throughout the organization.

It is nearly impossible to manage and enforce consistent policy across a complex, distributed enterprise.

Provisioning and de-provisioning users and user authorizations often is slow and impedes the business. Spreadsheet-based administration can effectively enforce policy but can become a bottleneck because of slow response by busy administrators and lag time awaiting evaluation and approvals by the responsible managers.

The process generally is error prone for a variety of reasons. Role-based controls are difficult to define and manage without automated systems, so individuals may be given too little, or more likely, too much privilege based on coarse individual and/or group assignments. Administrators are likely to error on the side of excessive authorization to assure that the individual has what he or she needs to perform their job. This is of even greater concern if the individual is a contractor or a vendor with high privilege inside of your organization.

In this kind of environment, "ghost" (a.k.a. rogue) accounts persist long after the individual has left, or temporary authorization was never withdrawn. The account might have belonged to a former employee who still has access to corporate systems and data, a contractor whose assignment is finished or an existing employee who was granted temporary rights or has changed jobs within the company.

Monitoring user activity is difficult, if not impossible. In most cases, there is no real-time monitoring and alerting capability. Because access control and, hence, monitoring is managed separately for each application or system, it is close to impossible to monitor individuals as a practical matter.

Auditing is similarly piecemeal and inefficient, as is reporting for PCI audits. Meeting regulatory and internal requirements often is inefficient because logs are split among applications and systems. This requires manual information-gathering, analysis and reporting, as well as correlation when multiple applications and systems are involved.

Oracle's Identity Management Solution

The Oracle suite of identity management products provides a fully integrated, centralized, managed and automated solution. It covers access control, authorization & authentication, granular role-based controls, and provisioning capabilities — all to help meet and exceed the directives in PCI requirements 7, 8 and 10.

Oracle Access Manager secures access control through centralized authorization, authentication and audit, enabling single sign-on capabilities transparent to the user. By separating authorization from the application, companies can manage and monitor access privileges on an enterprise-wide basis, according to business requirements and based on policies and customizable rules.

The access system provides centralized policy-based authorization services, allowing admins to define policies that restrict access to specific resources by user, role, group membership (static, nested or dynamic), time-of-day, day-of-week and IP address.

Oracle Access Manager implements PCI authentication requirements, supporting X.509 certificates, smart cards, two-factor tokens and forms-based authentication. It allows organizations to establish hierarchies of authentication, so that a user might require only password authentication for general login to standard applications and information sources but strong authentication, such as usage of tokens, for more sensitive access.

Oracle Identity Manager is a robust provisioning and de-provisioning product that enables organizations to quickly and easily provision users and provide both permanent and temporary authorization based on business requirements. Moreover, it enhances security and fulfills applicable sections of PCI Requirement 8. Oracle Identity Manager is most effective when used in conjunction with Oracle Identity Analytics, which provides fine-grained role-based access control, allowing companies to precisely define and assign roles according to need. This fulfills the requirement to assign access and authorization based on the principle of least privilege and job classification and function.

Oracle's identity management products provide powerful monitoring, auditing and reporting capabilities to effectively meet PCI requirement 10 terms for monitoring and audit. Oracle Access Manager sets and enforces policy-based authentication and monitors user access activity. Identity Manager detects rogue accounts and changes user access privileges.

Oracle Access Manager's auditing services provide detailed and flexible logging of monitored events such as authentication success or failure. Audit logs can be written either to a flat file or to a database and exported to any third-party reporting tool to produce comprehensive auditing reports.

Conclusion

PCI-DSS represents perhaps the most promising effort at industry self-policing we have seen since the ubiquitous use of the public Internet and the widespread growth in Internet fraud began moving information security concerns to the fore. As criminals moved online to exploit billions of insecure consumer information records, the credit card companies moved with the times to create a highly prescriptive blueprint for securing cardholder data. This blueprint provides a strong foundation that I.T. organizations can build upon to create good data security programs.

Compliance and data security have proven difficult for many companies, particularly those retailers whose security policies are not as mature as companies in some other industries. However, the goals of PCI DSS are attainable, and the requirements of the standard can be fulfilled in a sustainable continuous program. The path forward requires a combination of sound security policy and the support of automated tools from leaders in data security like Oracle that enable compliance while simultaneously empowering organizations to improve their business practices.



Sustainable Compliance for the Payment Card
Industry Data Security Standard
January 2011

Author:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com



| Oracle is committed to developing practices and products that help protect the environment

Copyright © 2011, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.